

内容审核

SDK 参考

文档版本 01
发布日期 2024-07-05



版权所有 © 华为技术有限公司 2024。保留一切权利。

非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。

商标声明



HUAWEI和其他华为商标均为华为技术有限公司的商标。

本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

注意

您购买的产品、服务或特性等应受华为公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为公司对本文档内容不做任何明示或暗示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

华为技术有限公司

地址： 深圳市龙岗区坂田华为总部办公楼 邮编： 518129

网址： <https://www.huawei.com>

客户服务邮箱： support@huawei.com

客户服务电话： 4008302118

安全声明

漏洞处理流程

华为公司对产品漏洞管理的规定以“漏洞处理流程”为准，该流程的详细内容请参见如下网址：

<https://www.huawei.com/cn/psirt/vul-response-process>

如企业客户须获取漏洞信息，请参见如下网址：

<https://securitybulletin.huawei.com/enterprise/cn/security-advisory>

目录

1 内容审核 SDK 简介	1
2 申请服务	3
3 获取认证信息	4
4 准备环境	5
5 SDK 列表及指导文档（通用版）	6
6 使用 SDK（Java）	7
6.1 Java 开发环境配置	7
6.2 SDK 获取和安装	7
6.3 SDK 调用示例	8
6.3.1 文本审核	8
6.3.1.1 文本内容审核（V3）	8
6.3.1.2 文本内容审核（V2）	9
6.3.2 图像审核	11
6.3.2.1 图像内容审核（V3）	11
6.3.2.2 图像内容审核（V2）	12
6.3.2.3 图像内容审核（批量）（V2）	13
6.3.2.4 图像内容审核（异步批量）（V2）	15
6.3.2.4.1 提交任务	15
6.3.2.4.2 查询任务	16
6.3.3 音频审核	17
6.3.3.1 创建音频内容审核作业	17
6.3.3.2 查询音频内容审核作业	18
6.3.4 视频审核	19
6.3.4.1 创建视频内容审核作业	19
6.3.4.2 查询视频内容审核作业	20
6.3.5 音频流审核	21
6.3.5.1 创建音频流内容审核作业	21
6.3.5.2 关闭音频流内容审核作业	22
6.3.6 视频流审核	23
6.3.6.1 创建视频流内容审核作业	23
6.3.6.2 关闭视频流内容审核作业	25

6.3.7 文档审核.....	25
6.3.7.1 创建文档内容审核作业.....	26
6.3.7.2 查询文档审核作业结果.....	27
7 使用 SDK (Python)	29
7.1 Python 开发环境配置.....	29
7.2 Python 语言 SDK 获取和安装.....	30
7.3 SDK 调用示例.....	30
7.3.1 文本审核.....	30
7.3.1.1 文本内容审核 (v3)	31
7.3.1.2 文本内容审核 (v2)	31
7.3.2 图像审核.....	32
7.3.2.1 图像内容审核 (V3)	32
7.3.2.2 图像内容审核 (V2)	33
7.3.2.3 图像内容审核 (批量) (V2)	34
7.3.2.4 图像内容审核 (异步批量) (V2)	35
7.3.2.4.1 提交任务.....	36
7.3.2.4.2 查询任务.....	36
7.3.3 音频审核.....	38
7.3.3.1 创建音频内容审核作业.....	38
7.3.3.2 查询音频内容审核作业.....	39
7.3.4 视频审核.....	39
7.3.4.1 创建视频内容审核作业.....	39
7.3.4.2 查询视频内容审核作业.....	40
7.3.5 音频流审核.....	41
7.3.5.1 创建音频流内容审核作业.....	41
7.3.5.2 关闭音频流内容审核作业.....	42
7.3.6 视频流审核.....	43
7.3.6.1 创建视频流内容审核作业.....	43
7.3.6.2 关闭视频流内容审核作业.....	44
7.3.7 文档审核.....	45
7.3.7.1 创建文档内容审核作业.....	45
7.3.7.2 查询文档审核作业结果.....	45
8 使用 SDK (.NET)	47
8.1 .NET 开发环境配置.....	47
8.2 .NET 环境 SDK 导入.....	47
8.3 SDK 调用示例.....	48
8.3.1 文本审核.....	48
8.3.1.1 文本内容审核 (V3)	48
8.3.1.2 文本内容审核 (V2)	49
8.3.2 图像审核.....	51
8.3.2.1 图像内容审核 (V3)	51
8.3.2.2 图像内容审核 (V2)	52

8.3.2.3 图像内容审核（批量）（V2）	54
8.3.2.4 内容审核-图像（异步批量）（V2）	55
8.3.2.4.1 提交任务	56
8.3.2.4.2 查询任务	57
8.3.3 音频审核	58
8.3.3.1 创建音频内容审核作业	59
8.3.3.2 查询音频内容审核作业	60
8.3.4 视频审核	62
8.3.4.1 创建视频内容审核作业	62
8.3.4.2 查询视频内容审核作业	63
8.3.5 音频流审核	66
8.3.5.1 创建音频流内容审核作业	66
8.3.5.2 关闭音频流内容审核作业	68
8.3.6 视频流审核	69
8.3.6.1 创建视频流内容审核作业	69
8.3.6.2 关闭视频流内容审核作业	70
8.3.7 文档审核	71
8.3.7.1 创建文档内容审核作业	72
8.3.7.2 查询文档审核作业结果	73
9 使用 SDK（GO）	75
9.1 GO 开发环境配置	75
9.2 GO 语言 SDK 获取和安装	75
9.3 SDK 调用示例	75
9.3.1 文本审核	75
9.3.1.1 文本内容审核（V3）	76
9.3.1.2 文本内容审核（V2）	76
9.3.2 图像审核	77
9.3.2.1 图像内容审核（V3）	77
9.3.2.2 图像内容审核（V2）	78
9.3.2.3 图像内容审核（批量）（V2）	79
9.3.2.4 内容审核-图像（异步批量）（V2）	80
9.3.2.4.1 提交任务	80
9.3.2.4.2 查询任务	81
9.3.3 音频审核	82
9.3.3.1 创建音频内容审核作业	82
9.3.3.2 查询音频内容审核作业	83
9.3.4 视频审核	84
9.3.4.1 创建视频内容审核作业	84
9.3.4.2 查询视频内容审核作业	85
9.3.5 音频流审核	86
9.3.5.1 创建音频流内容审核作业	86
9.3.5.2 关闭音频流内容审核作业	87

9.3.6 视频流审核.....	88
9.3.6.1 创建视频流内容审核作业.....	88
9.3.6.2 关闭视频流内容审核作业.....	89
9.3.7 文档审核.....	89
9.3.7.1 创建文档内容审核作业.....	89
9.3.7.2 查询文档审核作业结果.....	90
10 使用 SDK (PHP)	92
10.1 PHP 开发环境配置.....	92
10.2 PHP 环境 SDK 导入.....	93
10.3 SDK 调用示例.....	93
10.3.1 文本审核.....	93
10.3.1.1 文本内容审核 (V3)	93
10.3.1.2 文本内容审核 (V2)	95
10.3.2 图像审核.....	96
10.3.2.1 图像内容审核 (V3)	96
10.3.2.2 图像内容审核 (V2)	97
10.3.2.3 图像内容审核 (批量) (V2)	98
10.3.2.4 内容审核-图像 (异步批量) (V2)	99
10.3.2.4.1 提交任务.....	99
10.3.2.4.2 查询任务.....	100
10.3.3 音频审核.....	101
10.3.3.1 创建音频内容审核作业.....	101
10.3.3.2 查询音频内容审核作业.....	102
10.3.4 视频审核.....	103
10.3.4.1 创建视频内容审核作业.....	103
10.3.4.2 查询视频内容审核作业.....	104
10.3.5 音频流审核.....	105
10.3.5.1 创建音频流内容审核作业.....	106
10.3.5.2 关闭音频流内容审核作业.....	107
10.3.6 视频流审核.....	107
10.3.6.1 创建视频流内容审核作业.....	107
10.3.6.2 关闭视频流内容审核作业.....	109
10.3.7 文档审核.....	109
10.3.7.1 创建文档内容审核作业.....	110
10.3.7.2 查询文档审核作业结果.....	111
11 使用 SDK (Node.js)	112
11.1 Node.js 开发环境配置.....	112
11.2 Node.js 环境 SDK 导入.....	113
11.3 SDK 调用示例.....	113
11.3.1 文本审核.....	113
11.3.1.1 文本内容审核 (V3)	113
11.3.1.2 文本内容审核 (V2)	114

11.3.2 图像审核.....	115
11.3.2.1 图像内容审核（V3）.....	115
11.3.2.2 图像内容审核（V2）.....	116
11.3.2.3 图像内容审核（批量）（V2）.....	116
11.3.2.4 内容审核-图像（异步批量）（V2）.....	117
11.3.2.4.1 提交任务.....	117
11.3.2.4.2 查询任务.....	118
11.3.3 音频审核.....	119
11.3.3.1 创建音频内容审核作业.....	119
11.3.3.2 查询音频内容审核作业.....	119
11.3.4 视频审核.....	120
11.3.4.1 创建视频内容审核作业.....	120
11.3.4.2 查询视频内容审核作业.....	121
11.3.5 音频流审核.....	122
11.3.5.1 创建音频流内容审核作业.....	122
11.3.5.2 关闭音频流内容审核作业.....	123
11.3.6 视频流审核.....	123
11.3.6.1 创建视频流内容审核作业.....	123
11.3.6.2 关闭视频流内容审核作业.....	124
11.3.7 文档审核.....	124
11.3.7.1 创建文档内容审核作业.....	125
11.3.7.2 查询文档审核作业结果.....	125

1 内容审核 SDK 简介

内容审核概述

内容审核（Content Moderation），是基于图像、文本、音频、视频、音频流、视频流、文档的检测技术，可自动检测涉黄、广告等内容，对用户上传的图片、文字进行内容审核，帮助客户降低业务违规风险。

内容审核以开放API（Application Programming Interface，应用程序编程接口）的方式提供给用户，用户通过实时访问和调用API获取推理结果，帮助用户自动采集关键数据，打造智能化业务系统，提升业务效率。

SDK 概述

内容审核软件开发工具包（Content Moderation Software Development Kit，简称 Moderation SDK）是对内容审核提供的REST API进行的封装，以简化用户的开发工作。用户通过添加依赖或下载的方式调用API即可实现使用内容审核业务能力的目的。

接口与 API 对应关系

内容审核接口与API对应关系请参见[表1-1](#)。

表 1-1 接口与 API 对应关系表

接口	API
文本内容审核（V3）	POST /v3/{project_id}/moderation/text
文本内容审核（V2）	POST /v2/{project_id}/moderation/text
图像内容审核（V3）	POST /v3/{project_id}/moderation/image
图像内容审核（V2）	POST /v2/{project_id}/moderation/image
图像内容审核（批量）	POST /v2/{project_id}/moderation/image/batch
图像内容审核（异步批量）-任务提交	POST /v2/{project_id}/moderation/image/batch/jobs
图像内容审核（异步批量）-处理结果查询	GET /v2/{project_id}/moderation/image/batch

接口	API
图像内容审核（异步批量）-任务列表查询	GET /v2/{project_id}/moderation/image/batch/jobs
创建音频内容审核作业	POST /v3/{project_id}/moderation/audio/jobs
查询音频内容审核作业	GET /v3/{project_id}/moderation/audio/jobs/{job_id}
创建视频内容审核作业	POST /v3/{project_id}/moderation/video/jobs
查询视频内容审核作业	GET /v3/{project_id}/moderation/video/jobs/{job_id}
创建音频流内容审核作业	POST /v3/{project_id}/moderation/audio-stream/jobs
查询音频流内容审核作业	POST /v3/{project_id}/moderation/audio-stream/jobs/stop/{job_id}
创建视频流内容审核作业	POST /v3/{project_id}/moderation/video-stream/jobs
查询视频流内容审核作业	POST /v3/{project_id}/moderation/video-stream/jobs/stop/{job_id}
创建文档内容审核作业	POST /v3/{project_id}/moderation/document/jobs
查询文档审核作业结果	GET /v3/{project_id}/moderation/document/jobs/{job_id}

2 申请服务

申请内容审核服务的具体操作步骤请参见《内容审核API参考》的“[如何调用API > 申请服务](#)”章节。

3 获取认证信息

步骤1 登录华为云管理控制台。

步骤2 鼠标移动至用户名处，在下拉列表中单击“我的凭证”。

步骤3 选择“访问密钥”页签，单击“新增访问秘钥”按钮。

步骤4 通过邮箱或者手机进行验证，输入对应的验证码。

步骤5 单击“确定”，下载认证账号的AK/SK，AK/SK数据会以本地文件的形式保存，请妥善保管。

----结束

4 准备环境

在使用内容审核SDK时，各语言需要准备的环境 参照[表4-1](#)。

表 4-1 开发环境

开发语言	准备项	说明
JAVA	安装JDK	JAVA环境开发配置，支持 Java JDK 1.8 及其以上版本，推荐通过Maven 安装依赖的方式使用JAVA版本SDK。
PYTHON	安装python	python版本sdk支持python3.3以上。
.NET	安装.NET	.NET Standard 2.0 及其以上版本或C# 4.0 及其以上版本。
GO	安装GO	sdk支持 go 1.14 及其以上版本。
PHP	安装PHP	PHP版本sdk包推荐使用php.7.x版本。
Node.js	安装Node.js	Node.js版本sdk支持Node.js 8.x版本。

5 SDK 列表及指导文档（通用版）

[API Explorer](#)能根据需要动态生成SDK代码功能，降低您使用SDK的难度，推荐使用。

[SDK中心](#)提供服务SDK依赖引入的方式。

表5-1 提供了各个云服务支持的SDK列表，您可以在GitHub仓库查看SDK更新历史、获取安装包以及查看指导文档进行配置。

表 5-1 SDK 列表

编程语言	github地址	指导文档	视频指导
JAVA	huaweicloud-sdk-java-v3	Java SDK使用指导	Java SDK视频指导
PYTHON	huaweicloud-sdk-python-v3	Python SDK使用指导	Python SDK视频指导
.NET	huaweicloud-sdk-net-v3	.Net SDK使用指导	-
GO	huaweicloud-sdk-go-v3	Go SDK使用指导	-
PHP	huaweicloud-sdk-php-v3	PHP SDK使用指导	-
Node.js	huaweicloud-sdk-nodejs-v3	Node.js SDK使用指导	-

6 使用 SDK (Java)

6.1 Java 开发环境配置

本SDK包要求的JDK版本必须高于JDK8版本，以下步骤以win7环境配置JDK8 64位为例，若已经下载JDK并配置好环境请忽略本章节。

步骤1 单击[下载JDK文件](#)。

步骤2 下载完成后按照提示安装，位置自选，比如安装到本地“C:\Program Files\Java\jdk1.8.0_131”。

步骤3 配置Java环境变量：右键“计算机 > 属性 > 高级系统设置 > 环境变量”，进行如下操作。

1. 新建系统变量JAVA_HOME，变量值为实际JDK安装位置。
2. 在Path中添加“%JAVA_HOME%\bin;%JAVA_HOME%\jre\bin”（注意用英文分号分隔）。
3. 新建系统变量CLASSPATH，变量值为“%JAVA_HOME%\lib\dt.jar;%JAVA_HOME%\lib\tools.jar”。

步骤4 打开命令行窗口，输入“java -version”，显示如[图6-1](#)表示配置成功。

图 6-1 Java 版本信息

```
C:\>java -version
java version "1.8.0_131"
Java(TM) SE Runtime Environment (build 1.8.0_131-b11)
Java HotSpot(TM) 64-Bit Server VM (build 25.131-b11, mixed mode)
```

----结束

6.2 SDK 获取和安装

JAVA版本，内容审核服务推荐您通过 Maven 安装依赖的方式使用华为云 Java SDK：

步骤1 首先您需要在您的操作系统中 [下载](#) 并 [安装](#)Maven 。

步骤2 安装配置完成后，输入命令“mvn -v”，显示如下图 Maven版本信息 表示成功。

```
D:\>mvn -v
D:\
Apache Maven 3.3.9 (bb52d8502b132ec0a5a3f4c09453c07478323dc5; 2015-11-11T00:41:47+08:00)
Maven home: D:\maven\apache-maven-3.3.9\bin\..
Java version: 1.8.0_262, vendor: Huawei Technologies Co., Ltd
Java home: D:\develop\jdk_1.8\jre
Default locale: zh_CN, platform encoding: GBK
OS name: "windows 10", version: "10.0", arch: "amd64", family: "dos"
```

步骤3 Maven安装完成后，可根据开发需要直接引入依赖到已有的Maven工程或先用开发工具创建Maven工程。

创建项目以idea开发工具为例(已有Maven工程可跳过此步骤)：

1. 打开IntelliJ IDEA 开发工具。
2. 单击File - New - project...
3. 在New Project弹窗 单击-Maven-单击Next。
4. 输入GroupId和ArtifactId，单击Next。
5. 输入Project name 和 Project location，单击Finish。

步骤4 在Maven 项目的 pom.xml 文件加入相应版本的依赖项即可。

以引入3.0.55版本的内容审核SDK为例。请在[SDK中心](#)获取最新的sdk包版本，替换代码中的版本。

```
<dependency>
  <groupId>com.huaweicloud.sdk</groupId>
  <artifactId>huaweicloud-sdk-moderation</artifactId>
  <version>3.0.55</version>
</dependency>
```

示例代码因需要输出结果更直观，所以新增fastjson的依赖（如接入不涉及此项，可忽略并在示例代码中删除相关内容），依赖如下：

```
<dependency>
  <groupId>com.alibaba</groupId>
  <artifactId>fastjson</artifactId>
  <version>1.2.70</version>
</dependency>
```

----结束

6.3 SDK 调用示例

6.3.1 文本审核

6.3.1.1 文本内容审核（V3）

1. 本章节对文本内容审核AK/SK方式使用SDK进行示例说明。

示例代码中可以使用TextDetectionDataReq类的withText方法配置待检测的文本信息，配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
```



```
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.core.region.Region;
import com.huaweicloud.sdk.moderation.v3.*;
import com.huaweicloud.sdk.moderation.v3.model.*;

public class RunTextModerationSolution {

    public static void main(String[] args) {
        // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量
        // 中密文存放，使用时解密，确保安全
        // 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境
        // 变量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK
        String ak = System.getenv("HUAWEICLOUD_SDK_AK");
        String sk = System.getenv("HUAWEICLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        ModerationClient client = ModerationClient.newBuilder()
            .withCredential(auth)
            .withRegion(ModerationRegion.valueOf("xxx")) //把xxx替换成服务所在的区域，例如北京
            .build();
        RunTextModerationRequest request = new RunTextModerationRequest();
        TextDetectionReq body = new TextDetectionReq();
        TextDetectionDataReq databody = new TextDetectionDataReq();
        databody.withText("test");
        body.withData(databody);
        body.withEventType("comment");
        request.withBody(body);
        try {
            RunTextModerationResponse response = client.runTextModeration(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

- 控制台输出200即表示程序执行成功，[文本内容审核结果](#)输出到控制台。

```
class RunTextModerationResponse {
    requestId: 308b6ad2740e51de73597da9fdc94ee1
    result: class TextDetectionResult {
        suggestion: pass
        label: normal
        details: []
    }
}
```

6.3.1.2 文本内容审核 (V2)

- 本章节对文本内容审核AK/SK方式使用SDK进行示例说明。

示例代码中可以使用TextDetectionItemsReq类的withText方法配置待检测的文本信息，配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
package com.huaweicloud.sdk.test;

import com.alibaba.fastjson.JSON;
```

```
import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.moderation.v2.ModerationClient;
import com.huaweicloud.sdk.moderation.v2.region.ModerationRegion;
import com.huaweicloud.sdk.moderation.v2.model.RunTextModerationResponse;
import com.huaweicloud.sdk.moderation.v2.model.TextDetectionReq;
import com.huaweicloud.sdk.moderation.v2.model.RunTextModerationRequest;
import com.huaweicloud.sdk.moderation.v2.model.TextDetectionItemsReq;

import java.util.List;
import java.util.ArrayList;

public class RunTextModerationSolution {

    public static void main(String[] args) {
        // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量
        // 中密文存放，使用时解密，确保安全
        // 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境
        // 变量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK
        String ak = System.getenv("HUAWEICLOUD_SDK_AK");
        String sk = System.getenv("HUAWEICLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        ModerationClient client = ModerationClient.newBuilder()
            .withCredential(auth)
            .withRegion(ModerationRegion.valueOf("xxx")) //把xxx替换成服务所在的区域，例如北京
            .build();
        RunTextModerationRequest request = new RunTextModerationRequest();
        TextDetectionReq body = new TextDetectionReq();
        List<TextDetectionItemsReq> listbodyItems = new ArrayList<>();
        listbodyItems.add(
            new TextDetectionItemsReq()
                .withText("text") //此处输入待检测文本
                .withType("content")
        );
        List<String> listbodyCategories = new ArrayList<>();
        listbodyCategories.add("ad");
        listbodyCategories.add("abuse");
        listbodyCategories.add("porn");
        listbodyCategories.add("contraband");
        listbodyCategories.add("flood");
        body.withItems(listbodyItems);
        body.withCategories(listbodyCategories);
        request.withBody(body);
        try {
            RunTextModerationResponse response = client.runTextModeration(request);
            System.out.println(response.getHttpStatusCode());
            System.out.println(JSON.toJSONString(response));
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

2. 控制台输出200即表示程序执行成功，[文本内容审核结果](#)输出到控制台。

```
200
{"result":{"suggestion":"block","detail":{"contraband":["亚砷酸钾"],"porn":["luo聊"]}},"httpStatusCode":200}
```

6.3.2 图像审核

6.3.2.1 图像内容审核 (V3)

1. 本章节对图像内容审核AK/SK方式使用SDK进行示例说明。

示例代码中可以使用ImageDetectionReq类的withUrl或withImage方法配置图像信息（image和url参数二选一），示例中以url为例配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.moderation.v3.region.ModerationRegion;
import com.huaweicloud.sdk.moderation.v3.*;
import com.huaweicloud.sdk.moderation.v3.model.*;

import java.util.List;
import java.util.ArrayList;

public class CheckImageModerationSolution {

    public static void main(String[] args) {
        // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量
        // 中密文存放，使用时解密，确保安全
        // 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境
        // 变量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK
        String ak = System.getenv("HUAWEICLOUD_SDK_AK");
        String sk = System.getenv("HUAWEICLOUD_SDK_SK");
        String projectId = "project_id";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        ModerationClient client = ModerationClient.newBuilder()
            .withCredential(auth)
            .withRegion(ModerationRegion.valueOf("xxx")) //把xxx替换成服务所在的区域，例如北京
            .build();
        CheckImageModerationRequest request = new CheckImageModerationRequest();
        ImageDetectionReq body = new ImageDetectionReq();
        List<String> listbodyCategories = new ArrayList<>();
        listbodyCategories.add("porn");
        body.withUrl("http://www.xxxx.xxx");
        body.withCategories(listbodyCategories);
        body.withEventType("head_image");
        request.withBody(body);
        try {
            CheckImageModerationResponse response = client.checkImageModeration(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
        }
    }
}
```

```
        System.out.println(e.getErrorCode());  
        System.out.println(e.getErrorMsg());  
    }  
}
```

2. 控制台输出200即表示程序执行成功，[图像内容审核结果](#)输出到控制台。

```
class CheckImageModerationResponse {  
    requestId: 63b6b19eb9fa1db05d2f87f970472995  
    result: class ImageDetectionResult {  
        suggestion: block  
        category: image_text  
        details: [class ImageDetectionResultDetail {  
            suggestion: block  
            category: image_text  
            confidence: 1.0  
            faceLocation: null  
            qrLocation: class QRLocationDetail {  
                topLeftX: 74  
                topLeftY: 66  
                bottomRightX: 151  
                bottomRightY: 142  
            }  
            qrContent: http://xxxxxxxxxx  
            segments: null  
            label: qr_code  
        }, class ImageDetectionResultDetail {  
            suggestion: block  
            category: image_text  
            confidence: 0.98779297  
            faceLocation: null  
            qrLocation: null  
            qrContent: null  
            segments: null  
            label: ad  
        }, class ImageDetectionResultDetail {  
            suggestion: block  
            category: image_text  
            confidence: 1.0  
            faceLocation: null  
            qrLocation: null  
            qrContent: null  
            segments: null  
            label: qr_code  
        }  
    ]  
    ocrText: xxxxxxxxx  
}
```

6.3.2.2 图像内容审核 (V2)

1. 本章节对图像内容审核AK/SK方式使用SDK进行示例说明。

示例代码中可以使用ImageDetectionReq类的withUrl或withImage方法配置图像信息（image和url参数二选一），示例中以url为例配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
package com.huaweicloud.sdk.test;
```

```
import com.alibaba.fastjson.JSON;
```

```
import com.huaweicloud.sdk.core.auth.ICredential;  
import com.huaweicloud.sdk.core.auth.BasicCredentials;  
import com.huaweicloud.sdk.core.exception.ConnectionException;  
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;  
import com.huaweicloud.sdk.core.exception.ServiceResponseException;  
import com.huaweicloud.sdk.moderation.v2.ModerationClient;  
import com.huaweicloud.sdk.moderation.v2.region.ModerationRegion;  
import com.huaweicloud.sdk.moderation.v2.model.RunImageModerationRequest;
```

```
import com.huaweicloud.sdk.moderation.v2.model.ImageDetectionReq;
import com.huaweicloud.sdk.moderation.v2.model.RunImageModerationResponse;

import java.util.List;
import java.util.ArrayList;

public class RunImageModerationSolution {
    public static void main(String[] args) {
        // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量
        // 中密文存放，使用时解密，确保安全
        // 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境
        // 变量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK
        String ak = System.getenv("HUAWEICLOUD_SDK_AK");
        String sk = System.getenv("HUAWEICLOUD_SDK_SK");
        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);
        ModerationClient client = ModerationClient.newBuilder()
            .withCredential(auth)
            .withRegion(ModerationRegion.valueOf("xxx")) //把xxx替换成服务所在的区域，例如北京四：
            cn-north-4。
            .build();
        RunImageModerationRequest request = new RunImageModerationRequest();
        ImageDetectionReq body = new ImageDetectionReq();
        List<ImageDetectionReq.CategoriesEnum> listbodyCategories = new ArrayList<>();
        listbodyCategories.add(ImageDetectionReq.CategoriesEnum.fromValue("terrorism"));
        listbodyCategories.add(ImageDetectionReq.CategoriesEnum.fromValue("porn"));
        body.withCategories(listbodyCategories);
        body.withModerationRule("default");
        body.withUrl("https://XXX.jpg"); //此处替换为公网可以访问的图片地址
        request.withBody(body);
        try {
            RunImageModerationResponse response = client.runImageModeration(request);
            System.out.println(response.getHttpStatusCode());
            System.out.println(JSON.toJSONString(response));
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

- 控制台输出200即表示程序执行成功，**图像内容审核结果**输出到控制台。

```
200
{"httpStatusCode":200,"result":{"categorySuggestions":{"terrorism":"pass","porn":"pass"},"detail":{"porn":[{"confidence":0.9883,"label":"normal"},{"confidence":0.0039,"label":"porn"}, {"confidence":0.0078,"label":"sexy"}],"terrorism":[{"confidence":2.0E-4,"label":"terrorist"}, {"confidence":2.0E-4,"label":"fascist"}, {"confidence":1.0E-4,"label":"cult"}, {"confidence":2.0E-4, {"confidence":1.0E-4}, {"confidence":1.0E-4,"label":"kidnap"}, {"confidence":6.0E-4,"label":"riot"}, {"confidence":1.0E-4,"label":"parade"}, {"confidence":2.0E-4,"label":"corpse"}, {"confidence":1.0E-4,"label":"bloody"}, {"confidence":1.0E-4,"label":"knife"}, {"confidence":1.0E-4,"label":"gun"}, {"confidence":2.0E-4,"label":"military_weapon"}, {"confidence":0.0,"label":"special_characters"}, {"confidence":0.9994,"label":"normal"}]"},"suggestion":"pass"}}
```

Process finished with exit code 0

6.3.2.3 图像内容审核（批量）（V2）

- 本章节对图像内容审核AK/SK方式使用SDK进行示例说明。

示例代码中可以使用ImageBatchModerationReq类的withUrls配置待审核的图像url（仅支持url方式），配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
package com.huaweicloud.sdk.test;

import com.alibaba.fastjson.JSON;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.moderation.v2.ModerationClient;
import com.huaweicloud.sdk.moderation.v2.region.ModerationRegion;
import com.huaweicloud.sdk.moderation.v2.model.RunImageBatchModerationRequest;
import com.huaweicloud.sdk.moderation.v2.model.ImageBatchModerationReq;
import com.huaweicloud.sdk.moderation.v2.model.RunImageBatchModerationResponse;

import java.util.List;
import java.util.ArrayList;

public class RunImageBatchModerationSolution {
    public static void main(String[] args) {
        // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量
        // 中密文存放，使用时解密，确保安全
        // 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境
        // 变量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK
        String ak = System.getenv("HUAWEICLOUD_SDK_AK");
        String sk = System.getenv("HUAWEICLOUD_SDK_SK");
        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);
        ModerationClient client = ModerationClient.newBuilder()
            .withCredential(auth)
            .withRegion(ModerationRegion.valueOf("xxx")) //把xxx替换成服务所在的区域，例如北京四：
            cn-north-4。
            .build();
        RunImageBatchModerationRequest request = new RunImageBatchModerationRequest();
        ImageBatchModerationReq body = new ImageBatchModerationReq();
        List<ImageBatchModerationReq.CategoriesEnum> listbodyCategories = new ArrayList<>();
        listbodyCategories.add(ImageBatchModerationReq.CategoriesEnum.fromValue("terrorism"));
        listbodyCategories.add(ImageBatchModerationReq.CategoriesEnum.fromValue("porn"));
        List<String> listbodyUrls = new ArrayList<>();
        listbodyUrls.add("https://XXX.jpg");//此处替换为公网可以访问的图片地址
        body.withThreshold((double) 0);
        body.withCategories(listbodyCategories);
        body.withUrls(listbodyUrls);
        request.withBody(body);
        try {
            RunImageBatchModerationResponse response = client.runImageBatchModeration(request);
            System.out.println(response.getHttpStatusCode());
            System.out.println(JSON.toJSONString(response));
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

2. 控制台输出200即表示程序执行成功，[图像内容审核批量](#)提交的结果输出到控制台。

```
200
{"httpStatusCode":200,"result":[{"categorySuggestions":{"terrorism":"pass","porn":"pass"},"detail":
{"porn":[{"confidence":0.9883,"label":"normal"},{"confidence":0.0039,"label":"porn"},
{"confidence":0.0078,"label":"sexy"}],"terrorism":[{"confidence":2.0E-4,"label":"terrorist"}]}
```

```
{"confidence":2.0E-4,"label":"fascist"},{"confidence":1.0E-4,"label":"cult"},{"confidence":2.0E-4},
{"confidence":1.0E-4,"label":"kidnap"},{"confidence":6.0E-4,"label":"riot"},
{"confidence":1.0E-4,"label":"parade"},{"confidence":2.0E-4,"label":"corpse"},
{"confidence":1.0E-4,"label":"bloody"},{"confidence":1.0E-4,"label":"knife"},
{"confidence":1.0E-4,"label":"gun"},{"confidence":2.0E-4,"label":"military_weapon"},
{"confidence":0.0,"label":"special_characters"},
{"confidence":0.9994,"label":"normal"}], "suggestion":"pass", "url":"https://sdk-obs-source-save.obs.cn-
north-4.myhuaweicloud.com/terrorism.jpg"}}
```

6.3.2.4 图像内容审核（异步批量）（V2）

6.3.2.4.1 提交任务

1. 本章节对图像内容审核AK/SK方式使用SDK进行示例说明。

示例代码中可以使用RunTaskSubmitRequest 类的withUrls配置待审核的图像url（仅支持url方式），配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
package com.huaweicloud.sdk.test;

import com.alibaba.fastjson.JSON;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.moderation.v2.ModerationClient;
import com.huaweicloud.sdk.moderation.v2.region.ModerationRegion;
import com.huaweicloud.sdk.moderation.v2.model.RunTaskSubmitRequest;
import com.huaweicloud.sdk.moderation.v2.model.TaskSubmitReq;
import com.huaweicloud.sdk.moderation.v2.model.RunTaskSubmitResponse;

import java.util.List;
import java.util.ArrayList;

public class RunTaskSubmitSolution {
    public static void main(String[] args) {
        // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量
        // 中密文存放，使用时解密，确保安全
        // 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境
        // 变量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK
        String ak = System.getenv("HUAWEICLOUD_SDK_AK");
        String sk = System.getenv("HUAWEICLOUD_SDK_SK");
        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);
        ModerationClient client = ModerationClient.newBuilder()
            .withCredential(auth)
            .withRegion(ModerationRegion.valueOf("xxx")) //把xxx替换成服务所在的区域，例如北京四：
            cn-north-4。
            .build();
        RunTaskSubmitRequest request = new RunTaskSubmitRequest();
        TaskSubmitReq body = new TaskSubmitReq();
        List<TaskSubmitReq.CategoriesEnum> listbodyCategories = new ArrayList<>();
        listbodyCategories.add(TaskSubmitReq.CategoriesEnum.fromValue("porn"));
        listbodyCategories.add(TaskSubmitReq.CategoriesEnum.fromValue("terrorism"));
        List<String> listbodyUrls = new ArrayList<>();
        listbodyUrls.add("https://XXX.jpg"); //此处替换为公网可以访问的图片地址
        body.withCategories(listbodyCategories);
        body.withUrls(listbodyUrls);
        request.withBody(body);
        try {
            RunTaskSubmitResponse response = client.runTaskSubmit(request);
            System.out.println(response.getHttpStatusCode());
            System.out.println(JSON.toJSONString(response));
        }
    }
}
```

```
    } catch (ConnectionException e) {
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
```

2. 控制台输出200即表示程序执行成功，异步批量[任务提交](#)的结果输出到控制台。

```
200
{"httpStatusCode":200,"result":{"jobId":"7f6a5881-7774-40cf-903b-4548914b55a3"}}
```

```
Process finished with exit code 0
```

6.3.2.4.2 查询任务

1. 本章节对图像内容审核异步批量-查询任务AK/SK方式使用SDK进行示例说明。

示例代码中需要配置使用RunCheckResultRequest类的withJobId配置审核任务的id，配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
package com.huaweicloud.sdk.test;
```

```
import com.alibaba.fastjson.JSON;
```

```
import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.moderation.v2.ModerationClient;
import com.huaweicloud.sdk.moderation.v2.region.ModerationRegion;
import com.huaweicloud.sdk.moderation.v2.model.RunCheckResultRequest;
import com.huaweicloud.sdk.moderation.v2.model.RunCheckResultResponse;
```

```
public class RunCheckResultSolution {
```

```
    public static void main(String[] args) {
        // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中密文存放，使用时解密，确保安全
```

```
        // 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK
```

```
        String ak = System.getenv("HUAWEICLOUD_SDK_AK");
        String sk = System.getenv("HUAWEICLOUD_SDK_SK");
        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);
```

```
        ModerationClient client = ModerationClient.newBuilder()
            .withCredential(auth)
            .withRegion(ModerationRegion.valueOf("xxx")) //把xxx替换成服务所在的区域，例如北京四：cn-north-4。
            .build();
```

```
        RunCheckResultRequest request = new RunCheckResultRequest();
        request.withJobId("7f6a5881-7774-40cf-903b-4548914b55a3");
        try {
            RunCheckResultResponse response = client.runCheckResult(request);
            System.out.println(response.getHttpStatusCode());
            System.out.println(JSON.toJSONString(response));
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        }
```



```
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
```

2. 控制台输出200即表示程序执行成功，异步批量[查询任务](#)的结果输出到控制台。

```
200
{"httpStatusCode":200,"result":{"createTime":"2021-08-13T03:46:30Z","items":[{"categorySuggestions":
{
"terrorism":"pass","porn":"pass"},"detail":{"porn":[{"confidence":0.9883,"label":"normal"},
{"confidence":0.0039,"label":"porn"},{"confidence":0.0078,"label":"sexy"}],"terrorism":
[{"confidence":2.0E-4,"label":"terrorist"},{"confidence":2.0E-4,"label":"fascist"},
{"confidence":1.0E-4,"label":"cult"},{"confidence":2.0E-4},{"confidence":1.0E-4},
{"confidence":1.0E-4,"label":"kidnap"},{"confidence":6.0E-4,"label":"riot"},
{"confidence":1.0E-4,"label":"parade"},{"confidence":2.0E-4,"label":"corpse"},
{"confidence":1.0E-4,"label":"bloody"},{"confidence":1.0E-4,"label":"knife"},
{"confidence":1.0E-4,"label":"gun"},{"confidence":2.0E-4,"label":"military_weapon"},
{"confidence":0.0,"label":"special_characters"},
{"confidence":0.9994,"label":"normal"}]},"suggestion":"pass","url":"https://sdk-obs-source-save.obs.cn-
north-4.myhuaweicloud.com/
terrorism.jpg"},"jobId":"7f6a5881-7774-40cf-903b-4548914b55a3","status":"finish","updateTime":"202
1-08-13T03:46:30Z"}]}
Process finished with exit code 0
```

6.3.3 音频审核

6.3.3.1 创建音频内容审核作业

1. 本章节对创建音频审核AK/SK方式使用SDK进行示例说明。

示例代码中可以使用RunCreateAudioModerationJobRequest类的withUrls配置待审核的音频url（仅支持url方式），配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.moderation.v3.region.ModerationRegion;
import com.huaweicloud.sdk.moderation.v3.*;
import com.huaweicloud.sdk.moderation.v3.model.*;

import java.util.List;
import java.util.ArrayList;

public class RunCreateAudioModerationJobSolution {

    public static void main(String[] args) {
        // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量
        // 中密文存放，使用时解密，确保安全
        // 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境
        // 变量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK
        String ak = System.getenv("HUAWEICLOUD_SDK_AK");
        String sk = System.getenv("HUAWEICLOUD_SDK_SK");
        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        ModerationClient client = ModerationClient.newBuilder()
```

```
        .withCredential(auth)
        .withRegion(ModerationRegion.valueOf("xxx")) //把xxx替换成服务所在的区域，例如北京
四： cn-north-4。
        .build();
    RunCreateAudioModerationJobRequest request = new RunCreateAudioModerationJobRequest();
    AudioCreateRequest body = new AudioCreateRequest();
    List<AudioCreateRequest.CategoriesEnum> listbodyCategories = new ArrayList<>();
    listbodyCategories.add(AudioCreateRequest.CategoriesEnum.fromValue("porn"));
    AudioInputBody databody = new AudioInputBody();
    databody.withUrl("https://****.wav");
    body.withCategories(listbodyCategories);
    body.withEventType(AudioCreateRequest.EventTypeEnum.fromValue("default"));
    body.withData(databody);
    request.withBody(body);
    try {
        RunCreateAudioModerationJobResponse response =
client.runCreateAudioModerationJob(request);
        System.out.println(response.toString());
    } catch (ConnectionException e) {
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
    }
}
```

- 控制台输出创建音频内容审核作业提交的结果，[创建音频内容审核](#)结果输出到控制台。

```
class RunCreateAudioModerationJobResponse {
    requestId: 0f0d8ed4d85ba28d7461fdb7415a5819
    jobId: 01661702400_cd06d61c16a14565995fcbfb04877bd3
}
```

6.3.3.2 查询音频内容审核作业

- 本章节对查询音频审核AK/SK方式使用SDK进行示例说明。

示例代码中需要配置使用RunQueryAudioModerationJobRequest类的withJobId配置审核任务的id，配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
package com.huaweicloud.sdk.test;
```

```
import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.moderation.v3.region.ModerationRegion;
import com.huaweicloud.sdk.moderation.v3.*;
import com.huaweicloud.sdk.moderation.v3.model.*;
```

```
public class RunQueryAudioModerationJobSolution {
```

```
    public static void main(String[] args) {
        // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量
        中密文存放，使用时解密，确保安全
        // 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境
        变量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK
        String ak = System.getenv("HUAWEICLOUD_SDK_AK");
        String sk = System.getenv("HUAWEICLOUD_SDK_SK");
        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);
```

```
ModerationClient client = ModerationClient.newBuilder()
    .withCredential(auth)
    .withRegion(ModerationRegion.valueOf("xxx")) //把xxx替换成服务所在的区域，例如北京
四：cn-north-4。
    .build();
RunQueryAudioModerationJobRequest request = new RunQueryAudioModerationJobRequest();
request.withJobId("xxx");
try {
    RunQueryAudioModerationJobResponse response =
client.runQueryAudioModerationJob(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

2. 控制台输出查询音频内容审核作业的结果，[查询音频内容审核结果输出到控制台](#)。

```
class RunQueryAudioModerationJobResponse {
    jobId: 01661702400_cd06d61c16a14565995fcbfb04877bd3
    status: running
    result: null
    requestParams: class AudioModerationResultRequestParams {
        eventType: default
        data: class AudioModerationResultRequestParamsData {
            url: https://****.wav
        }
        callback: null
        categories: [porn]
    }
    createTime: 2022-08-29T11:57:14.014Z
    updateTime: 2022-08-29T11:57:14.014Z
    requestId: 57ea754a1fdec2d13e4e8ffbf6a458d
}
```

6.3.4 视频审核

6.3.4.1 创建视频内容审核作业

1. 本章节对创建视频审核AK/SK方式使用SDK进行示例说明。

示例代码中可以使用RunCreateVideoModerationJobRequest类的withUrls配置待审核的视频url（仅支持url方式），配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.moderation.v3.region.ModerationRegion;
import com.huaweicloud.sdk.moderation.v3.*;
import com.huaweicloud.sdk.moderation.v3.model.*;

import java.util.List;
import java.util.ArrayList;
```

```
public class RunCreateVideoModerationJobSolution {

    public static void main(String[] args) {
        // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量
        // 中密文存放，使用时解密，确保安全
        // 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境
        // 变量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK
        String ak = System.getenv("HUAWEICLOUD_SDK_AK");
        String sk = System.getenv("HUAWEICLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        ModerationClient client = ModerationClient.newBuilder()
            .withCredential(auth)
            .withRegion(ModerationRegion.valueOf("xxx")) //把xxx替换成服务所在的区域，例如北京
            .build();
        RunCreateVideoModerationJobRequest request = new RunCreateVideoModerationJobRequest();
        VideoCreateRequest body = new VideoCreateRequest();
        List<VideoCreateRequest.AudioCategoriesEnum> listbodyAudioCategories = new ArrayList<>();
        listbodyAudioCategories.add(VideoCreateRequest.AudioCategoriesEnum.fromValue("porn"));
        List<VideoCreateRequest.ImageCategoriesEnum> listbodyImageCategories = new ArrayList<>();
        listbodyImageCategories.add(VideoCreateRequest.ImageCategoriesEnum.fromValue("porn"));
        VideoCreateRequestData databody = new VideoCreateRequestData();
        databody.withUrl("http://xxx.xx.xxxx")
            .withFrameInterval(5.5f);
        body.withCallback("http://xxx.xx.xxxx");
        body.withAudioCategories(listbodyAudioCategories);
        body.withImageCategories(listbodyImageCategories);
        body.withEventType(VideoCreateRequest.EventTypeEnum.fromValue("default"));
        body.withData(databody);
        request.withBody(body);
        try {
            RunCreateVideoModerationJobResponse response =
            client.runCreateVideoModerationJob(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

- 控制台输出创建视频内容审核作业提交的结果，[创建视频内容审核](#)结果输出到控制台。

```
class RunCreateVideoModerationJobResponse {
    requestId: 5e353de1badd90049a2ac3d2e3ee925e
    jobId: 01663171200_48ce5a21498e47af8f2a49773c3d0fce
}
```

6.3.4.2 查询视频内容审核作业

- 本章节对查询视频审核AK/SK方式使用SDK进行示例说明。

示例代码中需要配置使用RunQueryVideoModerationJobRequest类的withJobId配置审核任务的id，配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
package com.huaweicloud.sdk.test;
```

```
import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.moderation.v3.region.ModerationRegion;
import com.huaweicloud.sdk.moderation.v3.*;
import com.huaweicloud.sdk.moderation.v3.model.*;

public class RunQueryVideoModerationJobSolution {

    public static void main(String[] args) {
        // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量
        // 中密文存放，使用时解密，确保安全
        // 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境
        // 变量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK
        String ak = System.getenv("HUAWEICLOUD_SDK_AK");
        String sk = System.getenv("HUAWEICLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        ModerationClient client = ModerationClient.newBuilder()
            .withCredential(auth)
            .withRegion(ModerationRegion.valueOf("xxx")) //把xxx替换成服务所在的区域，例如北京
            .build();
        RunQueryVideoModerationJobRequest request = new RunQueryVideoModerationJobRequest();
        request.withJobId("jobId");
        try {
            RunQueryVideoModerationJobResponse response =
                client.runQueryVideoModerationJob(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

2. 控制台输出查询视频内容审核作业的结果，[查询视频内容审核结果输出到控制台](#)。

6.3.5 音频流审核

6.3.5.1 创建音频流内容审核作业

1. 本章节对创建音频流审核AK/SK方式使用SDK进行示例说明。

示例代码中可以使用AudioStreamCreateRequestData类的withUrls配置待审核的音频流url，配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
```

```
import com.huaweicloud.sdk.core.region.Region;
import com.huaweicloud.sdk.moderation.v3.*;
import com.huaweicloud.sdk.moderation.v3.model.*;

import java.util.List;
import java.util.ArrayList;

public class RunCreateAudioStreamModerationJobSolution {

public static void main(String[] args) {
    // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量
    // 中密文存放，使用时解密，确保安全
    // 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境
    // 变量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK
    String ak = System.getenv("HUAWEICLOUD_SDK_AK");
    String sk = System.getenv("HUAWEICLOUD_SDK_SK");

    ICredential auth = new BasicCredentials()
        .withAk(ak)
        .withSk(sk);

    ModerationClient client = ModerationClient.newBuilder()
        .withCredential(auth)
        .withRegion(ModerationRegion.valueOf("xxx")) //把xxx替换成服务所在的区域，例如北京
        .build();
    RunCreateAudioStreamModerationJobRequest request = new
    RunCreateAudioStreamModerationJobRequest();
    AudioStreamCreateRequest body = new AudioStreamCreateRequest();
    List<String> listbodyCategories = new ArrayList<>();
    listbodyCategories.add("ad");
    AudioStreamCreateRequestData databody = new AudioStreamCreateRequestData();
    databody.withUrl("http://xxx");
    body.withCallback("http://xxxxxx");
    body.withCategories(listbodyCategories);
    body.withEventType("default");
    body.withData(databody);
    request.withBody(body);
    try {
        RunCreateAudioStreamModerationJobResponse response =
        client.runCreateAudioStreamModerationJob(request);
        System.out.println(response.toString());
    } catch (ConnectionException e) {
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
}
```

- 控制台输出创建音频流内容审核作业提交的结果。

```
class RunCreateAudioStreamModerationJobResponse {
    requestId: c40f626ee8d85cad5bc80dc00c42a4c6
    jobId: 11690387200_c40f626ee8d85cad5bc80dc00c42a4c6
}
```

6.3.5.2 关闭音频流内容审核作业

- 本章节对关闭音频流审核AK/SK方式使用SDK进行示例说明。
示例代码中需要配置使用RunCloseAudioStreamModerationJobRequest类的withJobId配置审核任务的id，配置完成后运行即可。
服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.core.region.Region;
import com.huaweicloud.sdk.moderation.v3.*;
import com.huaweicloud.sdk.moderation.v3.model.*;

public class RunCloseAudioStreamModerationJobSolution {

    public static void main(String[] args) {
        // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量
        // 中密文存放，使用时解密，确保安全
        // 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境
        // 变量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK
        String ak = System.getenv("HUAWEICLOUD_SDK_AK");
        String sk = System.getenv("HUAWEICLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        ModerationClient client = ModerationClient.newBuilder()
            .withCredential(auth)
            .withRegion(ModerationRegion.valueOf("xxx")) //把xxx替换成服务所在的区域，例如北京
            .build();
        RunCloseAudioStreamModerationJobRequest request = new
        RunCloseAudioStreamModerationJobRequest();
        request.withJobId("xxxxxx");
        try {
            RunCloseAudioStreamModerationJobResponse response =
            client.runCloseAudioStreamModerationJob(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}

2. 控制台输出关闭音频流内容审核作业的结果。
class RunCloseAudioStreamModerationJobResponse {
    result: success
}
```

6.3.6 视频流审核

6.3.6.1 创建视频流内容审核作业

1. 本章节对创建视频流审核AK/SK方式使用SDK进行示例说明。

示例代码中可以使用VideoStreamCreateRequestData类的withUrls配置待审核的视频流url（仅支持url方式），配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
package com.huaweicloud.sdk.test;
```

```
import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.core.region.Region;
import com.huaweicloud.sdk.moderation.v3.*;
import com.huaweicloud.sdk.moderation.v3.model.*;

import java.util.List;
import java.util.ArrayList;

public class RunCreateVideoStreamModerationJobSolution {

    public static void main(String[] args) {
        // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量
        // 中密文存放，使用时解密，确保安全
        // 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境
        // 变量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK
        String ak = System.getenv("HUAWEICLOUD_SDK_AK");
        String sk = System.getenv("HUAWEICLOUD_SDK_SK");
        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        ModerationClient client = ModerationClient.newBuilder()
            .withCredential(auth)
            .withRegion(ModerationRegion.valueOf("xxx")) //把xxx替换成服务所在的区域，例如北京
            .build();
        RunCreateVideoStreamModerationJobRequest request = new
        RunCreateVideoStreamModerationJobRequest();
        VideoStreamCreateRequest body = new VideoStreamCreateRequest();
        List<VideoStreamCreateRequest.ImageCategoriesEnum> listbodyImageCategories = new
        ArrayList<>();

        listbodyImageCategories.add(VideoStreamCreateRequest.ImageCategoriesEnum.fromValue("politics"));
        VideoStreamCreateRequestData databody = new VideoStreamCreateRequestData();
        databody.withUrl("http://xxx");
        body.withCallback("http://xxxxxx");
        body.withImageCategories(listbodyImageCategories);
        body.withEventType(VideoStreamCreateRequest.EventTypeEnum.fromValue("default"));
        body.withData(databody);
        request.withBody(body);
        try {
            RunCreateVideoStreamModerationJobResponse response =
            client.runCreateVideoStreamModerationJob(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

2. 控制台输出创建视频流内容审核作业提交的结果。

```
class RunCreateVideoStreamModerationJobResponse {
    requestId: 5b1dfe438c6bed5af1147fd2ebd0b958
    jobId: 01690387200_5b1dfe438c6bed5af1147fd2ebd0b958
}
```


6.3.6.2 关闭视频流内容审核作业

1. 本章节对关闭视频流审核AK/SK方式使用SDK进行示例说明。

示例代码中需要配置使用RunCloseVideoStreamModerationJobRequest类的withJobId配置审核任务的id，配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.core.region.Region;
import com.huaweicloud.sdk.moderation.v3.*;
import com.huaweicloud.sdk.moderation.v3.model.*;

public class RunCloseVideoStreamModerationJobSolution {

    public static void main(String[] args) {
        // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量
        // 中密文存放，使用时解密，确保安全
        // 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境
        // 变量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK
        String ak = System.getenv("HUAWEICLOUD_SDK_AK");
        String sk = System.getenv("HUAWEICLOUD_SDK_SK");
        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        ModerationClient client = ModerationClient.newBuilder()
            .withCredential(auth)
            .withRegion(ModerationRegion.valueOf("xxx")) //把xxx替换成服务所在的区域，例如北京
            .build();
        RunCloseVideoStreamModerationJobRequest request = new
        RunCloseVideoStreamModerationJobRequest();
        request.withJobId("xxxxx");
        try {
            RunCloseVideoStreamModerationJobResponse response =
            client.runCloseVideoStreamModerationJob(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

2. 控制台输出关闭视频流内容审核作业的结果。

```
class RunCloseVideoStreamModerationJobResponse {
    result: success
}
```

6.3.7 文档审核

6.3.7.1 创建文档内容审核作业

1. 本章节对创建文档内容审核作业AK/SK方式使用SDK进行示例说明。

示例代码中可以使用DocumentCreateRequestData类的withUrl配置待审核的文档url，配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.core.region.Region;
import com.huaweicloud.sdk.moderation.v3.*;
import com.huaweicloud.sdk.moderation.v3.model.*;

public class RunCreateDocumentModerationJobSolution {

    public static void main(String[] args) {
        // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量
        // 中密文存放，使用时解密，确保安全
        // 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境
        // 变量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK
        String ak = System.getenv("HUAWEICLOUD_SDK_AK");
        String sk = System.getenv("HUAWEICLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        ModerationClient client = ModerationClient.newBuilder()
            .withCredential(auth)
            .withRegion(ModerationRegion.valueOf("xxx")) //把xxx替换成服务所在的区域，例如北京
            // 四： cn-north-4。
            .build();
        RunCreateDocumentModerationJobRequest request = new
        RunCreateDocumentModerationJobRequest();
        DocumentCreateRequest body = new DocumentCreateRequest();
        DocumentCreateRequestData databody = new DocumentCreateRequestData();
        databody.withUrl("http://xxx")
            .withFormat("pdf");
        body.withEventType("default");
        body.withData(databody);
        request.withBody(body);
        try {
            RunCreateDocumentModerationJobResponse response =
            client.runCreateDocumentModerationJob(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

2. 控制台输出创建文档内容审核作业提交的结果。

```
class RunCreateDocumentModerationJobResponse {
    requestId: 86ad03c26917901d22e06d2f28ba2c7a
    jobId: 01690387200_1434ba6e1de44666877e748133b1d2e9
}
```

6.3.7.2 查询文档审核作业结果

1. 本章节对查询文档审核作业结果AK/SK方式使用SDK进行示例说明。

示例代码中需要配置使用RunQueryDocumentModerationJobRequest类的withJobId配置审核任务的id，配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.core.region.Region;
import com.huaweicloud.sdk.moderation.v3.*;
import com.huaweicloud.sdk.moderation.v3.model.*;

public class RunQueryDocumentModerationJobSolution {

    public static void main(String[] args) {
        // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量
        // 中密文存放，使用时解密，确保安全
        // 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境
        // 变量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK
        String ak = System.getenv("HUAWEICLOUD_SDK_AK");
        String sk = System.getenv("HUAWEICLOUD_SDK_SK");
        String iamEndpoint = "https://iam.cn-north-7.myhuaweicloud.com";
        String endpoint = "https://moderation.cn-north-7.myhuaweicloud.com";

        ICredential auth = new BasicCredentials()
            .withIamEndpoint(iamEndpoint)
            .withAk(ak)
            .withSk(sk);

        ModerationClient client = ModerationClient.newBuilder()
            .withCredential(auth)
            .withRegion(new Region("cn-north-7", endpoint))
            .build();
        RunQueryDocumentModerationJobRequest request = new
        RunQueryDocumentModerationJobRequest();
        request.withJobId("xxx");
        try {
            RunQueryDocumentModerationJobResponse response =
            client.runQueryDocumentModerationJob(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

2. 控制台输出查询文档审核作业结果。

```
class RunQueryDocumentModerationJobResponse {
  jobId: 01693238400_90b2f15d730f43958a74b93619f3ab78
  status: succeeded
  result: class DocumentQueryResponseResult {
    suggestion: pass
    details: []
  }
  requestParams: class DocumentQueryResponseRequestParams {
```

```
data: class DocumentQueryResponseRequestParamsData {  
    url: http://www.baidu.com  
    format: pdf  
    frameInterval: null  
}  
eventType: default  
imageCategories: null  
textCategories: null  
videoImageCategories: null  
audioCategories: null  
callback: null  
}  
createTime: 2023-08-29T15:08:12.012Z  
updateTime: 2023-08-29T15:08:13.013Z  
requestId: 33ca3825b6525bdc8cd7ee2502028692  
}
```

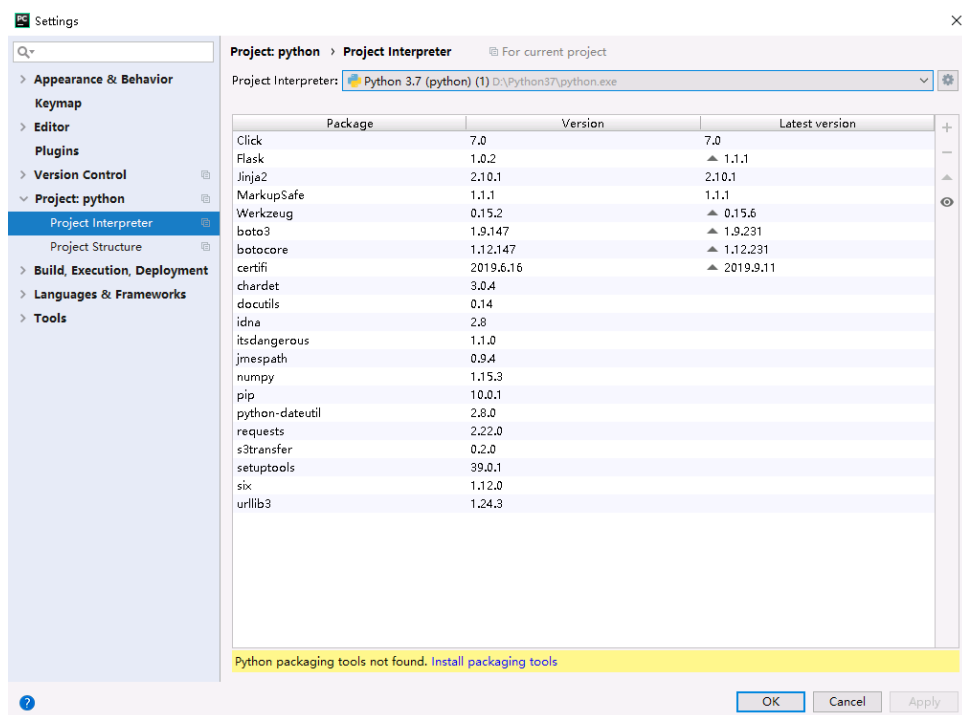
7 使用 SDK (Python)

7.1 Python 开发环境配置

使用内容审核Python版本SDK包，需要您配置Python开发环境。

1. 从[Python官网](#)下载并安装合适的Python版本。请使用Python3.3以上版本，如下以Python3.7 版本为例进行说明。
2. 从[PyCharm官网](#)下载并安装最新版本。
3. 在PyCharm开发工具中配置Python环境，在菜单依次选择“File > Settings > Project Interpreter”。
4. 在页面上方选择您的Python安装路径，如图 [PyCharm配置python环境所示](#)。选择好目标Python之后单击页面下方“Apply”完成配置。

图 7-1 PyCharm 配置 python 环境



7.2 Python 语言 SDK 获取和安装

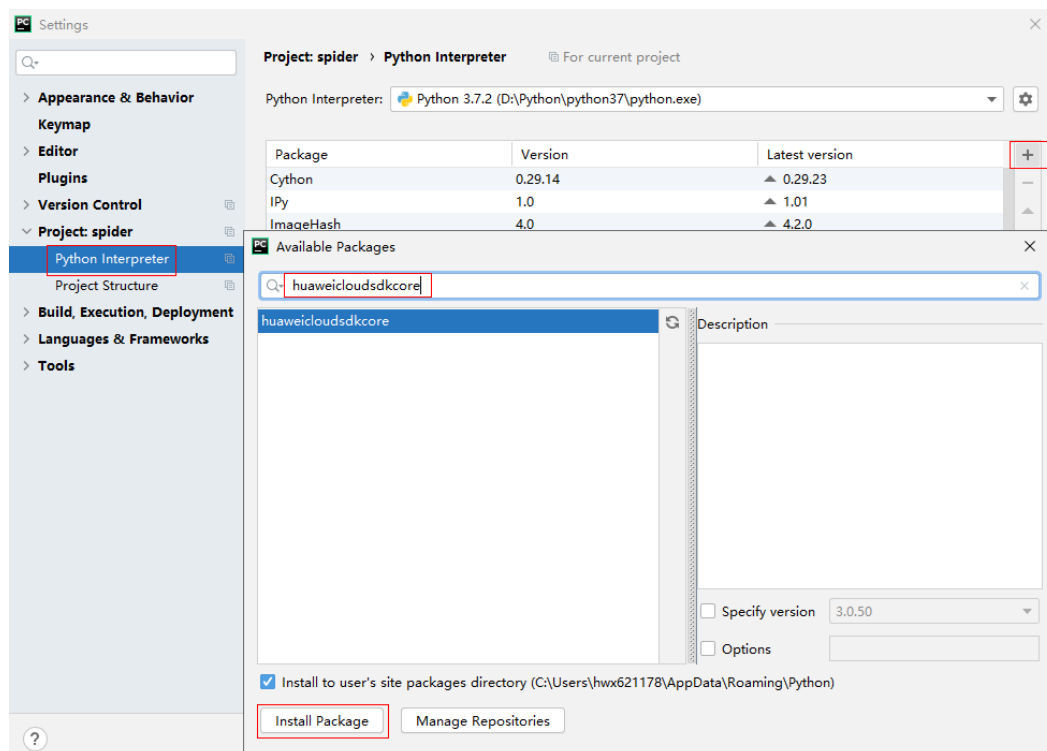
python语言SDK的获取和安装推荐使用pip命令进行安装或使用pycharm进行安装，需要安装huaweicloudsdkcore包以及huaweicloudsdkmoderation，参考方法如下：

```
pip 安装：  
# 安装核心库  
pip install huaweicloudsdkcore  
  
# 安装Moderation服务库  
pip install huaweicloudsdkmoderation
```

使用pycharm安装，步骤如下：

1. 打开pycharm，单击File -> Settings...
2. 单击Python Interpreter -> 单击+。
3. 单击+，分别搜索huaweicloudsdkcore及hwa如下图所示：安装内容审核.NEweicloudsdkmoderation，搜索到包内容单击左下角Install Package完成安装。

图 7-2 pycharm 安装内容审核 python 版本 sdk 包



7.3 SDK 调用示例

7.3.1 文本审核

7.3.1.1 文本内容审核 (v3)

1. 本章节对文本内容审核AK/SK方式使用SDK进行示例说明。

示例代码中修改TextDetectionDataReq的text值配置待检测的文本信息，配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkmoderation.v3.region.moderation_region import ModerationRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkmoderation.v3 import *
import os

if __name__ == "__main__":
    # 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中
    # 密文存放，使用时解密，确保安全；
    # 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
    # HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK。
    ak = os.getenv("HUAWEICLOUD_SDK_AK")
    sk = os.getenv("HUAWEICLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = ModerationClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ModerationRegion.value_of("xxx")) \ //把xxx替换成服务所在的区域，例如北京四：
        cn-north-4。
        .build()

    try:
        request = RunTextModerationRequest()
        databody = TextDetectionDataReq(
            text="test"
        )
        request.body = TextDetectionReq(
            data=databody,
            event_type="comment"
        )
        response = client.run_text_moderation(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

2. 控制台输出200即表示程序执行成功，[文本内容审核结果](#)输出到控制台。

```
{"request_id": "d81702cfd3599e103030c3d52d6f955f", "result": {"suggestion": "pass", "label": "normal", "details": []}}
```

7.3.1.2 文本内容审核 (v2)

1. 本章节对文本内容审核AK/SK方式使用SDK进行示例说明。

示例代码中修改TextDetectionItemsReq的text值配置待检测的文本信息，配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkmoderation.v2.region.moderation_region import ModerationRegion
from huaweicloudsdkmoderation.v2 import *
ModerationClient,RunTextModerationRequest,TextDetectionItemsReq,TextDetectionReq
import os
```

```
if __name__ == "__main__":
    # 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中
    # 密文存放，使用时解密，确保安全；
    # 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
    # HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK。
    ak = os.getenv("HUAWEICLOUD_SDK_AK")
    sk = os.getenv("HUAWEICLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk)

    client = ModerationClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ModerationRegion.value_of("xxx")) \ //把xxx替换成服务所在的区域，例如北京四：
        cn-north-4.
        .build()

    try:
        request = RunTextModerationRequest()
        listTextDetectionItemsReqItemsbody = [
            TextDetectionItemsReq(
                text="text", //此处输入待检测文本
                type="content"
            )
        ]
        listTextDetectionReqCategoriesbody = [
            "porn",
            "ad",
            "abuse",
            "contraband",
            "flood"
        ]
        request.body = TextDetectionReq(
            items=listTextDetectionItemsReqItemsbody,
            categories=listTextDetectionReqCategoriesbody
        )
        response = client.run_text_moderation(request)
        print(response.status_code)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

2. 控制台输出200即表示程序执行成功，[文本内容审核结果](#)输出到控制台。

```
200
{"result": {"suggestion": "block", "detail": {"contraband": ["亚砷酸钾"], "porn": ["luo聊"]}}}
```

7.3.2 图像审核

7.3.2.1 图像内容审核 (V3)

1. 本章节对图像内容审核AK/SK方式使用SDK进行示例说明。

示例代码中可以修改ImageDetectionReq的url或image方法配置图像信息（image和url参数二选一），示例中以url为例配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
# coding: utf-8
```

```
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkmoderation.v3.region.moderation_region import ModerationRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkmoderation.v3 import *
```



```
if __name__ == "__main__":
    # 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中
    # 密文存放，使用时解密，确保安全；
    # 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
    # HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK。
    ak = os.getenv("HUAWEICLOUD_SDK_AK")
    sk = os.getenv("HUAWEICLOUD_SDK_SK")
    projectId = "project_id"

    credentials = BasicCredentials(ak, sk, projectId) \

    client = ModerationClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ModerationRegion.value_of("xxx")) \ //把xxx替换成服务所在的区域，例如北京四：
        cn-north-4。
        .build()

    try:
        request = CheckImageModerationRequest()
        listCategoriesbody = [
            "porn"
        ]
        request.body = ImageDetectionReq(
            url="http://www.xxxx.xxx",
            categories=listCategoriesbody,
            event_type="head_image"
        )
        response = client.check_image_moderation(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

2. 控制台输出200即表示程序执行成功，[图像内容审核结果输出到控制台](#)。

```
{"request_id": "69b424920668cb15e6d25eef879a023b", "result": {"suggestion": "block", "category": "image_text", "details": [{"suggestion": "block", "category": "image_text", "confidence": 1.0, "qr_location": {"top_left_x": 74, "top_left_y": 66, "bottom_right_x": 151, "bottom_right_y": 142}, "qr_content": "http://xxxxx/xxxxx", "label": "qr_code"}, {"suggestion": "block", "category": "image_text", "confidence": 0.98779297, "label": "ad"}, {"suggestion": "block", "category": "image_text", "confidence": 1.0, "label": "qr_code"}], "ocr_text": "xxxxxxxxx"}}
```

7.3.2.2 图像内容审核 (V2)

1. 本章节对图像内容审核AK/SK方式使用SDK进行示例说明。

示例代码中可以修改ImageDetectionReq的url或image方法配置图像信息（image和url参数二选一），示例中以url为例配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#) 。

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkmoderation.v2.region.moderation_region import ModerationRegion
from huaweicloudsdkmoderation.v2 import ModerationClient, RunImageModerationRequest, ImageDetectionReq
import os

if __name__ == "__main__":
    # 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中
    # 密文存放，使用时解密，确保安全；
    # 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
    # HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK。
    ak = os.getenv("HUAWEICLOUD_SDK_AK")
    sk = os.getenv("HUAWEICLOUD_SDK_SK")
    credentials = BasicCredentials(ak, sk)
```

```
client = ModerationClient.new_builder() \
    .with_credentials(credentials) \
    .with_region(ModerationRegion.value_of("xxx")) \ //把xxx替换成服务所在的区域，例如北京四：
cn-north-4。
    .build()

try:
    request = RunImageModerationRequest()
    listImageDetectionReqCategoriesbody = [
        "terrorism",
        "porn",
        "ad"
    ]
    request.body = ImageDetectionReq(
        threshold=0,
        categories=listImageDetectionReqCategoriesbody,
        moderation_rule="default",
        url="https://XXX.jpg" //此处替换为公网可以访问的图片地址
    )
    response = client.run_image_moderation(request)
    print(response.status_code)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

2. 控制台输出200即表示程序执行成功，[图像内容审核结果](#)输出到控制台。

```
200
{"result": {"detail": { "porn": [{"confidence": 0.9883, "label": "normal"}, {"confidence": 0.0039, "label":
"porn"}, {"confidence": 0.0078, "label": "sexy"}], "terrorism": [{"confidence": 0.0002, "label":
"terrorist"}, {"confidence": 0.0002, "label": "fascist"}, {"confidence": 0.0001, "label": "cult"},
{"confidence": 0.0002}, {"confidence": 0.0001, "label": "kidnap"}, {"confidence": 0.0006, "label": "riot"},
{"confidence": 0.0001, "label": "parade"}, {"confidence": 0.0002, "label": "corpse"}, {"confidence":
0.0001, "label": "bloody"}, {"confidence": 0.0001, "label": "knife"}, {"confidence": 0.0001, "label":
"gun"}, {"confidence": 0.0002, "label": "military_weapon"}, {"confidence": 0.0, "label":
"special_characters"}, {"confidence": 0.9994, "label": "normal"}], "ad": [{"confidence": 0.0, "label":
"ad"}, {"confidence": 1.0, "label": "normal"}]}, "suggestion": "pass", "category_suggestions": { "ad":
"pass", "terrorism": "pass", "porn": "pass"}}
Process finished with exit code 0
```

7.3.2.3 图像内容审核（批量）（V2）

1. 本章节对图像内容审核AK/SK方式使用SDK进行示例说明。

示例代码中可以修改ImageBatchModerationReq的urls配置待审核的图像url（仅支持url方式），配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkmoderation.v2.region.moderation_region import ModerationRegion
from huaweicloudsdkmoderation.v2 import
ModerationClient,RunImageBatchModerationRequest,ImageBatchModerationReq
import os

if __name__ == "__main__":
    # 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中
    密文存放，使用时解密，确保安全；
    # 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变
    量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK。
    ak = os.getenv("HUAWEICLOUD_SDK_AK")
    sk = os.getenv("HUAWEICLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk)
```

```
client = ModerationClient.new_builder() \
    .with_credentials(credentials) \
    .with_region(ModerationRegion.value_of("xxx")) \ //把xxx替换成服务所在的区域，例如北京四：
cn-north-4。
    .build()

try:
    request = RunImageBatchModerationRequest()
    listImageBatchModerationReqCategoriesbody = [
        "terrorism",
        "porn",
        "ad"
    ]
    listImageBatchModerationReqUrlsbody = [
        "https://XXX.jpg" //此处替换为公网可以访问的图片地址
    ]
    request.body = ImageBatchModerationReq(
        threshold=0,
        categories=listImageBatchModerationReqCategoriesbody,
        urls=listImageBatchModerationReqUrlsbody
    )
    response = client.run_image_batch_moderation(request)
    print(response.status_code)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

2. 控制台输出200即表示程序执行成功，[图像内容审核批量任务](#)提交的结果输出到控制台。

```
200
{'result': [{'category_suggestions': {'ad': 'pass',
                                     'porn': 'pass',
                                     'terrorism': 'pass'},
            'detail': {
                'ad': [{'confidence': 0.0, 'label': 'ad'},
                      {'confidence': 1.0, 'label': 'normal'}],
                'porn': [{'confidence': 0.9883, 'label': 'normal'},
                         {'confidence': 0.0039, 'label': 'porn'},
                         {'confidence': 0.0078, 'label': 'sexy'}],
                'terrorism': [{'confidence': 0.0002,
                              'label': 'terrorist'},
                              {'confidence': 0.0002,
                               'label': 'fascist'},
                              {'confidence': 0.0001, 'label': 'cult'},
                              {'confidence': 0.0001, 'label': 'kidnap'},
                              {'confidence': 0.0006, 'label': 'riot'},
                              {'confidence': 0.0001, 'label': 'parade'},
                              {'confidence': 0.0002, 'label': 'corpse'},
                              {'confidence': 0.0001, 'label': 'bloody'},
                              {'confidence': 0.0001, 'label': 'knife'},
                              {'confidence': 0.0001, 'label': 'gun'},
                              {'confidence': 0.0002,
                               'label': 'military_weapon'},
                              {'confidence': 0.0,
                               'label': 'special_characters'},
                              {'confidence': 0.9994,
                               'label': 'normal'}]},
            'suggestion': 'pass',
            'url': 'https://sdk-obs-source-save.obs.cn-north-4.myhuaweicloud.com/terrorism.jpg']}]

Process finished with exit code 0
```

7.3.2.4 图像内容审核（异步批量）（V2）

7.3.2.4.1 提交任务

1. 本章节对图像内容审核AK/SK方式使用SDK进行示例说明。

示例代码中可以修改TaskSubmitReq的urls配置待审核的图像url（仅支持url方式），配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkmoderation.v2.region.moderation_region import ModerationRegion
from huaweicloudsdkmoderation.v2 import *
from huaweicloudsdkmoderation.v2 import ModerationClient,RunTaskSubmitRequest,TaskSubmitReq
import os

if __name__ == "__main__":
    # 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中
    # 密文存放，使用时解密，确保安全；
    # 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
    # HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK。
    ak = os.getenv("HUAWEICLOUD_SDK_AK")
    sk = os.getenv("HUAWEICLOUD_SDK_SK")
    credentials = BasicCredentials(ak, sk)

    client = ModerationClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ModerationRegion.value_of("xxx")) \ //把xxx替换成服务所在的区域，例如北京四：
        cn-north-4。
        .build()

    try:
        request = RunTaskSubmitRequest()
        listTaskSubmitReqCategoriesbody = [
            "porn",
            "terrorism"
        ]
        listTaskSubmitReqUrlsbody = [
            "https://XXX.jpg" //此处替换为公网可以访问的图片地址
        ]
        request.body = TaskSubmitReq(
            categories=listTaskSubmitReqCategoriesbody,
            urls=listTaskSubmitReqUrlsbody
        )
        response = client.run_task_submit(request)
        print(response.status_code)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

2. 控制台输出200即表示程序执行成功，异步批量[任务提交](#)的结果输出到控制台。

```
200
{'result': {'job_id': 'xxx'}}

Process finished with exit code 0
```

7.3.2.4.2 查询任务

1. 本章节对图像内容审核异步批量-查询任务AK/SK方式使用SDK进行示例说明。

示例代码中修改RunCheckResultRequest的job_id 配置审核任务的id，配置完成后运行即可。服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
# coding: utf-8
```

```
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkmoderation.v2.region.moderation_region import ModerationRegion
from huaweicloudsdkmoderation.v2 import *
from huaweicloudsdkmoderation.v2 import ModerationClient,RunCheckResultRequest
import os

if __name__ == "__main__":
    # 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中
    # 密文存放，使用时解密，确保安全；
    # 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
    # HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK。
    ak = os.getenv("HUAWEICLOUD_SDK_AK")
    sk = os.getenv("HUAWEICLOUD_SDK_SK")
    credentials = BasicCredentials(ak, sk)

    client = ModerationClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ModerationRegion.value_of("xxx")) \ //把xxx替换成服务所在的区域，例如北京四：
        cn-north-4。
        .build()

    try:
        request = RunCheckResultRequest()
        request.job_id = "xxx" //此处需要输入您的jobID
        response = client.run_check_result(request)
        print(response.status_code)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

2. 控制台输出200即表示程序执行成功，异步批量[查询任务](#)的结果输出到控制台。
200

```
{'result': {'create_time': '2021-08-04T08:30:41Z',
            'items': [{'category_suggestions': {
                        'porn': 'pass',
                        'terrorism': 'pass'},
                    'detail': {'ad': None,
                               'porn': [{'confidence': 0.9883,
                                         'label': 'normal'},
                                       {'confidence': 0.0039,
                                         'label': 'porn'},
                                       {'confidence': 0.0078,
                                         'label': 'sexy'}],
                               'terrorism': [{'confidence': 0.0002,
                                             'label': 'terrorist'},
                                             {'confidence': 0.0002,
                                             'label': 'fascist'},
                                             {'confidence': 0.0001,
                                             'label': 'cult'},
                                             {'confidence': 0.0001,
                                             'label': 'kidnap'},
                                             {'confidence': 0.0006,
                                             'label': 'riot'},
                                             {'confidence': 0.0001,
                                             'label': 'parade'},
                                             {'confidence': 0.0002,
                                             'label': 'corpse'},
                                             {'confidence': 0.0001,
                                             'label': 'bloody'},
                                             {'confidence': 0.0001,
                                             'label': 'knife'},
                                             {'confidence': 0.0001,
                                             'label': 'gun'},
                                             {'confidence': 0.0002,
                                             'label': 'military_weapon'}]
                    }]}
```

```
        {'confidence': 0.0,  
         'label': 'special_characters'},  
        {'confidence': 0.9994,  
         'label': 'normal'}]],  
    'suggestion': 'pass',  
    'url': 'https://sdk-obs-source-save.obs.cn-north-4.myhuaweicloud.com/terrorism.jpg']],  
    'job_id': 'xxx',  
    'status': 'finish',  
    'update_time': '2021-08-04T08:30:41Z'}}
```

Process finished with exit code 0

7.3.3 音频审核

7.3.3.1 创建音频内容审核作业

1. 本章节对创建音频审核AK/SK方式使用SDK进行示例说明。

示例代码中可以使用RunCreateAudioModerationJobRequest类的withUrls配置待审核的音频url（仅支持url方式），配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
# coding: utf-8  
  
from huaweicloudsdkcore.auth.credentials import BasicCredentials  
from huaweicloudsdkmoderation.v3.region.moderation_region import ModerationRegion  
from huaweicloudsdkcore.exceptions import exceptions  
from huaweicloudsdkmoderation.v3 import *  
import os  
  
if __name__ == "__main__":  
    # 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量  
    # 中密文存放，使用时解密，确保安全；  
    # 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变  
    # 量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK。  
  
    ak = __import__('os').getenv("CLOUD_SDK_AK")  
    sk = __import__('os').getenv("CLOUD_SDK_SK")  
  
    credentials = BasicCredentials(ak, sk) \  
  
    client = ModerationClient.new_builder() \  
        .with_credentials(credentials) \  
        .with_region(ModerationRegion.value_of("xxx")) \  
        //把xxx替换成服务所在的区域，例如北京四：cn-north-4。  
        .build()  
  
    try:  
        request = RunCreateAudioModerationJobRequest()  
        listCategoriesbody = [  
            "ad"  
        ]  
        databody = AudioInputBody(  
            url="http://xxx.wav"  
        )  
        request.body = AudioCreateRequest(  
            categories=listCategoriesbody,  
            event_type="default",  
            data=databody  
        )  
        response = client.run_create_audio_moderation_job(request)  
        print(response)  
  
    except exceptions.ClientRequestException as e:  
        print(e.status_code)  
        print(e.request_id)
```

```
print(e.error_code)
print(e.error_msg)
```

2. 控制台输出创建音频内容审核作业的结果，[创建音频内容审核](#)结果输出到控制台。

```
{"request_id": "0b111ce94ec3dXXX4e51a096d996", "job_id": "xxx"}
```

7.3.3.2 查询音频内容审核作业

1. 本章节对查询音频审核AK/SK方式使用SDK进行示例说明。

示例代码中需要配置使用RunQueryAudioModerationJobRequest类的withJobId配置审核任务的id，配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkmoderation.v3.region.moderation_region import ModerationRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkmoderation.v3 import *
import os

if __name__ == "__main__":
    # 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中
    # 密文存放，使用时解密，确保安全；
    # 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
    # HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK。
    ak = os.getenv("HUAWEICLOUD_SDK_AK")
    sk = os.getenv("HUAWEICLOUD_SDK_SK")
    credentials = BasicCredentials(ak, sk) \

    client = ModerationClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ModerationRegion.value_of("xxx")) \  //把xxx替换成服务所在的区域，例如北京四：
        cn-north-4。
        .build()

    try:
        request = RunQueryAudioModerationJobRequest()
        request.job_id = "xxx" //此处需要输入您的jobID
        response = client.run_query_audio_moderation_job(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

2. 控制台输出查询音频内容审核作业提交的结果，[查询音频内容审核](#)结果输出到控制台。

```
{"job_id": "xxxx", "status": "succeeded", "result": {"suggestion": "block", "details":
[{"start_time": 0, "suggestion": "block", "end_time": 10, "label": "porn", "audio_text": "xxxx", "segments":
[{"segment": "xxx"}, {"segment": "xxx"}, {"segment": "xxx"}]},
{"start_time": 30, "suggestion": "block", "end_time": 40, "label": "porn", "audio_text": "xxx", "segments":
[{"segment": "xx"}]}, "audio_text": "xxxxxx"}, "request_params": {"event_type": "default", "data":
{"url": "https://xxxx.wav", "callback": "http://xxx", "categories": ["porn", "ad"]}, "create_time": "2022-06-20
11:05:29", "update_time": "2022-06-20 11:07:08", "request_id": "2419446b1fe14203f64e4018d12db3dd"}
```

7.3.4 视频审核

7.3.4.1 创建视频内容审核作业

1. 本章节对创建视频审核AK/SK方式使用SDK进行示例说明。

示例代码中可以使用RunCreateVideoModerationJobRequest类的withUrls配置待审核的视频url（仅支持url方式），配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkmoderation.v3.region.moderation_region import ModerationRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkmoderation.v3 import *
import os

if __name__ == "__main__":
    # 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中
    # 密文存放，使用时解密，确保安全；
    # 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
    # HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK。
    ak = os.getenv("HUAWEICLOUD_SDK_AK")
    sk = os.getenv("HUAWEICLOUD_SDK_SK")
    credentials = BasicCredentials(ak, sk) \

    client = ModerationClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ModerationRegion.value_of("xxx")) \  //把xxx替换成服务所在的区域，例如北京四：
        cn-north-4.
        .build()

    try:
        request = RunCreateVideoModerationJobRequest()
        listAudioCategoriesbody = [
            "porn"
        ]
        listImageCategoriesbody = [
            "porn"
        ]
        databody = VideoCreateRequestData(
            url="http://xxx.xx.xxxx",
            frame_interval=5.5
        )
        request.body = VideoCreateRequest(
            callback="http://xxx.xx.xxxx",
            audio_categories=listAudioCategoriesbody,
            image_categories=listImageCategoriesbody,
            event_type="default",
            data=databody
        )
        response = client.run_create_video_moderation_job(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

- 控制台输出创建视频内容审核作业的结果，[创建视频内容审核](#)结果输出到控制台。

```
{"request_id": "dbbcb75d1a174bfa899a38be38fdd107", "job_id": "xxx"}
```

7.3.4.2 查询视频内容审核作业

- 本章节对查询视频审核AK/SK方式使用SDK进行示例说明。

示例代码中需要配置使用RunQueryVideoModerationJobRequest类的withJobId配置审核任务的id，配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
# coding: utf-8
```



```
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkmoderation.v3.region.moderation_region import ModerationRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkmoderation.v3 import *
import os

if __name__ == "__main__":
    # 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中
    # 密文存放，使用时解密，确保安全；
    # 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
    # HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK。
    ak = os.getenv("HUAWEICLOUD_SDK_AK")
    sk = os.getenv("HUAWEICLOUD_SDK_SK")
    credentials = BasicCredentials(ak, sk)

    client = ModerationClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ModerationRegion.value_of("xxx")) \ //把xxx替换成服务所在的区域，例如北京四：
        cn-north-4。
        .build()

    try:
        request = RunQueryVideoModerationJobRequest()
        request.job_id = "xxx" //此处需要输入您的jobID
        response = client.run_query_video_moderation_job(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

2. 控制台输出查询视频内容审核作业的结果，[查询视频内容审核结果输出到控制台](#)。

```
{"request_id": "xxxx", "job_id": "xxx", "status": "succeeded", "request_params": {"data":
{"url": "xxxx", "frame_interval": 5, "event_type": "default", "image_categories":
["porn", "terrorism", "image_text"], "audio_categories":
["porn", "ad", "moan", "abuse"], "callback": "xxx", "create_time": "2022-07-30T08:57:11.011Z", "update_time":
"2022-07-30T08:57:14.014Z", "result": {"suggestion": "block", "image_detail":
[{"suggestion": "block", "category": "porn", "ocr_text": "123", "time": 0, "detail":
[{"confidence": 0.676015138626099, "category": "porn", "suggestion": "block", "label": "pornography"},
{"confidence": 0.666015138626099, "category": "porn", "suggestion": "review", "label": "sexy_female"},
{"confidence": 0.666015138626099, "category": "xxx", "suggestion": "block", "label": "leader"},
{"confidence": 0.691423773765564, "category": "xxx", "suggestion": "block", "face_location":
{"top_left_x": 12, "top_left_y": 13, "bottom_right_x": 200, "bottom_right_y": 240, "label": "leader"},
{"confidence": 0.691423773765564, "category": "porn", "suggestion": "block", "face_location":
{"top_left_x": 12, "top_left_y": 13, "bottom_right_x": 100, "bottom_right_y": 2000, "label": "leader"},
{"confidence": 0.891423773765564, "category": "image_text", "suggestion": "block", "label": "qr_code"},
{"confidence": 0.891423773765564, "category": "image_text", "suggestion": "block", "label": "qr_code", "qr_content":
"xxxxxxxxxx", "qr_location":
{"top_left_x": 12, "top_left_y": 13, "bottom_right_x": 100, "bottom_right_y": 2000}],
{"confidence": 0.891423773765564, "category": "image_text", "suggestion": "block", "label": "porn", "segments":
[{"segment": "xxx"}, {"segment": "xx"}]}]}], "audio_detail":
[{"suggestion": "block", "label": "porn", "audio_text": "xxxxx", "end_time": 10, "start_time": 0, "detail":
[{"confidence": 0.999117187582349, "label": "porn", "suggestion": "block", "segments":
[{"segment": "xxx"}]}, {"confidence": 0.885117187582349, "label": "porn", "suggestion": "block"}]}]}
```

7.3.5 音频流审核

7.3.5.1 创建音频流内容审核作业

1. 本章节对创建音频流审核AK/SK方式使用SDK进行示例说明。

示例代码中可以使用AudioStreamCreateRequestData类的withUrls配置待审核的音频流url（仅支持url方式），配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkmoderation.v3.region.moderation_region import ModerationRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkmoderation.v3 import *
import os

if __name__ == "__main__":
    # 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中
    # 密文存放，使用时解密，确保安全；
    # 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
    # HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK。
    ak = os.getenv("HUAWEICLOUD_SDK_AK")
    sk = os.getenv("HUAWEICLOUD_SDK_SK")
    credentials = BasicCredentials(ak, sk)

    client = ModerationClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ModerationRegion.value_of("xxx")) \ //把xxx替换成服务所在的区域，例如北京四：
        cn-north-4。
        .build()

    try:
        request = RunCreateAudioStreamModerationJobRequest()
        listCategoriesbody = [
            "ad"
        ]
        databody = AudioStreamCreateRequestData(
            url="http://xxx"
        )
        request.body = AudioStreamCreateRequest(
            param_callback="http://xxxxxx",
            categories=listCategoriesbody,
            event_type="default",
            data=databody
        )
        response = client.run_create_audio_stream_moderation_job(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

2. 控制台输出创建音频流内容审核作业的结果。

```
{
  "request_id": "303fbd70c9813a2476bad125fa436f00",
  "job_id": "11690387200_ffa1f5bb9a58451b86816gdfc654610e"
}
```

7.3.5.2 关闭音频流内容审核作业

1. 本章节对关闭音频流审核AK/SK方式使用SDK进行示例说明。

示例代码中需要配置使用RunCloseAudioStreamModerationJobRequest类的withJobId配置审核任务的id，配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkmoderation.v3.region.moderation_region import ModerationRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkmoderation.v3 import *
import os

if __name__ == "__main__":
    # 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中
```

```
密文存放，使用时解密，确保安全；
# 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK。
ak = os.getenv("HUAWEICLOUD_SDK_AK")
sk = os.getenv("HUAWEICLOUD_SDK_SK")
credentials = BasicCredentials(ak, sk)

client = ModerationClient.new_builder() \
    .with_credentials(credentials) \
    .with_region(ModerationRegion.value_of("xxx"))\ //把xxx替换成服务所在的区域，例如北京四：cn-north-4。
    .build()

try:
    request = RunCloseAudioStreamModerationJobRequest()
    request.job_id = "xxxxx"
    response = client.run_close_audio_stream_moderation_job(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

2. 控制台输出关闭音频流内容审核作业提交的结果。

```
{
  "result": "success"
}
```

7.3.6 视频流审核

7.3.6.1 创建视频流内容审核作业

1. 本章节对创建视频流审核AK/SK方式使用SDK进行示例说明。

示例代码中可以使用VideoStreamCreateRequestData类的withUrls配置待审核的视频流url（仅支持url方式），配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkmoderation.v3.region.moderation_region import ModerationRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkmoderation.v3 import *
import os

if __name__ == "__main__":
    # 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中密文存放，使用时解密，确保安全；
    # 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK。
    ak = os.getenv("HUAWEICLOUD_SDK_AK")
    sk = os.getenv("HUAWEICLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk)

    client = ModerationClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ModerationRegion.value_of("xxx"))\ //把xxx替换成服务所在的区域，例如北京四：cn-north-4。
        .build()

    try:
        request = RunCreateVideoStreamModerationJobRequest()
        listImageCategoriesbody = [
            "politics"
```

```
]
databody = VideoStreamCreateRequestData(
    url="http://xxx"
)
request.body = VideoStreamCreateRequest(
    param_callback="http://xxxxxx",
    image_categories=listImageCategoriesbody,
    event_type="default",
    data=databody
)
response = client.run_create_video_stream_moderation_job(request)
print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

2. 控制台输出创建视频流内容审核作业的结果。

```
{
  "request_id": "fa520dddb6b54af51c8d1e6d19898656",
  "job_id": "01690387200_09ad14f155bb47e9bf2718ace57c73ed"
}
```

7.3.6.2 关闭视频流内容审核作业

1. 本章节对关闭视频流审核AK/SK方式使用SDK进行示例说明。

示例代码中需要配置使用RunCloseVideoStreamModerationJobRequest类的withJobId配置审核任务的id，配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkmoderation.v3.region.moderation_region import ModerationRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkmoderation.v3 import *
import os

if __name__ == "__main__":
    # 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中
    # 密文存放，使用时解密，确保安全；
    # 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
    # HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK。
    ak = os.getenv("HUAWEICLOUD_SDK_AK")
    sk = os.getenv("HUAWEICLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk)

    client = ModerationClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ModerationRegion.value_of("xxx")) \ //把xxx替换成服务所在的区域，例如北京四：
        cn-north-4.
        .build()
    try:
        request = RunCloseVideoStreamModerationJobRequest()
        request.job_id = "xxxxx"
        response = client.run_close_video_stream_moderation_job(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

2. 控制台输出关闭视频流内容审核作业提交的结果。

```
{
  "result": "success"
}
```

7.3.7 文档审核

7.3.7.1 创建文档内容审核作业

1. 本章节对创建文档内容审核作业AK/SK方式使用SDK进行示例说明。

示例代码中可以使用DocumentCreateRequestData类的withUrl配置待审核的文档url，配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkmoderation.v3.region.moderation_region import ModerationRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkmoderation.v3 import *
import os

if __name__ == "__main__":
    # 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中
    # 密文存放，使用时解密，确保安全；
    # 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
    # HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK。
    ak = os.getenv("HUAWEICLOUD_SDK_AK")
    sk = os.getenv("HUAWEICLOUD_SDK_SK")
    credentials = BasicCredentials(ak, sk)

    client = ModerationClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ModerationRegion.value_of("xxx")) \ //把xxx替换成服务所在的区域，例如北京四：
        cn-north-4。
        .build()

    try:
        request = RunCreateDocumentModerationJobRequest()
        databody = DocumentCreateRequestData(
            url="http://xxx",
            format="pdf"
        )
        request.body = DocumentCreateRequest(
            event_type="default",
            data=databody
        )
        response = client.run_create_document_moderation_job(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

2. 控制台输出创建文档内容审核作业的结果。

```
{"request_id": "c083957284ce1f2c0b910115d8232ec6", "job_id":
"01690387200_e3dfb78d9f20463f98aee25e6895203a"}
```

7.3.7.2 查询文档审核作业结果

1. 本章节对查询文档审核作业结果AK/SK方式使用SDK进行示例说明。

示例代码中需要配置使用RunQueryDocumentModerationJobRequest类的withJobId配置审核任务的id，配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
```

```
from huaweicloudsdkmoderation.v3.region.moderation_region import ModerationRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkmoderation.v3 import *
import os

if __name__ == "__main__":
    # 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中
    # 密文存放，使用时解密，确保安全；
    # 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
    # HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK。
    ak = os.getenv("HUAWEICLOUD_SDK_AK")
    sk = os.getenv("HUAWEICLOUD_SDK_SK")
    credentials = BasicCredentials(ak, sk)

    client = ModerationClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ModerationRegion.value_of("xxx")) \ //把xxx替换成服务所在的区域，例如北京四：
        cn-north-4。
        .build()

    try:
        request = RunQueryDocumentModerationJobRequest()
        request.job_id = "xxxx"
        response = client.run_query_document_moderation_job(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

2. 控制台输出查询文档审核作业结果。

```
{"job_id": "01693238400_cd7c8fd0a30b4c13b1fc9246c542c6", "status": "succeeded", "result":
{"suggestion": "pass", "details": []}, "request_params": {"data": {"url": "http://xxxxx", "format": "pdf"},
"event_type": "default", "create_time": "2023-08-29T16:33:50.050Z", "update_time":
"2023-08-29T16:33:50.050Z", "request_id": "e4eeb972aeb92f4269e8dfbb59ba7e3b"}
```

8 使用 SDK (.NET)

8.1 .NET 开发环境配置

使用内容审核.NET版本SDK包，需要先配置.Net开发环境。

1. [Visual Studio官网](#)，下载最新版Visual Studio。
2. 下载完毕后，单击exe文件，开始安装。
3. 安装过程中选择工作负荷ASP.NET和Web开发，安装完成后启动即可。

图 8-1 选择安装.NET 环境依赖



8.2 .NET 环境 SDK 导入

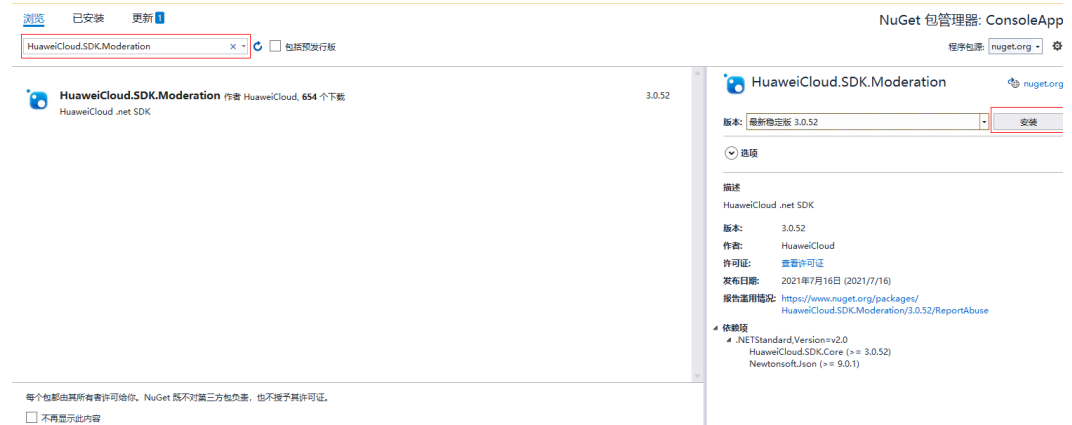
.NET语言SDK的获取和安装可以使用可以通过命令方式安装或使用Visual Studio开发工具安装第三方引用，参考方法如下：

```
//使用 .NET CLI 工具
dotnet add package HuaweiCloud.SDK.Core
dotnet add package HuaweiCloud.SDK.Moderation

//使用 Package Manager
Install-Package HuaweiCloud.SDK.Core
dotnet add package HuaweiCloud.SDK.Moderation
```

使用Visual Studio开发工具安装第三方引用：单击“工具->NuGet包管理器->管理解决方案的NuGet程序包”，选择浏览中搜索并安装HuaweiCloud.SDK.Core及HuaweiCloud.SDK.Moderation,如下图所示：

图 8-2 安装内容审核.NET 版本 sdk 包



8.3 SDK 调用示例

8.3.1 文本审核

8.3.1.1 文本内容审核（V3）

1. 本章节对文本内容审核AK/SK方式使用SDK进行示例说明。

示例代码中可以修改TextDetectionDataReq的Text 配置待检测的文本信息，配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
using System;
using System.Collections.Generic;
using HuaweiCloud.SDK.Core;
using HuaweiCloud.SDK.Core.Auth;
using HuaweiCloud.SDK.Moderation;
using HuaweiCloud.SDK.Moderation.V3;
using HuaweiCloud.SDK.Moderation.V3.Model;

namespace RunTextModerationSolution
{
    class Program
    {
        static void Main(string[] args)
        {
            //认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量
            中密文存放，使用时解密，确保安全
            //本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
            HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK
            string accessKey= Environment.GetEnvironmentVariable("HUAWEICLOUD_SDK_AK",
            EnvironmentVariableTarget.Machine);
            string secretKey= Environment.GetEnvironmentVariable("HUAWEICLOUD_SDK_SK",
            EnvironmentVariableTarget.Machine);

            var config = HttpConfig.GetDefaultConfig();
            config.IgnoreSslVerification = true;
```



```
var client = ModerationClient.NewBuilder()
    .WithCredential(auth)
    .WithRegion(ModerationRegion.ValueOf("xxx")) //把xxx替换成服务所在的区域，例如北京四：
cn-north-4。
    .WithHttpConfig(config)
    .Build();

var req = new RunTextModerationRequest
{
};
TextDetectionDataReq databody = new TextDetectionDataReq()
{
    Text = "test"
};
req.Body = new TextDetectionReq()
{
    Data = databody,
    EventType = "comment"
};

try
{
    var resp = client.RunTextModeration(req);
    var respStatusCode = resp.HttpStatusCode;
    Console.WriteLine(respStatusCode);
}
catch (RequestTimeoutException requestTimeoutException)
{
    Console.WriteLine(requestTimeoutException.ErrorMessage);
}
catch (ServiceResponseException clientRequestException)
{
    Console.WriteLine(clientRequestException.HttpStatusCode);
    Console.WriteLine(clientRequestException.ErrorCode);
    Console.WriteLine(clientRequestException.ErrorMsg);
}
catch (ConnectionException connectionException)
{
    Console.WriteLine(connectionException.ErrorMessage);
}
}
```

2. 控制台输出200即表示程序执行成功，[文本内容审核结果](#)输出到控制台。

```
200
{
  "request_id":"d81702cfd3599e103030c3d52d6f955f",
  "result":{
    "suggestion":"pass",
    "label":"normal",
    "details":[]
  }
}
```

8.3.1.2 文本内容审核（V2）

1. 本章节对文本内容审核AK/SK方式使用SDK进行示例说明。

示例代码中可以修改TextDetectionItemsReq的text配置待检测的文本信息，配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
using System;
using System.Collections.Generic;
using HuaweiCloud.SDK.Core;
using HuaweiCloud.SDK.Core.Auth;
using HuaweiCloud.SDK.Moderation;
using HuaweiCloud.SDK.Moderation.V2;
```

```
using HuaweiCloud.SDK.Moderation.V2.Model;

namespace RunTextModerationSolution
{
    class Program
    {
        static void Main(string[] args)
        {
            //认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量
            //中密文存放，使用时解密，确保安全
            //本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
            //HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK
            string accessKey= Environment.GetEnvironmentVariable("HUAWEICLOUD_SDK_AK",
            EnvironmentVariableTarget.Machine);
            string secretKey= Environment.GetEnvironmentVariable("HUAWEICLOUD_SDK_SK",
            EnvironmentVariableTarget.Machine);

            var config = HttpConfig.GetDefaultConfig();
            config.IgnoreSslVerification = true;
            var auth = new BasicCredentials(ak, sk);

            var client = ModerationClient.NewBuilder()
                .WithCredential(auth)
                .WithRegion(ModerationRegion.ValueOf("xxx")) //把xxx替换成服务所在的区域，例如北京
                .WithHttpConfig(config)
                .Build();

            var req = new RunTextModerationRequest
            {
            };
            List<TextDetectionItemsReq> listTextDetectionItemsReqItems = new
            List<TextDetectionItemsReq>();
            listTextDetectionItemsReqItems.Add(new TextDetectionItemsReq()
            {
                Text = "text", //此处输入待检测文本
                Type = "content"
            });
            List<String> listTextDetectionReqCategories = new List<String>();
            listTextDetectionReqCategories.Add("porn");
            listTextDetectionReqCategories.Add("abuse");
            listTextDetectionReqCategories.Add("contraband");
            listTextDetectionReqCategories.Add("flood");
            req.Body = new TextDetectionReq()
            {
                Items = listTextDetectionItemsReqItems,
                Categories = listTextDetectionReqCategories
            };

            try
            {
                var resp = client.RunTextModeration(req);
                var respStatusCode = resp.HttpStatusCode;
                Console.WriteLine(respStatusCode);
                Console.WriteLine(JsonConvert.DeserializeObject(resp.HttpBody));
            }
            catch (RequestTimeoutException requestTimeoutException)
            {
                Console.WriteLine(requestTimeoutException.ErrorMessage);
            }
            catch (ServiceResponseException clientRequestException)
            {
                Console.WriteLine(clientRequestException.HttpStatusCode);
                Console.WriteLine(clientRequestException.ErrorCode);
                Console.WriteLine(clientRequestException.ErrorMsg);
            }
            catch (ConnectionException connectionException)
            {
                Console.WriteLine(connectionException.ErrorMessage);
            }
        }
    }
}
```

```
}  
}  
}  
}
```

2. 控制台输出200即表示程序执行成功，[文本内容审核结果](#)输出到控制台。

```
200  
{'result': {'detail': {'contraband': ['亚砷酸钾'], 'porn': ['luo聊']},  
  'suggestion': 'block'}}
```

8.3.2 图像审核

8.3.2.1 图像内容审核 (V3)

1. 本章节对图像内容审核AK/SK方式使用SDK进行示例说明。

示例代码中可以修改ImageDetectionReq的Image或Url配置待审核图像信息（image和url参数二选一），示例中以url为例配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
using System;  
using System.Collections.Generic;  
using HuaweiCloud.SDK.Core;  
using HuaweiCloud.SDK.Core.Auth;  
using HuaweiCloud.SDK.Moderation;  
using HuaweiCloud.SDK.Moderation.V3;  
using HuaweiCloud.SDK.Moderation.V3.Model;  
  
namespace CheckImageModerationSolution  
{  
    class Program  
    {  
        static void Main(string[] args)  
        {  
            //认证的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量  
            //中密文存放，使用时解密，确保安全  
            //本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量  
            //HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK  
            string accessKey= Environment.GetEnvironmentVariable("HUAWEICLOUD_SDK_AK",  
                EnvironmentVariableTarget.Machine);  
            string secretKey= Environment.GetEnvironmentVariable("HUAWEICLOUD_SDK_SK",  
                EnvironmentVariableTarget.Machine);  
  
            const string projectId = "project_id";  
            var config = HttpConfig.GetDefaultConfig();  
            config.IgnoreSslVerification = true;  
            var auth = new BasicCredentials(ak, sk, projectId);  
  
            var client = ModerationClient.NewBuilder()  
                .WithCredential(auth)  
                .WithRegion(ModerationRegion.ValueOf("xxx")) //把xxx替换成服务所在的区域，例如北京四：  
                cn-north-4。  
                .WithHttpConfig(config)  
                .Build();  
  
            var req = new CheckImageModerationRequest  
            {  
            };  
            List<string> listCategoriesbody = new List<string>();  
            listCategoriesbody.Add("porn");  
            req.Body = new ImageDetectionReq()  
            {  
                Url = "http://www.xxxx.xxxx",  
                Categories = listCategoriesbody,  
                EventType = "head_image"  
            };  
        }  
    }  
}
```

```
try
{
    var resp = client.CheckImageModeration(req);
    var respStatusCode = resp.HttpStatusCode;
    Console.WriteLine(respStatusCode);
    Console.WriteLine(resp);
}
catch (RequestTimeoutException requestTimeoutException)
{
    Console.WriteLine(requestTimeoutException.ErrorMessage);
}
catch (ServiceResponseException clientRequestException)
{
    Console.WriteLine(clientRequestException.HttpStatusCode);
    Console.WriteLine(clientRequestException.ErrorCode);
    Console.WriteLine(clientRequestException.ErrorMsg);
}
catch (ConnectionException connectionException)
{
    Console.WriteLine(connectionException.ErrorMessage);
}
}
```

2. 控制台输出200即表示程序执行成功，[图像内容审核结果](#)输出到控制台。

```
200
{"request_id": "69b424920668cb15e6d25eef879a023b", "result": {"suggestion": "block", "category":
"image_text", "details": [{"suggestion": "block", "category": "image_text", "confidence": 1.0,
"qr_location": {"top_left_x": 74, "top_left_y": 66, "bottom_right_x": 151, "bottom_right_y": 142},
"qr_content": "http://xxxxxx/xxxxx", "label": "qr_code"}, {"suggestion": "block", "category":
"image_text", "confidence": 0.98779297, "label": "ad"}, {"suggestion": "block", "category":
"image_text", "confidence": 1.0, "label": "qr_code"}], "ocr_text": "xxxxxxxxx"}}
```

8.3.2.2 图像内容审核 (V2)

1. 本章节对图像内容审核AK/SK方式使用SDK进行示例说明。

示例代码中可以修改ImageDetectionReq的Image或Url配置待审核图像信息（image和url参数二选一），示例中以url为例配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
using System;
using System.Collections.Generic;
using HuaweiCloud.SDK.Core;
using HuaweiCloud.SDK.Core.Auth;
using HuaweiCloud.SDK.Moderation;
using HuaweiCloud.SDK.Moderation.V2;
using HuaweiCloud.SDK.Moderation.V2.Model;

namespace RunImageModerationSolution
{
    class Program
    {
        static void Main(string[] args)
        {
            //认证的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量
            中密文存放，使用时解密，确保安全
            //本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变
            量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK
            string accessKey= Environment.GetEnvironmentVariable("HUAWEICLOUD_SDK_AK",
            EnvironmentVariableTarget.Machine);
            string secretKey= Environment.GetEnvironmentVariable("HUAWEICLOUD_SDK_SK",
            EnvironmentVariableTarget.Machine);

            var config = HttpConfig.GetDefaultConfig();
            config.IgnoreSslVerification = true;
            var auth = new BasicCredentials(ak, sk);
```

```
var client = ModerationClient.NewBuilder()
    .WithCredential(auth)
    .WithRegion(ModerationRegion.ValueOf("xxx")) //把xxx替换成服务所在的区域，例如北京
四： cn-north-4。
    .WithHttpConfig(config)
    .Build();

var req = new RunImageModerationRequest
{
};
List<ImageDetectionReq.CategoriesEnum> listImageDetectionReqCategories = new
List<ImageDetectionReq.CategoriesEnum>();
listImageDetectionReqCategories.Add(ImageDetectionReq.CategoriesEnum.FromValue("terrorism"));
listImageDetectionReqCategories.Add(ImageDetectionReq.CategoriesEnum.FromValue("porn"));
listImageDetectionReqCategories.Add(ImageDetectionReq.CategoriesEnum.FromValue("ad"));
req.Body = new ImageDetectionReq()
{
    Threshold = 0,
    Categories = listImageDetectionReqCategories,
    ModerationRule = "default",
    Url = "https://XXX.jpg" //此处替换为公网可以访问的图片地址
};

try
{
    var resp = client.RunImageModeration(req);
    var respStatusCode = resp.HttpStatusCode;
    Console.WriteLine(respStatusCode);
    Console.WriteLine(JsonConvert.DeserializeObject(resp.HttpBody));
}
catch (RequestTimeoutException requestTimeoutException)
{
    Console.WriteLine(requestTimeoutException.ErrorMessage);
}
catch (ServiceResponseException clientRequestException)
{
    Console.WriteLine(clientRequestException.HttpStatusCode);
    Console.WriteLine(clientRequestException.ErrorCode);
    Console.WriteLine(clientRequestException.ErrorMessage);
}
catch (ConnectionException connectionException)
{
    Console.WriteLine(connectionException.ErrorMessage);
}
}
}
```

2. 控制台输出200即表示程序执行成功，[图像内容审核结果](#)输出到控制台。

```
200
{"result": {"category_suggestions": {"ad": "pass",
    "porn": "pass",
    "terrorism": "pass"},
  "detail": {"ad": [{"confidence": 0.0, "label": "ad"},
    {"confidence": 1.0, "label": "normal"}],
  "porn": [{"confidence": 0.9883, "label": "normal"},
    {"confidence": 0.0039, "label": "porn"},
    {"confidence": 0.0078, "label": "sexy"}],
  "terrorism": [{"confidence": 0.0002,
    "label": "terrorist"},
    {"confidence": 0.0002, "label": "fascist"},
    {"confidence": 0.0001, "label": "cult"},
    {"confidence": 0.0002,
    "label": "negative_politics"},
    {"confidence": 0.0001, "label": "kidnap"},
    {"confidence": 0.0006, "label": "riot"},
    {"confidence": 0.0001, "label": "parade"},
    {"confidence": 0.0002, "label": "corpse"},
```

```
{'confidence': 0.0001, 'label': 'bloody'},  
{'confidence': 0.0001, 'label': 'knife'},  
{'confidence': 0.0001, 'label': 'gun'},  
{'confidence': 0.0002,  
  'label': 'military_weapon'},  
{'confidence': 0.0,  
  'label': 'special_characters'},  
{'confidence': 0.9994,  
  'label': 'normal'}}},  
'suggestion': 'pass'}}
```

8.3.2.3 图像内容审核（批量）（V2）

1. 本章节对图像内容审核AK/SK方式使用SDK进行示例说明。

示例代码中可以修改ImageBatchModerationReq 的Urls 配置待审核的图像url（仅支持url方式），配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
using System;  
using System.Collections.Generic;  
using HuaweiCloud.SDK.Core;  
using HuaweiCloud.SDK.Core.Auth;  
using HuaweiCloud.SDK.Moderation;  
using HuaweiCloud.SDK.Moderation.V2;  
using HuaweiCloud.SDK.Moderation.V2.Model;  
  
namespace RunImageBatchModerationSolution  
{  
  class Program  
  {  
    static void Main(string[] args)  
    {  
      //认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量  
      //中密文存放，使用时解密，确保安全  
      //本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量  
      //HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK  
      string accessKey= Environment.GetEnvironmentVariable("HUAWEICLOUD_SDK_AK",  
        EnvironmentVariableTarget.Machine);  
      string secretKey= Environment.GetEnvironmentVariable("HUAWEICLOUD_SDK_SK",  
        EnvironmentVariableTarget.Machine);  
  
      var config = HttpConfig.GetDefaultConfig();  
      config.IgnoreSslVerification = true;  
      var auth = new BasicCredentials(ak, sk);  
  
      var client = ModerationClient.NewBuilder()  
        .WithCredential(auth)  
        .WithRegion(ModerationRegion.ValueOf("xxx")) //把xxx替换成服务所在的区域，例如北京四：cn-north-4。  
        .WithHttpConfig(config)  
        .Build();  
  
      var req = new RunImageBatchModerationRequest  
      {  
      };  
      List<ImageBatchModerationReq.CategoriesEnum> listImageBatchModerationReqCategories =  
      new List<ImageBatchModerationReq.CategoriesEnum>();  
  
      listImageBatchModerationReqCategories.Add(ImageBatchModerationReq.CategoriesEnum.FromValue("terrorism"));  
  
      listImageBatchModerationReqCategories.Add(ImageBatchModerationReq.CategoriesEnum.FromValue("porn"));  
  
      listImageBatchModerationReqCategories.Add(ImageBatchModerationReq.CategoriesEnum.FromValue("ad"));  
      List<String> listImageBatchModerationReqUrls = new List<String>();  
      listImageBatchModerationReqUrls.Add("https://XXX.jpg"); //此处替换为公网可以访问的图片地
```

```
    址
    req.Body = new ImageBatchModerationReq()
    {
        Threshold = 0,
        Categories = listImageBatchModerationReqCategories,
        Urls = listImageBatchModerationReqUrls
    };

    try
    {
        var resp = client.RunImageBatchModeration(req);
        var respStatusCode = resp.HttpStatusCode;
        Console.WriteLine(respStatusCode);
        Console.WriteLine(JsonConvert.DeserializeObject(resp.HttpBody));
    }
    catch (RequestTimeoutException requestTimeoutException)
    {
        Console.WriteLine(requestTimeoutException.ErrorMessage);
    }
    catch (ServiceResponseException clientRequestException)
    {
        Console.WriteLine(clientRequestException.HttpStatusCode);
        Console.WriteLine(clientRequestException.ErrorCode);
        Console.WriteLine(clientRequestException.ErrorMsg);
    }
    catch (ConnectionException connectionException)
    {
        Console.WriteLine(connectionException.ErrorMessage);
    }
    }
}
```

2. 控制台输出200即表示程序执行成功，[图像内容审核批量任务](#)提交的结果输出到控制台。

```
200
{'result': [{'category_suggestions': {'ad': 'pass',
                                     'porn': 'pass',
                                     'terrorism': 'pass'},
            'detail': {'ad': [{'confidence': 0.0, 'label': 'ad'},
                              {'confidence': 1.0, 'label': 'normal'}],
                       'porn': [{'confidence': 0.9883, 'label': 'normal'},
                                  {'confidence': 0.0039, 'label': 'porn'},
                                  {'confidence': 0.0078, 'label': 'sexy'}],
                       'terrorism': [{'confidence': 0.0002,
                                       'label': 'terrorist'},
                                      {'confidence': 0.0002,
                                       'label': 'fascist'},
                                      {'confidence': 0.0001, 'label': 'cult'},
                                      {'confidence': 0.0002,
                                       'label': 'negative_politics'},
                                      {'confidence': 0.0001, 'label': 'kidnap'},
                                      {'confidence': 0.0006, 'label': 'riot'},
                                      {'confidence': 0.0001, 'label': 'parade'},
                                      {'confidence': 0.0002, 'label': 'corpse'},
                                      {'confidence': 0.0001, 'label': 'bloody'},
                                      {'confidence': 0.0001, 'label': 'knife'},
                                      {'confidence': 0.0001, 'label': 'gun'},
                                      {'confidence': 0.0002,
                                       'label': 'military_weapon'},
                                      {'confidence': 0.0,
                                       'label': 'special_characters'},
                                      {'confidence': 0.9994,
                                       'label': 'normal'}]},
            'suggestion': 'pass',
            'url': 'https://sdk-obs-source-save.obs.cn-north-4.myhuaweicloud.com/terrorism.jpg']}]}
```

8.3.2.4 内容审核-图像（异步批量）（V2）

8.3.2.4.1 提交任务

1. 本章节对图像内容审核AK/SK方式使用SDK进行示例说明。

示例代码中可以配置RunTaskSumbitRequest 的Urls配置待审核的图像url（仅支持url方式），配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
using System;
using System.Collections.Generic;
using HuaweiCloud.SDK.Core;
using HuaweiCloud.SDK.Core.Auth;
using HuaweiCloud.SDK.Moderation;
using HuaweiCloud.SDK.Moderation.V2;
using HuaweiCloud.SDK.Moderation.V2.Model;

namespace RunTaskSumbitSolution
{
    class Program
    {
        static void Main(string[] args)
        {
            //认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量
            //中密文存放，使用时解密，确保安全
            //本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
            //HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK
            string accessKey= Environment.GetEnvironmentVariable("HUAWEICLOUD_SDK_AK",
            EnvironmentVariableTarget.Machine);
            string secretKey= Environment.GetEnvironmentVariable("HUAWEICLOUD_SDK_SK",
            EnvironmentVariableTarget.Machine);

            var config = HttpConfig.GetDefaultConfig();
            config.IgnoreSslVerification = true;
            var auth = new BasicCredentials(ak, sk);

            var client = ModerationClient.NewBuilder()
                .WithCredential(auth)
                .WithRegion(ModerationRegion.ValueOf("xxx")) //把xxx替换成服务所在的区域，例如北京四：cn-north-4。
                .WithHttpConfig(config)
                .Build();

            var req = new RunTaskSumbitRequest
            {
            };
            List<TaskSumbitReq.CategoriesEnum> listTaskSumbitReqCategories = new
            List<TaskSumbitReq.CategoriesEnum>();
            listTaskSumbitReqCategories.Add(TaskSumbitReq.CategoriesEnum.FromValue("porn"));
            listTaskSumbitReqCategories.Add(TaskSumbitReq.CategoriesEnum.FromValue("terrorism"));
            List<String> listTaskSumbitReqUrls = new List<String>();
            listTaskSumbitReqUrls.Add("https://XXX.jpg"); //此处替换为公网可以访问的图片地址
            req.Body = new TaskSumbitReq()
            {
                Categories = listTaskSumbitReqCategories,
                Urls = listTaskSumbitReqUrls
            };
        }
        try
        {
            var resp = client.RunTaskSumbit(req);
            var respStatusCode = resp.HttpStatusCode;
            Console.WriteLine(respStatusCode);
            Console.WriteLine(JsonConvert.DeserializeObject(resp.HttpBody));
        }
        catch (RequestTimeoutException requestTimeoutException)
        {
            Console.WriteLine(requestTimeoutException.ErrorMessage);
        }
        catch (ServiceResponseException clientRequestException)
        {
        }
    }
}
```



```
        {
            Console.WriteLine(clientRequestException.HttpStatusCode);
            Console.WriteLine(clientRequestException.ErrorCode);
            Console.WriteLine(clientRequestException.ErrorMsg);
        }
        catch (ConnectionException connectionException)
        {
            Console.WriteLine(connectionException.ErrorMessage);
        }
    }
}
```

2. 控制台输出200即表示程序执行成功，异步批量[任务提交](#)的结果输出到控制台。

```
200
{"result": {"job_id": "xxx"}}
```

8.3.2.4.2 查询任务

1. 本章节对图像内容审核异步批量-查询任务AK/SK方式使用SDK进行示例说明。示例代码中需要修改RunCheckResultRequest的JobId配置审核任务的id，配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
using System;
using System.Collections.Generic;
using HuaweiCloud.SDK.Core;
using HuaweiCloud.SDK.Core.Auth;
using HuaweiCloud.SDK.Moderation;
using HuaweiCloud.SDK.Moderation.V2;
using HuaweiCloud.SDK.Moderation.V2.Model;

namespace RunCheckResultSolution
{
    class Program
    {
        static void Main(string[] args)
        {
            //认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中密文存放，使用时解密，确保安全
            //本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK
            string accessKey= Environment.GetEnvironmentVariable("HUAWEICLOUD_SDK_AK",
                EnvironmentVariableTarget.Machine);
            string secretKey= Environment.GetEnvironmentVariable("HUAWEICLOUD_SDK_SK",
                EnvironmentVariableTarget.Machine);

            var config = HttpConfig.GetDefaultConfig();
            config.IgnoreSslVerification = true;
            var auth = new BasicCredentials(ak, sk);

            var client = ModerationClient.NewBuilder()
                .WithCredential(auth)
                .WithRegion(ModerationRegion.ValueOf("xxx")) //把xxx替换成服务所在的区域，例如北京四：cn-north-4。
                .WithHttpConfig(config)
                .Build();

            var req = new RunCheckResultRequest
            {
                JobId = "2991572e-f503-11eb-a32d-88bd0dde8061"
            };

            try
            {
                var resp = client.RunCheckResult(req);
                var respStatusCode = resp.HttpStatusCode;
                Console.WriteLine(respStatusCode);
            }
        }
    }
}
```

```
        Console.WriteLine(JsonConvert.DeserializeObject(resp.HttpBody));
    }
    catch (RequestTimeoutException requestTimeoutException)
    {
        Console.WriteLine(requestTimeoutException.ErrorMessage);
    }
    catch (ServiceResponseException clientRequestException)
    {
        Console.WriteLine(clientRequestException.HttpStatusCode);
        Console.WriteLine(clientRequestException.ErrorCode);
        Console.WriteLine(clientRequestException.ErrorMessage);
    }
    catch (ConnectionException connectionException)
    {
        Console.WriteLine(connectionException.ErrorMessage);
    }
}
}
```

1. 控制台输出200即表示程序执行成功，异步批量[查询任务](#)的结果输出到控制台。

```
200
{result: {'create_time': '2021-08-04T08:30:41Z',
  'items': [{'category_suggestions': {
    'porn': 'pass',
    'terrorism': 'pass'},
  'detail': {'ad': None,
    'porn': [{'confidence': 0.9883,
      'label': 'normal'},
    {'confidence': 0.0039,
      'label': 'porn'},
    {'confidence': 0.0078,
      'label': 'sexy'}],
    'terrorism': [{'confidence': 0.0002,
      'label': 'terrorist'},
    {'confidence': 0.0002,
      'label': 'fascist'},
    {'confidence': 0.0001,
      'label': 'cult'},
    {'confidence': 0.0002,
      'label': 'negative_politics'},
    {'confidence': 0.0001,
      'label': 'kidnap'},
    {'confidence': 0.0006,
      'label': 'riot'},
    {'confidence': 0.0001,
      'label': 'parade'},
    {'confidence': 0.0002,
      'label': 'corpse'},
    {'confidence': 0.0001,
      'label': 'bloody'},
    {'confidence': 0.0001,
      'label': 'knife'},
    {'confidence': 0.0001,
      'label': 'gun'},
    {'confidence': 0.0002,
      'label': 'military_weapon'},
    {'confidence': 0.0,
      'label': 'special_characters'},
    {'confidence': 0.9994,
      'label': 'normal'}]},
  'suggestion': 'pass',
  'url': 'https://sdk-obs-source-save.obs.cn-north-4.myhuaweicloud.com/terrorism.jpg'}],
  'job_id': 'xxx',
  'status': 'finish',
  'update_time': '2021-08-04T08:30:41Z'}}
```

8.3.3 音频审核

8.3.3.1 创建音频内容审核作业

1. 本章节对创建音频审核AK/SK方式使用SDK进行示例说明。

示例代码中可以使用RunCreateAudioModerationJobRequest类的withUrls配置待审核的音频url（仅支持url方式），配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
using System;
using System.Collections.Generic;
using HuaweiCloud.SDK.Core;
using HuaweiCloud.SDK.Core.Auth;
using HuaweiCloud.SDK.Moderation;
using HuaweiCloud.SDK.Moderation.V3;
using HuaweiCloud.SDK.Moderation.V3.Model;

namespace RunCreateAudioModerationJobSolution
{
    class Program
    {
        static void Main(string[] args)
        {
            //认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量
            //中密文存放，使用时解密，确保安全
            //本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
            //HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK
            string accessKey= Environment.GetEnvironmentVariable("HUAWEICLOUD_SDK_AK",
            EnvironmentVariableTarget.Machine);
            string secretKey= Environment.GetEnvironmentVariable("HUAWEICLOUD_SDK_SK",
            EnvironmentVariableTarget.Machine);

            var config = HttpConfig.GetDefaultConfig();
            config.IgnoreSslVerification = true;
            var auth = new BasicCredentials(ak, sk);

            var client = ModerationClient.NewBuilder()
                .WithCredential(auth)
                .WithRegion(ModerationRegion.ValueOf("xxx")) //把xxx替换成服务所在的区域，例如北京四：cn-north-4。
                .WithHttpConfig(config)
                .Build();

            var req = new RunCreateAudioModerationJobRequest
            {
            };
            List<AudioCreateRequest.CategoriesEnum> listCategoriesbody = new
            List<AudioCreateRequest.CategoriesEnum>();
            listCategoriesbody.Add(AudioCreateRequest.CategoriesEnum.FromValue("porn"));
            AudioInputBody databody = new AudioInputBody()
            {
                Url = ""
            };
            req.Body = new AudioCreateRequest()
            {
                Categories = listCategoriesbody,
                EventType = AudioCreateRequest.EventTypeEnum.FromValue("default"),
                Data = databody
            };

            try
            {
                var resp = client.RunCreateAudioModerationJob(req);
                var respStatusCode = resp.HttpStatusCode;
                Console.WriteLine(respStatusCode); Console.WriteLine(JsonConvert.DeserializeObject(resp.HttpBody));
            }
            catch (RequestTimeoutException requestTimeoutException)
            {
                Console.WriteLine(requestTimeoutException.ErrorMessage);
            }
        }
    }
}
```

```
catch (ServiceResponseException clientRequestException)
{
    Console.WriteLine(clientRequestException.HttpStatusCode);
    Console.WriteLine(clientRequestException.ErrorCode);
    Console.WriteLine(clientRequestException.ErrorMsg);
}
catch (ConnectionException connectionException)
{
    Console.WriteLine(connectionException.ErrorMessage);
}
}
}
```

2. 控制台输出创建音频内容审核作业提交的结果，[创建音频内容审核](#)结果输出到控制台。

```
200
{
  "job_id": "xxx",
  "request_id": "172776779c903659cd1469f272e20e38"
}
```

8.3.3.2 查询音频内容审核作业

1. 本章节对查询音频内容审核AK/SK方式使用SDK进行示例说明。

示例代码中需要配置使用RunQueryAudioModerationJobRequest类的withJobId配置审核任务的id，配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
using System;
using System.Collections.Generic;
using HuaweiCloud.SDK.Core;
using HuaweiCloud.SDK.Core.Auth;
using HuaweiCloud.SDK.Moderation;
using HuaweiCloud.SDK.Moderation.V3;
using HuaweiCloud.SDK.Moderation.V3.Model;

namespace RunQueryAudioModerationJobSolution
{
    class Program
    {
        static void Main(string[] args)
        {
            //认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中密文存放，使用时解密，确保安全
            //本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK
            string accessKey= Environment.GetEnvironmentVariable("HUAWEICLOUD_SDK_AK",
            EnvironmentVariableTarget.Machine);
            string secretKey= Environment.GetEnvironmentVariable("HUAWEICLOUD_SDK_SK",
            EnvironmentVariableTarget.Machine);

            var config = HttpConfig.GetDefaultConfig();
            config.IgnoreSslVerification = true;
            var auth = new BasicCredentials(ak, sk);

            var client = ModerationClient.NewBuilder()
                .WithCredential(auth)
                .WithRegion(ModerationRegion.ValueOf("xxx")) //把xxx替换成服务所在的区域，例如北京四：cn-north-4。
                .WithHttpConfig(config)
                .Build();

            var req = new RunQueryAudioModerationJobRequest
            {
                JobId = "xxx" //此处需要输入您的jobID
            };
        }
    }
}
```

```
try
{
    var resp = client.RunQueryAudioModerationJob(req);
    var respStatusCode = resp.HttpStatusCode;
    Console.WriteLine(respStatusCode);
    Console.WriteLine(JsonConvert.DeserializeObject(resp.HttpBody));
}
catch (RequestTimeoutException requestTimeoutException)
{
    Console.WriteLine(requestTimeoutException.ErrorMessage);
}
catch (ServiceResponseException clientRequestException)
{
    Console.WriteLine(clientRequestException.HttpStatusCode);
    Console.WriteLine(clientRequestException.ErrorCode);
    Console.WriteLine(clientRequestException.ErrorMessage);
}
catch (ConnectionException connectionException)
{
    Console.WriteLine(connectionException.ErrorMessage);
}
}
```

2. 控制台输出200即表示程序执行成功，[查询音频内容审核](#)结果输出到控制台。

```
200
{
  "job_id":"xxx",
  "status":"succeeded",
  "result":{
    "suggestion":"block",
    "details":[
      {
        "start_time":0,
        "suggestion":"block",
        "end_time":10,
        "label":"porn",
        "audio_text":"xxx",
        "segments":[
          {
            "segment":"xxx"
          },
          {
            "segment":"xxx"
          },
          {
            "segment":"xxx"
          }
        ]
      },
      {
        "start_time":30,
        "suggestion":"block",
        "end_time":40,
        "label":"porn",
        "audio_text":"xxx",
        "segments":[
          {
            "segment":"xx"
          }
        ]
      }
    ],
    "audio_text":"xxxxxx"
  },
  "request_params":{
    "event_type":"default",
    "data":{
```

```
    "url": "https://xxxx.wav"
  },
  "callback": "http://xxx",
  "categories": [
    "porn",
    "ad"
  ]
},
"create_time": "2022-06-20 11:05:29",
"update_time": "2022-06-20 11:07:08",
"request_id": "2419446b1fe14203f64e4018d12db3dd"
}
```

8.3.4 视频审核

8.3.4.1 创建视频内容审核作业

1. 本章节对创建视频审核AK/SK方式使用SDK进行示例说明。

示例代码中可以使用RunCreateVideoModerationJobRequest类的withUrls配置待审核的视频url（仅支持url方式），配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
using System;
using System.Collections.Generic;
using HuaweiCloud.SDK.Core;
using HuaweiCloud.SDK.Core.Auth;
using HuaweiCloud.SDK.Moderation;
using HuaweiCloud.SDK.Moderation.V3;
using HuaweiCloud.SDK.Moderation.V3.Model;

namespace RunCreateVideoModerationJobSolution
{
    class Program
    {
        static void Main(string[] args)
        {
            //认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中
            //明文存放，使用时解密，确保安全
            //本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
            HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK
            string accessKey= Environment.GetEnvironmentVariable("HUAWEICLOUD_SDK_AK",
            EnvironmentVariableTarget.Machine);
            string secretKey= Environment.GetEnvironmentVariable("HUAWEICLOUD_SDK_SK",
            EnvironmentVariableTarget.Machine);

            var config = HttpConfig.GetDefaultConfig();
            config.IgnoreSslVerification = true;
            var auth = new BasicCredentials(ak, sk);

            var client = ModerationClient.NewBuilder()
                .WithCredential(auth)
                .WithRegion(ModerationRegion.ValueOf("xxx")) //把xxx替换成服务所在的区域，例如北京四：cn-north-4
                .WithHttpConfig(config)
                .Build();

            var req = new RunCreateVideoModerationJobRequest
            {
            };
            List<VideoCreateRequest.AudioCategoriesEnum> listAudioCategoriesbody = new
            List<VideoCreateRequest.AudioCategoriesEnum>();
            listAudioCategoriesbody.Add(VideoCreateRequest.AudioCategoriesEnum.FromValue("porn"));
            List<VideoCreateRequest.ImageCategoriesEnum> listImageCategoriesbody = new
            List<VideoCreateRequest.ImageCategoriesEnum>();
            listImageCategoriesbody.Add(VideoCreateRequest.ImageCategoriesEnum.FromValue("porn"));
            VideoCreateRequestData databody = new VideoCreateRequestData()
```

```
{
    Url = "http://xxx.xx.xxx",
    FrameInterval = 5.5
};
req.Body = new VideoCreateRequest()
{
    Callback = "http://xxx.xx.xxx",
    AudioCategories = listAudioCategoriesbody,
    ImageCategories = listImageCategoriesbody,
    EventType = VideoCreateRequest.EventTypeEnum.FromValue("default"),
    Data = databody
};

try
{
    var resp = client.RunCreateVideoModerationJob(req);
    var respStatusCode = resp.HttpStatusCode;
    Console.WriteLine(respStatusCode);
}
catch (RequestTimeoutException requestTimeoutException)
{
    Console.WriteLine(requestTimeoutException.ErrorMessage);
}
catch (ServiceResponseException clientRequestException)
{
    Console.WriteLine(clientRequestException.HttpStatusCode);
    Console.WriteLine(clientRequestException.ErrorCode);
    Console.WriteLine(clientRequestException.ErrorMsg);
}
catch (ConnectionException connectionException)
{
    Console.WriteLine(connectionException.ErrorMessage);
}
}
```

2. 控制台输出创建视频内容审核作业的结果，[创建视频内容审核结果输出到控制台](#)。

```
200
{
  "request_id":"dbbc75d1a174bfa899a38be38fdd107",
  "job_id":"xxx"
}
```

8.3.4.2 查询视频内容审核作业

1. 本章节对查询视频审核AK/SK方式使用SDK进行示例说明。

示例代码中需要配置使用RunQueryVideoModerationJobRequest类的withJobId配置审核任务的id，配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
using System;
using System.Collections.Generic;
using HuaweiCloud.SDK.Core;
using HuaweiCloud.SDK.Core.Auth;
using HuaweiCloud.SDK.Moderation;
using HuaweiCloud.SDK.Moderation.V3;
using HuaweiCloud.SDK.Moderation.V3.Model;
```

```
namespace RunQueryVideoModerationJobSolution
{
```

```
    class Program
    {
        static void Main(string[] args)
        {
```

```
            //认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中密文存放，使用时解密，确保安全
```

```
//本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
//HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK
string accessKey= Environment.GetEnvironmentVariable("HUAWEICLOUD_SDK_AK",
EnvironmentVariableTarget.Machine);
string secretKey= Environment.GetEnvironmentVariable("HUAWEICLOUD_SDK_SK",
EnvironmentVariableTarget.Machine);

var config = HttpConfig.GetDefaultConfig();
config.IgnoreSslVerification = true;
var auth = new BasicCredentials(ak, sk);

var client = ModerationClient.NewBuilder()
    .WithCredential(auth)
    .WithRegion(ModerationRegion.ValueOf("xxx")) //把xxx替换成服务所在的区域，例如北京四：cn-north-4
    .WithHttpConfig(config)
    .Build();

var req = new RunQueryVideoModerationJobRequest
{
    JobId = "xxxx" //此处需要输入您的jobID
};

try
{
    var resp = client.RunQueryVideoModerationJob(req);
    var respStatusCode = resp.HttpStatusCode;
    Console.WriteLine(respStatusCode);
    Console.WriteLine(JsonConvert.DeserializeObject(resp.HttpBody));
}
catch (RequestTimeoutException requestTimeoutException)
{
    Console.WriteLine(requestTimeoutException.ErrorMessage);
}
catch (ServiceResponseException clientRequestException)
{
    Console.WriteLine(clientRequestException.HttpStatusCode);
    Console.WriteLine(clientRequestException.ErrorCode);
    Console.WriteLine(clientRequestException.ErrorMsg);
}
catch (ConnectionException connectionException)
{
    Console.WriteLine(connectionException.ErrorMessage);
}
}
}
```

- 控制台输出查询视频内容审核作业提交的结果，[查询视频内容审核](#)结果输出到控制台。

```
200
{
  "request_id": "xxxx",
  "job_id": "xxx",
  "status": "succeeded",
  "request_params": {
    "data": {
      "url": "xxxx",
      "frame_interval": 5
    },
    "event_type": "default",
    "image_categories": [
      "porn",
      "terrorism",
      "image_text"
    ],
    "audio_categories": [
      "porn",
      "ad",
      "moan",

```



```
"abuse"
  ],
  "callback":"xxx"
},
"create_time":"2022-07-30T08:57:11.011Z",
"update_time":"2022-07-30T08:57:14.014Z",
"result":{
  "suggestion":"block",
  "image_detail":[
    {
      "suggestion":"block",
      "category":"porn",
      "ocr_text":"123",
      "time":0,
      "detail":[
        {
          "confidence":0.676015138626099,
          "category":"porn",
          "suggestion":"block",
          "label":"pornography"
        },
        {
          "confidence":0.666015138626099,
          "category":"porn",
          "suggestion":"review",
          "label":"sexy_female"
        },
        {
          "confidence":0.666015138626099,
          "category":"xxx",
          "suggestion":"block",
          "label":"leader"
        },
        {
          "confidence":0.691423773765564,
          "category":"xxx",
          "suggestion":"block",
          "face_location":{
            "top_left_x":12,
            "top_left_y":13,
            "bottom_right_x":200,
            "bottom_right_y":240
          },
          "label":"leader"
        },
        {
          "confidence":0.691423773765564,
          "category":"porn",
          "suggestion":"block",
          "face_location":{
            "top_left_x":12,
            "top_left_y":13,
            "bottom_right_x":100,
            "bottom_right_y":2000
          },
          "label":"leader"
        },
        {
          "confidence":0.891423773765564,
          "category":"image_text",
          "suggestion":"block",
          "label":"qr_code"
        },
        {
          "confidence":0.891423773765564,
          "category":"image_text",
          "suggestion":"block",
          "label":"qr_code",
          "qr_content":"xxxxxxxxx",

```

```
        "qr_location":{
            "top_left_x":12,
            "top_left_y":13,
            "bottom_right_x":100,
            "bottom_right_y":2000
        }
    },
    {
        "confidence":0.891423773765564,
        "category":"image_text",
        "suggestion":"block",
        "label":"porn",
        "segments":[
            {
                "segment":"xxx"
            },
            {
                "segment":"xx"
            }
        ]
    }
]
},
"audio_detail":[
    {
        "suggestion":"block",
        "label":"porn",
        "audio_text":"xxxxx",
        "end_time":10,
        "start_time":0,
        "detail":[
            {
                "confidence":0.999117187582349,
                "label":"porn",
                "suggestion":"block",
                "segments":[
                    {
                        "segment":"xxx"
                    }
                ]
            },
            {
                "confidence":0.885117187582349,
                "label":"porn",
                "suggestion":"block"
            }
        ]
    }
]
}
```

8.3.5 音频流审核

8.3.5.1 创建音频流内容审核作业

1. 本章节对创建音频流审核AK/SK方式使用SDK进行示例说明。

示例代码中可以使用AudioStreamCreateRequestData类的withUrls配置待审核的音频流url（仅支持url方式），配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
using System;
using System.Collections.Generic;
using HuaweiCloud.SDK.Core;
using HuaweiCloud.SDK.Core.Auth;
```

```
using HuaweiCloud.SDK.Moderation;
using HuaweiCloud.SDK.Moderation.V3;
using HuaweiCloud.SDK.Moderation.V3.Model;

namespace RunCreateAudioStreamModerationJobSolution
{
    class Program
    {
        static void Main(string[] args)
        {
            //认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量
            //中密文存放，使用时解密，确保安全
            //本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
            //HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK
            string accessKey= Environment.GetEnvironmentVariable("HUAWEICLOUD_SDK_AK",
            EnvironmentVariableTarget.Machine);
            string secretKey= Environment.GetEnvironmentVariable("HUAWEICLOUD_SDK_SK",
            EnvironmentVariableTarget.Machine);
            const string projectId = "<YOUR_PROJECT_ID>";

            var config = HttpConfig.GetDefaultConfig();
            config.IgnoreSslVerification = true;
            var auth = new BasicCredentials(ak, sk, projectId);

            var client = ModerationClient.NewBuilder()
                .WithCredential(auth)
                .WithRegion(ModerationRegion.ValueOf("xxx")) //把xxx替换成服务所在的区域，例如北京
                //四：cn-north-4。
                .WithHttpConfig(config)
                .Build();

            var req = new RunCreateAudioStreamModerationJobRequest
            {
            };
            List<string> listCategoriesbody = new List<string>();
            listCategoriesbody.Add("ad");
            AudioStreamCreateRequestData databody = new AudioStreamCreateRequestData()
            {
                Url = "http://xxx"
            };
            req.Body = new AudioStreamCreateRequest()
            {
                Callback = "http://xxxxxx",
                Categories = listCategoriesbody,
                EventType = "default",
                Data = databody
            };
        }
        try
        {
            var resp = client.RunCreateAudioStreamModerationJob(req);
            var respStatusCode = resp.HttpStatusCode;
            Console.WriteLine(respStatusCode);
            Console.WriteLine(resp);
        }
        catch (RequestTimeoutException requestTimeoutException)
        {
            Console.WriteLine(requestTimeoutException.ErrorMessage);
        }
        catch (ServiceResponseException clientRequestException)
        {
            Console.WriteLine(clientRequestException.HttpStatusCode);
            Console.WriteLine(clientRequestException.RequestId);
            Console.WriteLine(clientRequestException.ErrorCode);
            Console.WriteLine(clientRequestException.ErrorMsg);
        }
        catch (ConnectionException connectionException)
        {
            Console.WriteLine(connectionException.ErrorMessage);
        }
    }
}
```

```
    }  
  }  
}
```

2. 控制台输出创建音频流内容审核作业的结果。

```
200  
{  
  "request_id":"303fbd70c9813a2476bad125fa436f00",  
  "job_id":"11690387200_ffa1f5bb9a58451b86816gdfc654610e"  
}
```

8.3.5.2 关闭音频流内容审核作业

1. 本章节对关闭音频流审核AK/SK方式使用SDK进行示例说明。

示例代码中需要配置使用RunCloseAudioStreamModerationJobRequest类的withJobId配置审核任务的id，配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
using System;  
using System.Collections.Generic;  
using HuaweiCloud.SDK.Core;  
using HuaweiCloud.SDK.Core.Auth;  
using HuaweiCloud.SDK.Moderation;  
using HuaweiCloud.SDK.Moderation.V3;  
using HuaweiCloud.SDK.Moderation.V3.Model;  
  
namespace RunCloseAudioStreamModerationJobSolution  
{  
  class Program  
  {  
    static void Main(string[] args)  
    {  
      //认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量  
      //中密文存放，使用时解密，确保安全  
      //本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量  
      HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK  
      string accessKey= Environment.GetEnvironmentVariable("HUAWEICLOUD_SDK_AK",  
EnvironmentVariableTarget.Machine);  
      string secretKey= Environment.GetEnvironmentVariable("HUAWEICLOUD_SDK_SK",  
EnvironmentVariableTarget.Machine);  
      const string projectId = "<YOUR_PROJECT_ID>";  
  
      var config = HttpConfig.GetDefaultConfig();  
      config.IgnoreSslVerification = true;  
      var auth = new BasicCredentials(ak, sk, projectId);  
  
      var client = ModerationClient.NewBuilder()  
        .WithCredential(auth)  
        .WithRegion(ModerationRegion.ValueOf("xxx"))//把xxx替换成服务所在的区域，例如北京  
        .WithHttpConfig(config)  
        .Build();  
  
      var req = new RunCloseAudioStreamModerationJobRequest  
      {  
        JobId = "xxxxx"  
      };  
  
      try  
      {  
        var resp = client.RunCloseAudioStreamModerationJob(req);  
        var respStatusCode = resp.HttpStatusCode;  
        Console.WriteLine(respStatusCode);  
        Console.WriteLine(resp);  
      }  
      catch (RequestTimeoutException requestTimeoutException)  
      {
```

```
        Console.WriteLine(requestTimeoutException.ErrorMessage);
    }
    catch (ServiceResponseException clientRequestException)
    {
        Console.WriteLine(clientRequestException.HttpStatusCode);
        Console.WriteLine(clientRequestException.RequestId);
        Console.WriteLine(clientRequestException.ErrorCode);
        Console.WriteLine(clientRequestException.ErrorMessage);
    }
    catch (ConnectionException connectionException)
    {
        Console.WriteLine(connectionException.ErrorMessage);
    }
}
}
```

2. 控制台输出关闭音频流内容审核作业提交的结果。

```
200
{
  "result": "success"
}
```

8.3.6 视频流审核

8.3.6.1 创建视频流内容审核作业

1. 本章节对创建视频审核AK/SK方式使用SDK进行示例说明。

示例代码中可以使用VideoStreamCreateRequestData类的withUrls配置待审核的视频url（仅支持url方式），配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
using System;
using System.Collections.Generic;
using HuaweiCloud.SDK.Core;
using HuaweiCloud.SDK.Core.Auth;
using HuaweiCloud.SDK.Moderation;
using HuaweiCloud.SDK.Moderation.V3;
using HuaweiCloud.SDK.Moderation.V3.Model;

namespace RunCreateVideoStreamModerationJobSolution
{
    class Program
    {
        static void Main(string[] args)
        {
            //认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中密文存放，使用时解密，确保安全
            //本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK
            string accessKey= Environment.GetEnvironmentVariable("HUAWEICLOUD_SDK_AK",
            EnvironmentVariableTarget.Machine);
            string secretKey= Environment.GetEnvironmentVariable("HUAWEICLOUD_SDK_SK",
            EnvironmentVariableTarget.Machine);
            const string projectId = "<YOUR_PROJECT_ID>";

            var config = HttpConfig.GetDefaultConfig();
            config.IgnoreSslVerification = true;
            var auth = new BasicCredentials(ak, sk, projectId);

            var client = ModerationClient.NewBuilder()
                .WithCredential(auth)
                .WithRegion(ModerationRegion.ValueOf("xxx"))//把xxx替换成服务所在的区域，例如北京
                .WithHttpConfig(config)
                .Build();
```

```
var req = new RunCreateVideoStreamModerationJobRequest
{
};
List<VideoStreamCreateRequest.ImageCategoriesEnum> listImageCategoriesbody = new
List<VideoStreamCreateRequest.ImageCategoriesEnum>();
listImageCategoriesbody.Add(VideoStreamCreateRequest.ImageCategoriesEnum.FromValue("politics"));
;
VideoStreamCreateRequestData databody = new VideoStreamCreateRequestData()
{
    Url = "http://xxx"
};
req.Body = new VideoStreamCreateRequest()
{
    Callback = "http://xxxxxx",
    ImageCategories = listImageCategoriesbody,
    EventType = VideoStreamCreateRequest.EventTypeEnum.FromValue("default"),
    Data = databody
};
try
{
    var resp = client.RunCreateVideoStreamModerationJob(req);
    var respStatusCode = resp.HttpStatusCode;
    Console.WriteLine(respStatusCode);
    Console.WriteLine(resp);
}
catch (RequestTimeoutException requestTimeoutException)
{
    Console.WriteLine(requestTimeoutException.ErrorMessage);
}
catch (ServiceResponseException clientRequestException)
{
    Console.WriteLine(clientRequestException.HttpStatusCode);
    Console.WriteLine(clientRequestException.RequestId);
    Console.WriteLine(clientRequestException.ErrorCode);
    Console.WriteLine(clientRequestException.ErrorMsg);
}
catch (ConnectionException connectionException)
{
    Console.WriteLine(connectionException.ErrorMessage);
}
}
}
```

2. 控制台输出创建视频内容审核作业的结果。

```
200
{
  "request_id":"fa520ddb6b54af51c8d1e6d19898656",
  "job_id":"01690387200_09ad14f155bb47e9bf2718ace57c73ed"
}
```

8.3.6.2 关闭视频流内容审核作业

1. 本章节对关闭视频流AK/SK方式使用SDK进行示例说明。

示例代码中需要配置使用RunCloseVideoStreamModerationJobRequest类的withJobId配置审核任务的id，配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
using System;
using System.Collections.Generic;
using HuaweiCloud.SDK.Core;
using HuaweiCloud.SDK.Core.Auth;
using HuaweiCloud.SDK.Moderation;
using HuaweiCloud.SDK.Moderation.V3;
using HuaweiCloud.SDK.Moderation.V3.Model;
```

```
namespace RunCloseVideoStreamModerationJobSolution
{
    class Program
    {
        static void Main(string[] args)
        {
            //认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量
            //中密文存放，使用时解密，确保安全
            //本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
            //HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK
            string accessKey= Environment.GetEnvironmentVariable("HUAWEICLOUD_SDK_AK",
            EnvironmentVariableTarget.Machine);
            string secretKey= Environment.GetEnvironmentVariable("HUAWEICLOUD_SDK_SK",
            EnvironmentVariableTarget.Machine);
            const string projectId = "<YOUR_PROJECT_ID>";

            var config = HttpConfig.GetDefaultConfig();
            config.IgnoreSslVerification = true;
            var auth = new BasicCredentials(ak, sk, projectId);

            var client = ModerationClient.NewBuilder()
                .WithCredential(auth)
                .WithRegion(ModerationRegion.ValueOf("xxx"))//把xxx替换成服务所在的区域，例如北京
                .WithHttpConfig(config)
                .Build();

            var req = new RunCloseVideoStreamModerationJobRequest
            {
                JobId = "xxxxx"
            };

            try
            {
                var resp = client.RunCloseVideoStreamModerationJob(req);
                var respStatusCode = resp.HttpStatusCode;
                Console.WriteLine(respStatusCode);
                Console.WriteLine(resp);
            }
            catch (RequestTimeoutException requestTimeoutException)
            {
                Console.WriteLine(requestTimeoutException.ErrorMessage);
            }
            catch (ServiceResponseException clientRequestException)
            {
                Console.WriteLine(clientRequestException.HttpStatusCode);
                Console.WriteLine(clientRequestException.RequestId);
                Console.WriteLine(clientRequestException.ErrorCode);
                Console.WriteLine(clientRequestException.ErrorMessage);
            }
            catch (ConnectionException connectionException)
            {
                Console.WriteLine(connectionException.ErrorMessage);
            }
        }
    }
}
```

2. 控制台输出关闭视频流内容审核作业提交的结果。

```
200
{
    "result": "success"
}
```

8.3.7 文档审核

8.3.7.1 创建文档内容审核作业

1. 本章节对创建文档内容审核作业AK/SK方式使用SDK进行示例说明。

示例代码中可以使用DocumentCreateRequestData类的withUrl配置待审核的文档url，配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
using System;
using System.Collections.Generic;
using HuaweiCloud.SDK.Core;
using HuaweiCloud.SDK.Core.Auth;
using HuaweiCloud.SDK.Moderation;
using HuaweiCloud.SDK.Moderation.V3;
using HuaweiCloud.SDK.Moderation.V3.Model;

namespace RunCreateDocumentModerationJobSolution
{
    class Program
    {
        static void Main(string[] args)
        {
            //认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量
            //中密文存放，使用时解密，确保安全
            //本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
            //HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK
            string accessKey= Environment.GetEnvironmentVariable("HUAWEICLOUD_SDK_AK",
            EnvironmentVariableTarget.Machine);
            string secretKey= Environment.GetEnvironmentVariable("HUAWEICLOUD_SDK_SK",
            EnvironmentVariableTarget.Machine);
            const string projectId = "<YOUR_PROJECT_ID>";

            var config = HttpConfig.GetDefaultConfig();
            config.IgnoreSslVerification = true;
            var auth = new BasicCredentials(ak, sk, projectId);

            var client = ModerationClient.NewBuilder()
                .WithCredential(auth)
                .WithRegion(ModerationRegion.ValueOf("xxx"))//把xxx替换成服务所在的区域，例如北京
                .WithHttpConfig(config)
                .Build();

            var req = new RunCreateDocumentModerationJobRequest
            {
            };
            DocumentCreateRequestData databody = new DocumentCreateRequestData()
            {
                Url = "http://xxx",
                Format = "pdf"
            };
            req.Body = new DocumentCreateRequest()
            {
                EventType = "default",
                Data = databody
            };
            try
            {
                var resp = client.RunCreateDocumentModerationJob(req);
                var respStatusCode = resp.HttpStatusCode;
                Console.WriteLine(respStatusCode);
                Console.WriteLine(resp);
            }
            catch (RequestTimeoutException requestTimeoutException)
            {
                Console.WriteLine(requestTimeoutException.ErrorMessage);
            }
            catch (ServiceResponseException clientRequestException)
            {
            }
        }
    }
}
```



```
        {
            Console.WriteLine(clientRequestException.HttpStatusCode);
            Console.WriteLine(clientRequestException.RequestId);
            Console.WriteLine(clientRequestException.ErrorCode);
            Console.WriteLine(clientRequestException.ErrorMessage);
        }
        catch (ConnectionException connectionException)
        {
            Console.WriteLine(connectionException.ErrorMessage);
        }
    }
}
```

2. 控制台输出创建视频内容审核作业的结果。

```
200
{
  "request_id":"c083957284ce1f2c0b910115d8232ec6",
  "job_id":"01690387200_e3dfb78d9f20463f98aee25e6895203a"
}
```

8.3.7.2 查询文档审核作业结果

1. 本章节对查询文档审核作业结果AK/SK方式使用SDK进行示例说明。

示例代码中需要配置使用RunQueryDocumentModerationJobRequest类的withJobId配置审核任务的id，配置完成后运行即可。

服务所在的应用区域和终端节点，详情请查看 [地区和终端节点](#)。

```
using System;
using System.Collections.Generic;
using HuaweiCloud.SDK.Core;
using HuaweiCloud.SDK.Core.Auth;
using HuaweiCloud.SDK.Moderation;
using HuaweiCloud.SDK.Moderation.V3;
using HuaweiCloud.SDK.Moderation.V3.Model;

namespace RunQueryDocumentModerationJobSolution
{
    class Program
    {
        static void Main(string[] args)
        {
            //认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量
            //中密文存放，使用时解密，确保安全
            //本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
            //HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK
            string accessKey= Environment.GetEnvironmentVariable("HUAWEICLOUD_SDK_AK",
            EnvironmentVariableTarget.Machine);
            string secretKey= Environment.GetEnvironmentVariable("HUAWEICLOUD_SDK_SK",
            EnvironmentVariableTarget.Machine);
            const string projectId = "<YOUR_PROJECT_ID>";

            var config = HttpConfig.GetDefaultConfig();
            config.IgnoreSslVerification = true;
            var auth = new BasicCredentials(ak, sk, projectId);

            var client = ModerationClient.NewBuilder()
                .WithCredential(auth)
                .WithRegion(ModerationRegion.ValueOf("xxx"))//把xxx替换成服务所在的区域，例如北京
                .WithHttpConfig(config)
                .Build();

            var req = new RunQueryDocumentModerationJobRequest
            {
                JobId = "xxxx"
            };
        }
    }
}
```

```
try
{
    var resp = client.RunQueryDocumentModerationJob(req);
    var respStatusCode = resp.HttpStatusCode;
    Console.WriteLine(respStatusCode);
1.1    Console.WriteLine(resp);
}
catch (RequestTimeoutException requestTimeoutException)
{
    Console.WriteLine(requestTimeoutException.ErrorMessage);
}
catch (ServiceResponseException clientRequestException)
{
    Console.WriteLine(clientRequestException.HttpStatusCode);
    Console.WriteLine(clientRequestException.RequestId);
    Console.WriteLine(clientRequestException.ErrorCode);
    Console.WriteLine(clientRequestException.ErrorMessage);
}
catch (ConnectionException connectionException)
{
    Console.WriteLine(connectionException.ErrorMessage);
}
}
```

2. 控制台输出查询文档审核作业结果。

```
200
{
  "job_id": "01693238400_cd7c8fd0a30b4c13b1fcaf9246c542c6",
  "status": "succeeded",
  "result": {
    "suggestion": "pass",
    "details": [
      ]
    },
  "request_params": {
    "data": {
      "url": "http://xxxxx",
      "format": "pdf"
    },
    "event_type": "default"
  },
  "create_time": "2023-08-29T16:33:50.050Z",
  "update_time": "2023-08-29T16:33:50.050Z",
  "request_id": "e4eeb972aeb92f4269e8dfbb59ba7e3b"
}
```

9 使用 SDK (GO)

9.1 GO 开发环境配置

使用图像识别GO版本SDK包，需要您配置Go语言开发环境。如已经安装配置go开发环境，可跳过

1. 在[golang 官网](#) 下载并安装对应的go版本，sdk支持 go 1.14 及以上版本。
2. 下载完成后按照提示安装，位置自选，比如安装到本地D:\develop\go。
3. 配置GO环境变量：右键“计算机>属性>高级系统设置>环境变量”，进行如下操作。
 - a. 新建系统变量GOROOT，变量值为实际go安装位置。新增环境变量GOPATH，变量为go语言工程所在目录。
 - b. 修改PATH变量，末尾增加GO的编译器路径“;%GOROOT%\bin”，中间分隔符是【;】。
 - c. 修改CLASSPATH变量，末尾增加GO的编译器路径“;%GOROOT%\lib”，中间分隔符是【;】。
4. 在vs code官网：<https://code.visualstudio.com/Download>，下载安装操作系统的vs code。
5. 安装go语言的插件Go和 Go Critic。

9.2 GO 语言 SDK 获取和安装

1. GO语言SDK的获取和安装使用命令，参考方法如下：

```
go get -u github.com/huaweicloud/huaweicloud-sdk-go-v3
```

9.3 SDK 调用示例

9.3.1 文本审核

9.3.1.1 文本内容审核 (V3)

1. 本章节对文本内容审核AK/SK方式使用SDK进行示例说明。

示例代码中可以将TextDetectionDataReq的text配置待检测的文本信息，配置完成后运行即可。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    moderation "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/region"
)

func main() {
    // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中
    // 密文存放，使用时解密，确保安全；
    // 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
    // HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK。
    ak := os.Getenv("HUAWEICLOUD_SDK_AK")
    sk := os.Getenv("HUAWEICLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := moderation.NewModerationClient(
        moderation.ModerationClientBuilder().
            WithRegion(region.ValueOf("xxx")). //把xxx替换成服务所在的区域，例如北京四：cn-north-4。
            WithCredential(auth).
            Build())

    request := &model.RunTextModerationRequest{}
    databody := &model.TextDetectionDataReq{
        Text: "text",
    }
    request.Body = &model.TextDetectionReq{
        Data: databody,
        EventType: "comment",
    }
    response, err := client.RunTextModeration(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

2. 审核结果输出到控制台表示执行成功，[文本内容审核结果](#)输出到控制台。

```
RunTextModerationResponse {"request_id": "d81702cfd3599e103030c3d52d6f955f", "result": {"suggestion": "pass", "label": "normal", "details": []}}
```

9.3.1.2 文本内容审核 (V2)

1. 本章节对文本内容审核AK/SK方式使用SDK进行示例说明。

示例代码中可以将TextDetectionItemsReq的Text配置待检测的文本信息，配置完成后运行即可。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    moderation "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v2/model"
)
```

```
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v2/region"
  )

  func main() {
    // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中
    // 密文存放，使用时解密，确保安全；
    // 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
    // HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK。
    ak := os.Getenv("HUAWEICLOUD_SDK_AK")
    sk := os.Getenv("HUAWEICLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
      WithAk(ak).
      WithSk(sk).
      Build()

    client := moderation.NewModerationClient(
      moderation.ModerationClientBuilder().
        WithRegion(region.ValueOf("xxx")).//把xxx替换成服务所在的区域，例如北京四：cn-north-4。
        WithCredential(auth).
        Build()

    request := &model.RunTextModerationRequest{
      typeItemsTextDetectionItemsReq:= "content"
      var listItemsbody = []model.TextDetectionItemsReq{
        {
          Text: "text", //此处输入待检测文本
          Type: &typeItemsTextDetectionItemsReq,
        },
      }
      var listCategoriesbody = []string{
        "porn",
        "ad",
        "abuse",
        "contraband",
        "flood",
      }
      request.Body = &model.TextDetectionReq{
        Items: listItemsbody,
        Categories: &listCategoriesbody,
      }
      response, err := client.RunTextModeration(request)
      if err == nil {
        fmt.Printf("%+v\n", response)
      } else {
        fmt.Println(err)
      }
    }
  }
```

2. 审核结果输出到控制台表示执行成功，**文本内容审核结果**输出到控制台。

```
RunTextModerationResponse {"result":{"suggestion":"block","detail":{"contraband":["亚神酸钾"],"porn":["luo聊"]}}}
```

Process exiting with code: 0

9.3.2 图像审核

9.3.2.1 图像内容审核 (V3)

1. 本章节对图像内容审核AK/SK方式使用SDK进行示例说明。

示例代码中可以使用Url或Image配置图像信息(代表使用url或者图像的base64)，示例中以url为例配置完成后运行即可。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    moderation "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v3"
```

```
"github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v3/model"
region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v3/region"
)

func main() {
    // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中
    // 密文存放，使用时解密，确保安全；
    // 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
    // HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK。
    ak := os.Getenv("HUAWEICLOUD_SDK_AK")
    sk := os.Getenv("HUAWEICLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := moderation.NewModerationClient(
        moderation.ModerationClientBuilder().
            WithRegion(region.ValueOf("xxx")). //把xxx替换成服务所在的区域，例如北京四：cn-north-4。
            WithCredential(auth).
            Build())

    request := &model.CheckImageModerationRequest{}
    var listCategoriesbody = []string{
        "porn",
    }
    urlImageDetectionReq := "http://www.xxxx.xxx"
    request.Body = &model.ImageDetectionReq{
        Url: &urlImageDetectionReq,
        Categories: listCategoriesbody,
        EventType: "head_image",
    }
    response, err := client.CheckImageModeration(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

2. 审核结果输出到控制台表示执行成功，[图像内容审核结果](#)输出到控制台。

```
CheckImageModerationResponse
{"request_id": "69b424920668cb15e6d25eef879a023b", "result": {"suggestion": "block", "category":
"image_text", "details": [{"suggestion": "block", "category": "image_text", "confidence": 1.0,
"qr_location": {"top_left_x": 74, "top_left_y": 66, "bottom_right_x": 151, "bottom_right_y": 142},
"qr_content": "http://xxxxxx/xxxxx", "label": "qr_code"}, {"suggestion": "block", "category":
"image_text", "confidence": 0.98779297, "label": "ad"}, {"suggestion": "block", "category":
"image_text", "confidence": 1.0, "label": "qr_code"}], "ocr_text": "xxxxxxxxx"}}
```

9.3.2.2 图像内容审核 (V2)

1. 本章节对图像内容审核AK/SK方式使用SDK进行示例说明。

示例代码中可以使用Url或Image配置图像信息(代表使用url或者图像的base64)，示例中以url为例配置完成后运行即可。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    moderation "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v2/region"
)

func main() {
    // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中
```

```
密文存放, 使用时解密, 确保安全;
// 本示例以ak和sk保存在环境变量中来实现身份验证为例, 运行本示例前请先在本地环境中设置环境变量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK。
ak := os.Getenv("HUAWEICLOUD_SDK_AK")
sk := os.Getenv("HUAWEICLOUD_SDK_SK")

auth := basic.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    Build()

client := moderation.NewModerationClient(
    moderation.ModerationClientBuilder().
        WithRegion(region.ValueOf("xxx")). //把xxx替换成服务所在的区域, 例如北京四: cn-north-4。
        WithCredential(auth).
        Build())

request := &model.RunImageModerationRequest{
var listCategoriesbody = []model.ImageDetectionReqCategories{
    model.ImageDetectionReqCategoriesEnum().PORN,
    model.ImageDetectionReqCategoriesEnum().TERRORISM,
    model.ImageDetectionReqCategoriesEnum().AD,
}
thresholdImageDetectionReq:= float32(0)
moderationRuleImageDetectionReq:= "default"
urlImageDetectionReq:= "https://XXX.jpg" //此处替换为公网可以访问的图片地址
request.Body = &model.ImageDetectionReq{
    Threshold: &thresholdImageDetectionReq,
    Categories: &listCategoriesbody,
    ModerationRule: &moderationRuleImageDetectionReq,
    Url: &urlImageDetectionReq,
}
}
response, err := client.RunImageModeration(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}}
```

2. 审核结果输出到控制台表示执行成功, **图像内容审核结果**输出到控制台。

```
RunImageModerationResponse {"result":{"detail":{"porn":{"confidence":0.9883,"label":"normal"},
{"confidence":0.0039,"label":"porn"},{"confidence":0.0078,"label":"sexy"},"terrorism":
[{"confidence":0.0002,"label":"terrorist"},{"confidence":0.0002,"label":"fascist"},
{"confidence":0.0001,"label":"cult"},{"confidence":0.0001,"label":"kidnap"},
{"confidence":0.0006,"label":"riot"},{"confidence":0.0001,"label":"parade"},
{"confidence":0.0002,"label":"corpse"},{"confidence":0.0001,"label":"bloody"},
{"confidence":0.0001,"label":"knife"},{"confidence":0.0001,"label":"gun"},
{"confidence":0.0002,"label":"military_weapon"},{"confidence":0,"label":"special_characters"},
{"confidence":0.9994,"label":"normal"},"ad":{"confidence":0,"label":"ad"},
{"confidence":1,"label":"normal"},"suggestion":"pass","category_suggestions":
{"ad":"pass","porn":"pass","terrorism":"pass"}}
```

Process exiting with code: 0

9.3.2.3 图像内容审核（批量）（V2）

1. 本章节对图像内容审核（批量）AK/SK方式使用SDK进行示例说明。

示例代码中修改ImageBatchModerationReq的Urls配置待审核的图像url（仅支持url方式），配置完成后运行即可。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    moderation "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v2/region"
)
```

```
func main() {
    // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中
    // 密文存放，使用时解密，确保安全；
    // 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
    // HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK。
    ak := os.Getenv("HUAWEICLOUD_SDK_AK")
    sk := os.Getenv("HUAWEICLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := moderation.NewModerationClient(
        moderation.ModerationClientBuilder().
            WithRegion(region.ValueOf("xxx")). //把xxx替换成服务所在的区域，例如北京四：cn-north-4。
            WithCredential(auth).
            Build())

    request := &model.RunImageBatchModerationRequest{}
    var listCategoriesbody = []model.ImageBatchModerationReqCategories{
        model.GetImageBatchModerationReqCategoriesEnum().TERRORISM,
        model.GetImageBatchModerationReqCategoriesEnum().PORN,
        model.GetImageBatchModerationReqCategoriesEnum().AD,
    }
    var listUrlsbody = []string{
        "https://XXX.jpg", //此处替换为公网可以访问的图片地址
    }
    thresholdImageBatchModerationReq:= float64(0)
    request.Body = &model.ImageBatchModerationReq{
        Threshold: &thresholdImageBatchModerationReq,
        Categories: &listCategoriesbody,
        Urls: listUrlsbody,
    }
    response, err := client.RunImageBatchModeration(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

2. 审核结果输出到控制台表示执行成功，[图像内容审核批量任务](#)提交的结果输出到控制台。

```
RunImageBatchModerationResponse {"result":[{"url":"https://sdk-obs-source-save.obs.cn-north-4.myhuaweicloud.com/terrorism.jpg","suggestion":"pass","detail":{"porn":[{"confidence":0.9883,"label":"normal"},{"confidence":0.0039,"label":"porn"}, {"confidence":0.0078,"label":"sexy"}],"terrorism":[{"confidence":0.0002,"label":"terrorist"}, {"confidence":0.0002,"label":"fascist"}, {"confidence":0.0001,"label":"cult"}, {"confidence":0.0001,"label":"kidnap"}, {"confidence":0.0006,"label":"riot"}, {"confidence":0.0001,"label":"parade"}, {"confidence":0.0002,"label":"corpse"}, {"confidence":0.0001,"label":"bloody"}, {"confidence":0.0001,"label":"knife"}, {"confidence":0.0001,"label":"gun"}, {"confidence":0.0002,"label":"military_weapon"}, {"confidence":0,"label":"special_characters"}, {"confidence":0.9994,"label":"normal"}],"ad": [{"confidence":0,"label":"ad"}, {"confidence":1,"label":"normal"}],"category_suggestions":{"ad":"pass", "porn":"pass","terrorism":"pass"}}}]}
```

Process exiting with code: 0

9.3.2.4 内容审核-图像（异步批量）（V2）

9.3.2.4.1 提交任务

1. 本章节对图像内容审核（异步批量）-提交任务AK/SK方式使用SDK进行示例说明。
示例代码中可以修改TaskSumbitReq的Urls配置待审核的图像url（仅支持url方式），配置完成后运行即可。


```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    moderation "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v2/region"
)

func main() {
    // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中
    // 密文存放，使用时解密，确保安全；
    // 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
    // HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK。
    ak := os.Getenv("HUAWEICLOUD_SDK_AK")
    sk := os.Getenv("HUAWEICLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := moderation.NewModerationClient(
        moderation.ModerationClientBuilder().
            WithRegion(region.ValueOf("xxx")). //把xxx替换成服务所在的区域，例如北京四：cn-north-4。
            WithCredential(auth).
            Build())

    request := &model.RunTaskSubmitRequest{}
    var listCategoriesbody = []model.TaskSubmitReqCategories{
        model.GetTaskSubmitReqCategoriesEnum().PORN,
        model.GetTaskSubmitReqCategoriesEnum().TERRORISM,
    }
    var listUrlsbody = []string{
        "https://XXX.jpg", //此处替换为公网可以访问的图片地址
    }
    request.Body = &model.TaskSubmitReq{
        Categories: &listCategoriesbody,
        Urls: listUrlsbody,
    }
    response, err := client.RunTaskSubmit(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

2. 提交任务结果输出到控制台表示执行成功，异步批量任务提交的结果输出到控制台。

```
RunTaskSubmitResponse {"result":{"job_id":"xxx"}}
Process exiting with code: 0
```

9.3.2.4.2 查询任务

1. 本章节对图像内容审核异步批量-查询任务AK/SK方式使用SDK进行示例说明。示例代码中需要配置使用RunCheckResultRequest的JobId配置审核任务的id，配置完成后运行即可。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    moderation "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v2/region"
)

func main() {
    // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中
    // 密文存放，使用时解密，确保安全；
    // 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
    // HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK。
    ak := os.Getenv("HUAWEICLOUD_SDK_AK")
    sk := os.Getenv("HUAWEICLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := moderation.NewModerationClient(
        moderation.ModerationClientBuilder().
            WithRegion(region.ValueOf("xxx")). //把xxx替换成服务所在的区域，例如北京四：cn-north-4。
            WithCredential(auth).
            Build())

    request := &model.RunCheckResultRequest{}
    var jobidbody = []string{
        "xxx", //此处替换为任务id
    }
    request.Body = &model.RunCheckResultReq{
        JobId: &jobidbody,
    }
    response, err := client.RunCheckResult(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

```
func main() {
    // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中
    // 密文存放，使用时解密，确保安全；
    // 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
    // HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK。
    ak := os.Getenv("HUAWEICLOUD_SDK_AK")
    sk := os.Getenv("HUAWEICLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := moderation.NewModerationClient(
        moderation.ModerationClientBuilder().
            WithRegion(region.ValueOf("xxx")). //把xxx替换成服务所在的区域，例如北京四：cn-north-4。
            WithCredential(auth).
            Build())

    request := &model.RunCheckResultRequest{
        request.JobId = "70e952b5-5fa3-4d18-9ce4-75bba7226d7c"
    }
    response, err := client.RunCheckResult(request)
    if err == nil {
        fmt.Printf("%v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

2. 审核结果输出到控制台表示执行成功，异步批量[查询任务](#)的结果输出到控制台。

```
RunCheckResultResponse {"result":
{"job_id":"xxx","status":"finish","create_time":"2021-08-05T03:44:37Z","update_time":"2021-08-05T03:44:37Z","items":[{"url":"https://sdk-obs-source-save.obs.cn-north-4.myhuaweicloud.com/terrorism.jpg","suggestion":"pass","detail":{"porn":{"confidence":0.9883,"label":"normal"},{"confidence":0.0039,"label":"porn"},{"confidence":0.0078,"label":"sexy"},"terrorism":[{"confidence":0.0002,"label":"terrorist"},{"confidence":0.0002,"label":"fascist"},{"confidence":0.0001,"label":"cult"},{"confidence":0.0001,"label":"kidnap"},{"confidence":0.0006,"label":"riot"},{"confidence":0.0001,"label":"parade"},{"confidence":0.0002,"label":"corpse"},{"confidence":0.0001,"label":"bloody"},{"confidence":0.0001,"label":"knife"},{"confidence":0.0001,"label":"gun"},{"confidence":0.0002,"label":"military_weapon"},{"confidence":0,"label":"special_characters"},{"confidence":0.9994,"label":"normal"}]},category_suggestions":{"porn":"pass","terrorism":"pass"}}}]
Process exiting with code: 0
```

9.3.3 音频审核

9.3.3.1 创建音频内容审核作业

1. 本章节对创建音频审核AK/SK方式使用SDK进行示例说明。

示例代码中可以使用RunCreateAudioModerationJobRequest类的withUrls配置待审核的音频url（仅支持url方式），配置完成后运行即可。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    moderation "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v3/region"
)

func main() {
    // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中
    // 密文存放，使用时解密，确保安全；
    // 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变
```

```
量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK。
ak := os.Getenv("HUAWEICLOUD_SDK_AK")
sk := os.Getenv("HUAWEICLOUD_SDK_SK")

auth := basic.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    Build()

client := moderation.NewModerationClient(
    moderation.ModerationClientBuilder().
        WithRegion(region.ValueOf("xxx")). //把xxx替换成服务所在的区域，例如北京四：cn-north-4。
        WithCredential(auth).
        Build())

request := &model.RunCreateAudioModerationJobRequest{}
var listCategoriesbody = []model.AudioCreateRequestCategories{
    model.GetAudioCreateRequestCategoriesEnum().PORN,
}
databody := &model.AudioInputBody{
    Url: "",
}
request.Body = &model.AudioCreateRequest{
    Categories: &listCategoriesbody,
    EventType: model.GetAudioCreateRequestEventTypeEnum().DEFAULT,
    Data: databody,
}
response, err := client.RunCreateAudioModerationJob(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

2. 控制台输出创建音频内容审核提交的结果，[创建音频内容审核结果输出到控制台](#)。

```
RunCreateAudioModerationJobResponse{"request_id":"0b111ce94ec3d563a40d4e51a096d996",
"job_id": "xxx"}
```

9.3.3.2 查询音频内容审核作业

1. 本章节对查询音频审核AK/SK方式使用SDK进行示例说明。

示例代码中需要配置使用RunQueryAudioModerationJobRequest类的withJobId配置审核任务的id，配置完成后运行即可。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    moderation "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v3/region"
)

func main() {
    // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中
    密文存放，使用时解密，确保安全；
    // 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
    量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK。
    ak := os.Getenv("HUAWEICLOUD_SDK_AK")
    sk := os.Getenv("HUAWEICLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()
}
```

```
client := moderation.NewModerationClient(
    moderation.ModerationClientBuilder().
        WithRegion(region.ValueOf("xxx")). //把xxx替换成服务所在的区域, 例如北京四: cn-north-4。
        WithCredential(auth).
        Build())

request := &model.RunQueryAudioModerationJobRequest{}
request.JobId = ""
response, err := client.RunQueryAudioModerationJob(request)
if err == nil {
    fmt.Printf("%v\n", response)
} else {
    fmt.Println(err)
}
}
```

2. 控制台输出查询音频内容审核的结果, [查询音频内容审核](#)结果输出到控制台。

```
RunQueryAudioModerationJobResponse {"job_id": "xxx", "status": "running", "request_params":
{"event_type": "default", "data": {"url": "https://***.wav"}, "categories": ["porn"]}, "create_time":
"2022-08-29T11:57:14.014Z", "update_time": "2022-08-29T11:57:14.014Z", "request_id":
"c16c9fdeb3c1b838d9cec08ada4b6b96"}
```

9.3.4 视频审核

9.3.4.1 创建视频内容审核作业

1. 本章节对创建视频审核AK/SK方式使用SDK进行示例说明。

示例代码中可以使用RunCreateVideoModerationJobRequest类的withUrls配置待审核的视频url（仅支持url方式），配置完成后运行即可。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    moderation "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v3/region"
)

func main() {
    // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险, 建议在配置文件或者环境变量中
    密文存放, 使用时解密, 确保安全;
    // 本示例以ak和sk保存在环境变量中来实现身份验证为例, 运行本示例前请先在本地环境中设置环境变
    量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK。
    ak := os.Getenv("HUAWEICLOUD_SDK_AK")
    sk := os.Getenv("HUAWEICLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := moderation.NewModerationClient(
        moderation.ModerationClientBuilder().
            WithRegion(region.ValueOf("xxx")). //把xxx替换成服务所在的区域, 例如北京四: cn-
            north-4。
            WithCredential(auth).
            Build())

    request := &model.RunCreateVideoModerationJobRequest{}
    var listAudioCategoriesbody = []model.VideoCreateRequestAudioCategories{
        model.GetVideoCreateRequestAudioCategoriesEnum().PORN,
    }
    var listImageCategoriesbody = []model.VideoCreateRequestImageCategories{
        model.GetVideoCreateRequestImageCategoriesEnum().PORN,
    }
}
```

```
}
frameIntervalData:= float32(5.5)
databody := &model.VideoCreateRequestData{
    Url: "http://xxx.xx.xxxx",
    FrameInterval: &frameIntervalData,
}
callbackVideoCreateRequest:= "http://xxx.xx.xxxx"
request.Body = &model.VideoCreateRequest{
    Callback: &callbackVideoCreateRequest,
    AudioCategories: &listAudioCategoriesbody,
    ImageCategories: listImageCategoriesbody,
    EventType: model.GetVideoCreateRequestEventTypeEnum().DEFAULT,
    Data: databody,
}
response, err := client.RunCreateVideoModerationJob(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

2. 控制台输出创建视频内容审核作业的结果，[创建视频内容审核结果输出到控制台](#)。

```
RunCreateVideoModerationJobResponse{"request_id":"dbbcb75d1a174bfa899a38be38fdd107",
"job_id": "xxx"}
```

9.3.4.2 查询视频内容审核作业

1. 本章节对查询视频审核AK/SK方式使用SDK进行示例说明。

示例代码中需要配置使用RunQueryVideoModerationJobRequest类的withJobId配置审核任务的id，配置完成后运行即可。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    moderation "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v3/region"
)

func main() {
    // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中
    // 密文存放，使用时解密，确保安全；
    // 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
    // HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK。
    ak := os.Getenv("HUAWEICLOUD_SDK_AK")
    sk := os.Getenv("HUAWEICLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := moderation.NewModerationClient(
        moderation.ModerationClientBuilder().
            WithRegion(region.ValueOf("xxx")). //把xxx替换成服务所在的区域，例如北京四：cn-north-4。
            WithCredential(auth).
            Build())

    request := &model.RunQueryVideoModerationJobRequest{}
    request.JobId = "xxxx" //此处需要输入您的jobID
    response, err := client.RunQueryVideoModerationJob(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
```

```
    fmt.Println(err)
  }
}
```

2. 控制台输出查询视频内容审核作业提交的结果，[查询视频内容审核](#)结果输出到控制台。

```
RunQueryVideoModerationJobResponse
{"request_id":"xxxx","job_id":"xxx","status":"succeeded","request_params":{"data":
{"url":"xxxx","frame_interval":5,"event_type":"default","image_categories":
["porn","terrorism","image_text"],"audio_categories":
["porn","ad","moan","abuse"],"callback":"xxx"},"create_time":"2022-07-30T08:57:11.011Z","update_time":
"2022-07-30T08:57:14.014Z","result":{"suggestion":"block","image_detail":
[{"suggestion":"block","category":"porn","ocr_text":"123","time":0,"detail":
[{"confidence":0.676015138626099,"category":"porn","suggestion":"block","label":"pornography"},
{"confidence":0.666015138626099,"category":"porn","suggestion":"review","label":"sexy_female"},
{"confidence":0.666015138626099,"category":"xxx","suggestion":"block","label":"leader"},
{"confidence":0.691423773765564,"category":"xxx","suggestion":"block","face_location":
{"top_left_x":12,"top_left_y":13,"bottom_right_x":200,"bottom_right_y":240,"label":"leader"},
{"confidence":0.691423773765564,"category":"porn","suggestion":"block","face_location":
{"top_left_x":12,"top_left_y":13,"bottom_right_x":100,"bottom_right_y":2000,"label":"leader"},
{"confidence":0.891423773765564,"category":"image_text","suggestion":"block","label":"qr_code"},
{"confidence":0.891423773765564,"category":"image_text","suggestion":"block","label":"qr_code","qr_content":
"xxxxxxxxxx","qr_location":
{"top_left_x":12,"top_left_y":13,"bottom_right_x":100,"bottom_right_y":2000}},
{"confidence":0.891423773765564,"category":"image_text","suggestion":"block","label":"porn","segments":
[{"segment":"xxx"},{"segment":"xx"}]}]}],"audio_detail":
[{"suggestion":"block","label":"porn","audio_text":"xxxxx","end_time":10,"start_time":0,"detail":
[{"confidence":0.999117187582349,"label":"porn","suggestion":"block","segments":
[{"segment":"xxx"}]},{confidence":0.885117187582349,"label":"porn","suggestion":"block"}]}]}
```

9.3.5 音频流审核

9.3.5.1 创建音频流内容审核作业

1. 本章节对创建音频流审核AK/SK方式使用SDK进行示例说明。

示例代码中可以使用AudioStreamCreateRequestData类的withUrls配置待审核的音频流url（仅支持url方式），配置完成后运行即可。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    moderation "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v3/region"
)

func main() {
    // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中
    // 密文存放，使用时解密，确保安全；
    // 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
    // HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK。
    ak := os.Getenv("HUAWEICLOUD_SDK_AK")
    sk := os.Getenv("HUAWEICLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := moderation.NewModerationClient(
        moderation.ModerationClientBuilder().
            WithRegion(region.ValueOf("xxx")). //把xxx替换成服务所在的区域，例如北京四：cn-north-4。
            WithCredential(auth).
            Build())
```

```
request := &model.RunCreateAudioStreamModerationJobRequest{}
var listCategoriesbody = []string{
    "ad",
}
databody := &model.AudioStreamCreateRequestData{
    Url: "http://xxx",
}
request.Body = &model.AudioStreamCreateRequest{
    Callback: "http://xxxxxx",
    Categories: listCategoriesbody,
    EventType: "default",
    Data: databody,
}
response, err := client.RunCreateAudioStreamModerationJob(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

2. 控制台输出创建音频流内容审核作业的结果。

```
RunCreateAudioStreamModerationJobResponse {"request_id":
"303fbd70c9813a2476bad125fa436f00", "job_id":
"11690387200_ffa1f5bb9a58451b86816gdfc654610e"}
```

9.3.5.2 关闭音频流内容审核作业

1. 本章节对关闭音频流审核AK/SK方式使用SDK进行示例说明。

示例代码中需要配置使用RunCloseAudioStreamModerationJobRequest类的withJobId配置审核任务的id，配置完成后运行即可。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    moderation "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v3/region"
)

func main() {
    // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中
    // 密文存放，使用时解密，确保安全；
    // 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
    // HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK。
    ak := os.Getenv("HUAWEICLOUD_SDK_AK")
    sk := os.Getenv("HUAWEICLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := moderation.NewModerationClient(
        moderation.ModerationClientBuilder().
            WithRegion(region.ValueOf("xxx")). //把xxx替换成服务所在的区域，例如北京四：cn-north-4。
            WithCredential(auth).
            Build())

    request := &model.RunCloseAudioStreamModerationJobRequest{}
    request.JobId = "xxxxx"
    response, err := client.RunCloseAudioStreamModerationJob(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

```
}  
}
```

2. 控制台输出关闭音频流内容审核作业提交的结果。

```
RunCloseAudioStreamModerationJobResponse {"result": "success"}
```

9.3.6 视频流审核

9.3.6.1 创建视频流内容审核作业

1. 本章节对创建视频流审核AK/SK方式使用SDK进行示例说明。

示例代码中可以使用VideoStreamCreateRequestData类的withUrls配置待审核的视频流url（仅支持url方式），配置完成后运行即可。

```
package main  
  
import (  
    "fmt"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"  
    moderation "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v3"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v3/model"  
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v3/region"  
)  
  
func main() {  
    // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中  
    // 密文存放，使用时解密，确保安全；  
    // 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变  
    // 量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK。  
    ak := os.Getenv("HUAWEICLOUD_SDK_AK")  
    sk := os.Getenv("HUAWEICLOUD_SDK_SK")  
  
    auth := basic.NewCredentialsBuilder().  
        WithAk(ak).  
        WithSk(sk).  
        Build()  
  
    client := moderation.NewModerationClient(  
        moderation.ModerationClientBuilder().  
            WithRegion(region.ValueOf("xxx")). //把xxx替换成服务所在的区域，例如北京四：cn-north-4。  
            WithCredential(auth).  
            Build())  
  
    request := &model.RunCreateVideoStreamModerationJobRequest{}  
    var listImageCategoriesbody = []model.VideoStreamCreateRequestImageCategories{  
        model.GetVideoStreamCreateRequestImageCategoriesEnum().POLITICS,  
    }  
    databody := &model.VideoStreamCreateRequestData{  
        Url: "http://xxx",  
    }  
    request.Body = &model.VideoStreamCreateRequest{  
        Callback: "http://xxxxxx",  
        ImageCategories: listImageCategoriesbody,  
        EventType: model.GetVideoStreamCreateRequestEventTypeEnum().DEFAULT,  
        Data: databody,  
    }  
    response, err := client.RunCreateVideoStreamModerationJob(request)  
    if err == nil {  
        fmt.Printf("%+v\n", response)  
    } else {  
        fmt.Println(err)  
    }  
}
```

2. 控制台输出创建视频流内容审核作业的结果。

```
RunCreateVideoStreamModerationJobResponse {"request_id":  
"fa520dddb6b54af51c8d1e6d19898656", "job_id":  
"01690387200_09ad14f155bb47e9bf2718ace57c73ed"}
```


9.3.6.2 关闭视频流内容审核作业

1. 本章节对关闭视频流审核AK/SK方式使用SDK进行示例说明。

示例代码中需要配置使用RunCloseVideoStreamModerationJobRequest类的withJobId配置审核任务的id，配置完成后运行即可。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    moderation "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v3/region"
)

func main() {
    // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中
    // 密文存放，使用时解密，确保安全；
    // 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
    // HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK。
    ak := os.Getenv("HUAWEICLOUD_SDK_AK")
    sk := os.Getenv("HUAWEICLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := moderation.NewModerationClient(
        moderation.ModerationClientBuilder().
            WithRegion(region.ValueOf("xxx")). //把xxx替换成服务所在的区域，例如北京四：cn-north-4。
            WithCredential(auth).
            Build())

    request := &model.RunCloseVideoStreamModerationJobRequest{}
    request.JobId = "xxxxx"
    response, err := client.RunCloseVideoStreamModerationJob(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

2. 控制台输出关闭视频流内容审核作业提交的结果。

```
RunCloseVideoStreamModerationJobResponse {"result": "success"}
```

9.3.7 文档审核

9.3.7.1 创建文档内容审核作业

1. 本章节对创建文档内容审核作业AK/SK方式使用SDK进行示例说明。

示例代码中可以使用DocumentCreateRequestData类的withUrl配置待审核的文档url，配置完成后运行即可。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    moderation "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v3/region"
)
```

```
func main() {
    // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中
    // 密文存放，使用时解密，确保安全；
    // 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
    // HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK。
    ak := os.Getenv("HUAWEICLOUD_SDK_AK")
    sk := os.Getenv("HUAWEICLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := moderation.NewModerationClient(
        moderation.ModerationClientBuilder().
            WithRegion(region.ValueOf("xxx")). //把xxx替换成服务所在的区域，例如北京四：cn-north-4。
            WithCredential(auth).
            Build())

    request := &model.RunCreateDocumentModerationJobRequest{}
    databody := &model.DocumentCreateRequestData{
        Url: "http://xxx",
        Format: "pdf",
    }
    request.Body = &model.DocumentCreateRequest{
        EventType: "default",
        Data: databody,
    }
    response, err := client.RunCreateDocumentModerationJob(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

2. 控制台输出创建文档内容审核作业的结果。

```
RunCreateDocumentModerationJobResponse {"request_id": "c083957284ce1f2c0b910115d8232ec6",
"job_id": "01690387200_e3dfb78d9f20463f98aee25e6895203a"}
```

9.3.7.2 查询文档审核作业结果

1. 本章节对查询文档审核作业结果AK/SK方式使用SDK进行示例说明。

示例代码中需要配置使用RunQueryDocumentModerationJobRequest类的withJobId配置审核任务的id，配置完成后运行即可。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    moderation "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/moderation/v3/region"
)

func main() {
    // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中
    // 密文存放，使用时解密，确保安全；
    // 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
    // HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK。
    ak := os.Getenv("HUAWEICLOUD_SDK_AK")
    sk := os.Getenv("HUAWEICLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithIamEndpointOverride(iamEndpoint).
        Build()
}
```

```
client := moderation.NewModerationClient(  
    moderation.ModerationClientBuilder().  
        WithRegion(region.ValueOf("xxx")). //把xxx替换成服务所在的区域, 例如北京四: cn-north-4。  
        WithCredential(auth).  
        Build())  
  
request := &model.RunQueryDocumentModerationJobRequest{  
    request.JobId = "xxxx"  
}  
response, err := client.RunQueryDocumentModerationJob(request)  
if err == nil {  
    fmt.Printf("%+v\n", response)  
} else {  
    fmt.Println(err)  
}  
}
```

2. 控制台输出查询文档审核作业结果。

```
RunQueryDocumentModerationJobResponse {"job_id":  
"01693238400_cd7c8fd0a30b4c13b1fcaf9246c542c6", "status": "succeeded", "result": {"suggestion":  
"pass", "details": []}, "request_params": {"data": {"url": "http://xxxx", "format": "pdf"}, "event_type":  
"default"}, "create_time": "2023-08-29T16:33:50.050Z", "update_time": "2023-08-29T16:33:50.050Z",  
"request_id": "e4eeb972aeb92f4269e8dfbb59ba7e3b"}
```

10 使用 SDK (PHP)

10.1 PHP 开发环境配置

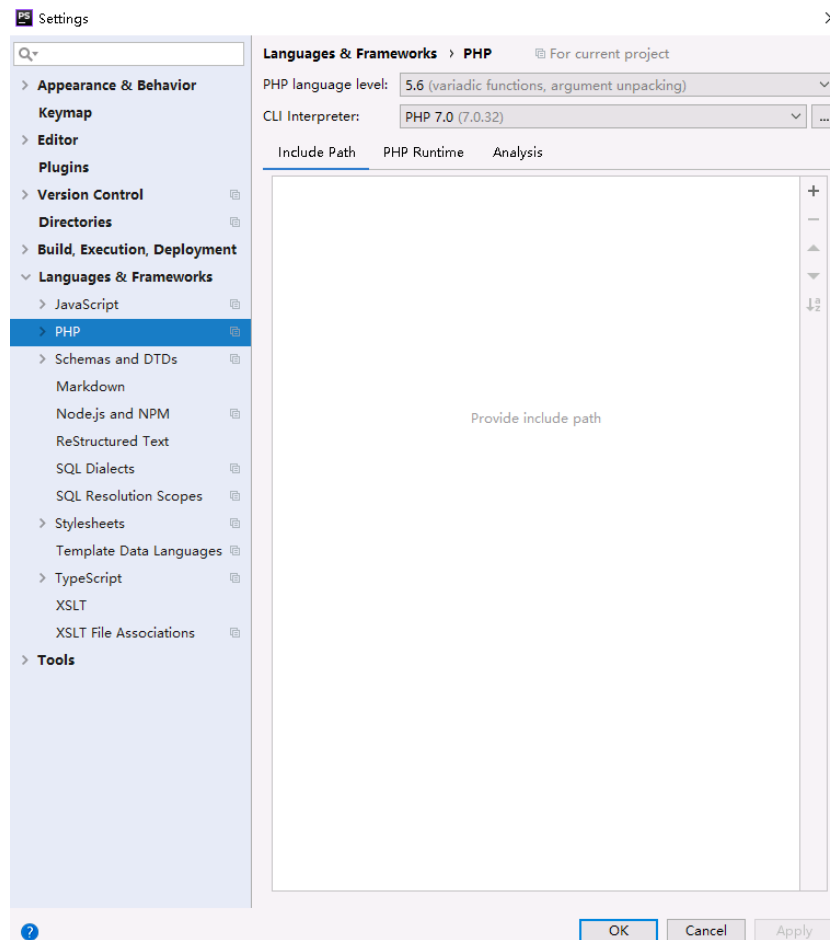
使用内容审核PHP版本SDK包，需要您配置PHP开发环境。

1. 从[PHP官网](#)下载并安装合适的PHP版本。内容审核PHP版本SDK包推荐使用php.7.x版本，如下以php 7.0.32 版本为例进行说明。

说明

- PHP安装目录应尽量避免中文和特殊字符。
 - windows环境下运行cmd命令**php -v**，显示找不到指定模块，需查看“php.ini”中“extension_dir”以及“include_path”的配置是否正确。
1. 从[PhoStorm官网](#)下载并安装最新社区版本。
 2. 在PhpStorm开发工具中配置PHP环境，在菜单依次选择“File > Settings > Languages & Frameworks > PHP”。
 3. 在页面上方选择您的PHP路径，如[图10-1](#)所示，然后单击页面下方“Apply”完成配置。

图 10-1 PhpStorm 配置 php 环境



10.2 PHP 环境 SDK 导入

PHP版本，内容审核服务推荐您使用 [Composer](#) 安装 SDK。Composer 是 Php 的依赖管理工具，允许您在项目中声明依赖关系并安装这些依赖：

```
// 安装 Composer  
curl -sS https://getcomposer.org/installer | php  
// 安装 PHP SDK  
composer require huaweicloud/huaweicloud-sdk-php
```

安装完毕后，您需要引入 Composer 的自动加载文件：

```
require 'path/to/vendor/autoload.php';
```

10.3 SDK 调用示例

10.3.1 文本审核

10.3.1.1 文本内容审核 (V3)

1. 本章节对文本内容审核AK/SK方式使用SDK进行示例说明。

示例代码中可以将TextDetectionDataReq的text配置待检测的文本信息，配置完成后运行即可。

项目ID的获取请参考[获取项目ID](#)。

```
<?php
namespace HuaweiCloud\SDK\Moderation\V3\Model;
require_once "vendor/autoload.php";
use HuaweiCloud\SDK\Core\Auth\BasicCredentials;
use HuaweiCloud\SDK\Core\Http\HttpConfig;
use HuaweiCloud\SDK\Core\Exceptions\ConnectionException;
use HuaweiCloud\SDK\Core\Exceptions\RequestTimeoutException;
use HuaweiCloud\SDK\Core\Exceptions\ServiceResponseException;
use HuaweiCloud\SDK\Moderation\V3\ModerationClient;
// 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中密文存放，使用时解密，确保安全；
// 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK、HUAWEICLOUD_SDK_PROJECT_ID。
$ak = getenv('HUAWEICLOUD_SDK_AK');
$sk = getenv('HUAWEICLOUD_SDK_SK');
$projectId = getenv('HUAWEICLOUD_SDK_PROJECT_ID');

$endpoint = "https://moderation.cn-north-4.myhuaweicloud.com"; //把cn-north-4替换成服务所在的区域。
$credentials = new BasicCredentials($ak,$sk,$projectId);
$config = HttpConfig::getDefaultConfig();
$config->setIgnoreSslVerification(true);

$client = ModerationClient::newBuilder(new ModerationClient)
->withHttpConfig($config)
->withEndpoint($endpoint)
->withCredentials($credentials)
->build();
$request = new RunTextModerationRequest();

$body = new TextDetectionReq();
$databody = new TextDetectionDataReq();
$databody->setText("text"); //此处输入待检测文本
$body->setData($databody);
$body->setEventType("comment");
$request->setBody($body);
try {
    $response = $client->RunTextModeration($request);
    echo "\n";
    echo $response;
} catch (ConnectionException $e) {
    $msg = $e->getMessage();
    echo "\n". $msg ."\n";
} catch (RequestTimeoutException $e) {
    $msg = $e->getMessage();
    echo "\n". $msg ."\n";
} catch (ServiceResponseException $e) {
    echo "\n";
    echo $e->getStatusCode(). "\n";
    echo $e->getErrorCode() . "\n";
    echo $e->getErrorMsg() . "\n";
}
```

2. 审核结果输出到控制台表示执行成功，[文本内容审核结果](#)输出到控制台。

```
{
  "request_id":"d81702cfd3599e103030c3d52d6f955f",
  "result":{
    "suggestion":"pass",
    "label":"normal",
    "details":[
    ]
  }
}
```

10.3.1.2 文本内容审核 (V2)

1. 本章节对文本内容审核AK/SK方式使用SDK进行示例说明。

示例代码中可以将TextDetectionItemsReq的text配置待检测的文本信息，配置完成后运行即可。

项目ID的获取请参考[获取项目ID](#)。

```
<?php
namespace HuaweiCloud\SDK\Moderation\V2\Model;
require_once "vendor/autoload.php";
use HuaweiCloud\SDK\Core\Auth\BasicCredentials;
use HuaweiCloud\SDK\Core\Http\HttpConfig;
use HuaweiCloud\SDK\Core\Exceptions\ConnectionException;
use HuaweiCloud\SDK\Core\Exceptions\RequestTimeoutException;
use HuaweiCloud\SDK\Core\Exceptions\ServiceResponseException;
use HuaweiCloud\SDK\Moderation\V2\ModerationClient;
// 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中密
文存放，使用时解密，确保安全；
// 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK、HUAWEICLOUD_SDK_PROJECT_ID。
$ak = getenv('HUAWEICLOUD_SDK_AK');
$sk = getenv('HUAWEICLOUD_SDK_SK');
$projectId = getenv('HUAWEICLOUD_SDK_PROJECT_ID');

$endpoint = "https://moderation.cn-north-4.myhuaweicloud.com"; //把cn-north-4替换成服务所在的区
域。
$credentials = new BasicCredentials($ak,$sk,$projectId);
$config = HttpConfig::getDefaultConfig();
$config->setIgnoreSslVerification(true);

$client = ModerationClient::newBuilder(new ModerationClient)
->withHttpConfig($config)
->withEndpoint($endpoint)
->withCredentials($credentials)
->build();
$request = new RunTextModerationRequest();

$body = new TextDetectionReq();
$listbodyItems = array();
array_push($listbodyItems,(new TextDetectionItemsReq())
->setText("text") //此处输入待检测文本
->setType("content")
);
$listbodyCategories = array();
array_push($listbodyCategories,"porn");
array_push($listbodyCategories,"ad");
array_push($listbodyCategories,"abuse");
array_push($listbodyCategories,"contraband");
array_push($listbodyCategories,"flood");
$body->setItems($listbodyItems);
$body->setCategories($listbodyCategories);
$request->setBody($body);
try {
    $response = $client->RunTextModeration($request);
    echo "\n";
    echo $response;
} catch (ConnectionException $e) {
    $msg = $e->getMessage();
    echo "\n". $msg ."\n";
} catch (RequestTimeoutException $e) {
    $msg = $e->getMessage();
    echo "\n". $msg ."\n";
} catch (ServiceResponseException $e) {
    echo "\n";
    echo $e->getHttpStatusCode(). "\n";
    echo $e->getErrorCode() . "\n";
    echo $e->getErrMsg() . "\n";
}
}
```

2. 审核结果输出到控制台表示执行成功，[文本内容审核结果](#)输出到控制台。

```
{
  "result":{
    "suggestion":"block",
    "detail":{
      "contraband":[
        "亚磷酸钾"
      ],
      "porn":[
        "luo聊"
      ]
    }
  }
}
```

10.3.2 图像审核

10.3.2.1 图像内容审核 (V3)

1. 本章节对图像内容审核AK/SK方式使用SDK进行示例说明。

示例代码中可以使用Url或Image配置图像信息(代表使用url或者图像的base64)，示例中以url为例配置完成后运行即可。

项目ID的获取请参考[获取项目ID](#)。

```
<?php
namespace HuaweiCloud\SDK\Moderation\V3\Model;
require_once "vendor/autoload.php";
use HuaweiCloud\SDK\Core\Auth\BasicCredentials;
use HuaweiCloud\SDK\Core\Http\HttpConfig;
use HuaweiCloud\SDK\Core\Exceptions\ConnectionException;
use HuaweiCloud\SDK\Core\Exceptions\RequestTimeoutException;
use HuaweiCloud\SDK\Core\Exceptions\ServiceResponseException;
use HuaweiCloud\SDK\Moderation\V3\ModerationClient;

// 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中密
文存放，使用时解密，确保安全；
// 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK、HUAWEICLOUD_SDK_PROJECT_ID。
$ak = getenv('HUAWEICLOUD_SDK_AK');
$sk = getenv('HUAWEICLOUD_SDK_SK');
$projectId = getenv('HUAWEICLOUD_SDK_PROJECT_ID');

$endpoint = "https://moderation.cn-north-4.myhuaweicloud.com"; //把cn-north-4替换成服务所在的区
域。
$credentials = new BasicCredentials($ak,$sk,$projectId);
$config = HttpConfig::getDefaultConfig();
$config->setIgnoreSslVerification(true);

$client = ModerationClient::newBuilder(new ModerationClient)
->withHttpConfig($config)
->withEndpoint($endpoint)
->withCredentials($credentials)
->build();
$request = new CheckImageModerationRequest();

$body = new ImageDetectionReq();
$listbodyCategories = array();
array_push($listbodyCategories,"porn");
$body->setUrl("http://www.xxx.xxx");
$body->setCategories($listbodyCategories);
$body->setEventType("head_image");
$request->setBody($body);
try {
    $response = $client->CheckImageModeration($request);
    echo "\n";
}
```



```
echo $response;
} catch (ConnectionException $e) {
    $msg = $e->getMessage();
    echo "\n". $msg ."\n";
} catch (RequestTimeoutException $e) {
    $msg = $e->getMessage();
    echo "\n". $msg ."\n";
} catch (ServiceResponseException $e) {
    echo "\n";
    echo $e->getHttpStatusCode(). "\n";
    echo $e->getErrorCode() . "\n";
    echo $e->getErrMsg() . "\n";
}
}
```

2. 审核结果输出到控制台表示执行成功，**图像内容审核结果**输出到控制台。
{ "request_id": "69b424920668cb15e6d25eef879a023b", "result": { "suggestion": "block", "category": "image_text", "details": [{ "suggestion": "block", "category": "image_text", "confidence": 1.0, "qr_location": { "top_left_x": 74, "top_left_y": 66, "bottom_right_x": 151, "bottom_right_y": 142}, "qr_content": "http://xxxxxx/xxxxx", "label": "qr_code"}, { "suggestion": "block", "category": "image_text", "confidence": 0.98779297, "label": "ad"}, { "suggestion": "block", "category": "image_text", "confidence": 1.0, "label": "qr_code"}], "ocr_text": "xxxxxxxxx" }

10.3.2.2 图像内容审核 (V2)

1. 本章节对图像内容审核AK/SK方式使用SDK进行示例说明。

示例代码中可以使用Url或Image配置图像信息(代表使用url或者图像的base64)，示例中以url为例配置完成后运行即可。

项目ID的获取请参考[获取项目ID](#)。

```
<?php
namespace HuaweiCloud\SDK\Moderation\V2\Model;
require_once "vendor/autoload.php";
use HuaweiCloud\SDK\Core\Auth\BasicCredentials;
use HuaweiCloud\SDK\Core\Http\HttpConfig;
use HuaweiCloud\SDK\Core\Exceptions\ConnectionException;
use HuaweiCloud\SDK\Core\Exceptions\RequestTimeoutException;
use HuaweiCloud\SDK\Core\Exceptions\ServiceResponseException;
use HuaweiCloud\SDK\Moderation\V2\ModerationClient;
// 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中密文存放，使用时解密，确保安全；
// 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK、HUAWEICLOUD_SDK_PROJECT_ID。
$ak = getenv('HUAWEICLOUD_SDK_AK');
$sk = getenv('HUAWEICLOUD_SDK_SK');
$projectId = getenv('HUAWEICLOUD_SDK_PROJECT_ID');

$endpoint = "https://moderation.cn-north-4.myhuaweicloud.com"; //把cn-north-4替换成服务所在的区域。
$credentials = new BasicCredentials($ak,$sk,$projectId);
$config = HttpConfig::getDefaultConfig();
$config->setIgnoreSslVerification(true);

$client = ModerationClient::newBuilder(new ModerationClient)
->withHttpConfig($config)
->withEndpoint($endpoint)
->withCredentials($credentials)
->build();
$request = new RunImageModerationRequest();

$body = new ImageDetectionReq();
$listbodyCategories = array();
array_push($listbodyCategories,"terrorism");
array_push($listbodyCategories,"porn");
$body->setCategories($listbodyCategories);
$body->setModerationRule("default");
$body->setUrl("https://XXX.jpg"); //此处替换为公网可以访问的图片地址
$request->setBody($body);
try {
```

```
$response = $client->RunImageModeration($request);
echo "\n";
echo $response;
} catch (ConnectionException $e) {
    $msg = $e->getMessage();
    echo "\n". $msg . "\n";
} catch (RequestTimeoutException $e) {
    $msg = $e->getMessage();
    echo "\n". $msg . "\n";
} catch (ServiceResponseException $e) {
    echo "\n";
    echo $e->getHttpStatusCode(). "\n";
    echo $e->getErrorCode() . "\n";
    echo $e->getErrMsg() . "\n";
}
}
```

2. 审核结果输出到控制台表示执行成功，[图像内容审核结果](#)输出到控制台。

```
{"result": {"detail": { "porn": [{"confidence": 0.9883, "label": "normal"}, {"confidence": 0.0039, "label": "porn"}, {"confidence": 0.0078, "label": "sexy"}], "terrorism": [{"confidence": 0.0002, "label": "terrorist"}, {"confidence": 0.0002, "label": "fascist"}, {"confidence": 0.0001, "label": "cult"}, {"confidence": 0.0002, "label": "kidnap"}, {"confidence": 0.0006, "label": "riot"}, {"confidence": 0.0001, "label": "parade"}, {"confidence": 0.0002, "label": "corpse"}, {"confidence": 0.0001, "label": "bloody"}, {"confidence": 0.0001, "label": "knife"}, {"confidence": 0.0001, "label": "gun"}, {"confidence": 0.0002, "label": "military_weapon"}, {"confidence": 0.0, "label": "special_characters"}, {"confidence": 0.9994, "label": "normal"}], "ad": [{"confidence": 0.0, "label": "ad"}, {"confidence": 1.0, "label": "normal"}]}, "suggestion": "pass", "category_suggestions": { "ad": "pass", "terrorism": "pass", "porn": "pass"}}
```

10.3.2.3 图像内容审核（批量）（V2）

1. 本章节对图像内容审核（批量）AK/SK方式使用SDK进行示例说明。

示例代码中修改ImageBatchModerationReq的Urls配置待审核的图像url（仅支持url方式），配置完成后运行即可。

项目ID的获取请参考[获取项目ID](#)。

```
<?php
namespace HuaweiCloud\SDK\Moderation\V2\Model;
require_once "vendor/autoload.php";
use HuaweiCloud\SDK\Core\Auth\BasicCredentials;
use HuaweiCloud\SDK\Core\Http\HttpConfig;
use HuaweiCloud\SDK\Core\Exceptions\ConnectionException;
use HuaweiCloud\SDK\Core\Exceptions\RequestTimeoutException;
use HuaweiCloud\SDK\Core\Exceptions\ServiceResponseException;
use HuaweiCloud\SDK\Moderation\V2\ModerationClient;
// 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中密文存放，使用时解密，确保安全；
// 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK、HUAWEICLOUD_SDK_PROJECT_ID。
$ak = getenv('HUAWEICLOUD_SDK_AK');
$sk = getenv('HUAWEICLOUD_SDK_SK');
$projectId = getenv('HUAWEICLOUD_SDK_PROJECT_ID');

$endpoint = "https://moderation.cn-north-4.myhuaweicloud.com"; //把cn-north-4替换成服务所在的区域。
$credentials = new BasicCredentials($ak,$sk,$projectId);
$config = HttpConfig::getDefaultConfig();
$config->setIgnoreSslVerification(true);

$client = ModerationClient::newBuilder(new ModerationClient)
    ->withHttpConfig($config)
    ->withEndpoint($endpoint)
    ->withCredentials($credentials)
    ->build();
$request = new RunImageBatchModerationRequest();

$body = new ImageBatchModerationReq();
$listbodyCategories = array();
array_push($listbodyCategories,"terrorism");
```

```
array_push($listbodyCategories,"porn");
array_push($listbodyCategories,"ad");
$listbodyUrls = array();
array_push($listbodyUrls,"https://XXX.jpg"); //此处替换为公网可以访问的图片地址
$body->setThreshold(0);
$body->setCategories($listbodyCategories);
$body->setUrls($listbodyUrls);
$request->setBody($body);
try {
    $response = $client->RunImageBatchModeration($request);
    echo "\n";
    echo $response;
} catch (ConnectionException $e) {
    $msg = $e->getMessage();
    echo "\n". $msg ."\n";
} catch (RequestTimeoutException $e) {
    $msg = $e->getMessage();
    echo "\n". $msg ."\n";
} catch (ServiceResponseException $e) {
    echo "\n";
    echo $e->getHttpStatusCode(). "\n";
    echo $e->getErrorCode() . "\n";
    echo $e->getErrorMsg() . "\n";
}
```

2. 审核结果输出到控制台表示执行成功，[图像内容审核批量任务](#)提交的结果输出到控制台。

```
{"httpStatusCode":200,"result":{"categorySuggestions":{"terrorism":"pass","porn":"pass"},"detail":{"porn":{"confidence":0.9883,"label":"normal"},{"confidence":0.0039,"label":"porn"}, {"confidence":0.0078,"label":"sexy"},"terrorism":{"confidence":2.0E-4,"label":"terrorist"}, {"confidence":2.0E-4,"label":"fascist"}, {"confidence":1.0E-4,"label":"cult"}, {"confidence":2.0E-4,"label":"kidnap"}, {"confidence":6.0E-4,"label":"riot"}, {"confidence":1.0E-4,"label":"parade"}, {"confidence":2.0E-4,"label":"corpse"}, {"confidence":1.0E-4,"label":"bloody"}, {"confidence":1.0E-4,"label":"knife"}, {"confidence":1.0E-4,"label":"gun"}, {"confidence":2.0E-4,"label":"military_weapon"}, {"confidence":0.0,"label":"special_characters"}, {"confidence":0.9994,"label":"normal"}}, "suggestion":"pass", "url":"https://sdk-obs-source-save.obs.cn-north-4.myhuaweicloud.com/terrorism.jpg"}}
```

10.3.2.4 内容审核-图像（异步批量）（V2）

10.3.2.4.1 提交任务

1. 本章节对图像内容审核（异步批量）-提交任务AK/SK方式使用SDK进行示例说明。

示例代码中可以修改TaskSubmitReq的Urls配置待审核的图像url（仅支持url方式），配置完成后运行即可。

项目ID的获取请参考[获取项目ID](#)。

```
<?php
namespace HuaweiCloud\SDK\Moderation\V2\Model;
require_once "vendor/autoload.php";
use HuaweiCloud\SDK\Core\Auth\BasicCredentials;
use HuaweiCloud\SDK\Core\Http\HttpConfig;
use HuaweiCloud\SDK\Core\Exceptions\ConnectionException;
use HuaweiCloud\SDK\Core\Exceptions\RequestTimeoutException;
use HuaweiCloud\SDK\Core\Exceptions\ServiceResponseException;
use HuaweiCloud\SDK\Moderation\V2\ModerationClient;
// 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中密文存放，使用时解密，确保安全；
// 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK、HUAWEICLOUD_SDK_PROJECT_ID。
$ak = getenv('HUAWEICLOUD_SDK_AK');
$sk = getenv('HUAWEICLOUD_SDK_SK');
$projectId = getenv('HUAWEICLOUD_SDK_PROJECT_ID');
```

```
$endpoint = "https://moderation.cn-north-4.myhuaweicloud.com"; //把cn-north-4替换成服务所在的区域。
$credentials = new BasicCredentials($ak,$sk,$projectId);
$config = HttpConfig::getDefaultConfig();
$config->setIgnoreSslVerification(true);

$client = ModerationClient::newBuilder(new ModerationClient)
->withHttpConfig($config)
->withEndpoint($endpoint)
->withCredentials($credentials)
->build();
$request = new RunTaskSubmitRequest();

$body = new TaskSubmitReq();
$listbodyCategories = array();
array_push($listbodyCategories,"porn");
array_push($listbodyCategories,"terrorism");
$listbodyUrls = array();
array_push($listbodyUrls,"https://XXX.jpg"); //此处替换为公网可以访问的图片地址
$body->setCategories($listbodyCategories);
$body->setUrls($listbodyUrls);
$request->setBody($body);
try {
    $response = $client->RunTaskSubmit($request);
    echo "\n";
    echo $response;
} catch (ConnectionException $e) {
    $msg = $e->getMessage();
    echo "\n". $msg ."\n";
} catch (RequestTimeoutException $e) {
    $msg = $e->getMessage();
    echo "\n". $msg ."\n";
} catch (ServiceResponseException $e) {
    echo "\n";
    echo $e->getStatusCode(). "\n";
    echo $e->getErrorCode() . "\n";
    echo $e->getErrMsg() . "\n";
}
}
```

2. 提交任务结果输出到控制台表示执行成功，异步批量[任务提交](#)的结果输出到控制台。

```
{"httpStatusCode":200,"result":{"jobId":"7f6a5881-7774-40cf-903b-4548914b55a3"}}
```

10.3.2.4.2 查询任务

1. 本章节对图像内容审核异步批量-查询任务AK/SK方式使用SDK进行示例说明。示例代码中需要配置使用RunCheckResultRequest的JobId配置审核任务的id，配置完成后运行即可。

项目ID的获取请参考[获取项目ID](#)。

```
<?php
namespace HuaweiCloud\SDK\Moderation\V2\Model;
require_once "vendor/autoload.php";
use HuaweiCloud\SDK\Core\Auth\BasicCredentials;
use HuaweiCloud\SDK\Core\Http\HttpConfig;
use HuaweiCloud\SDK\Core\Exceptions\ConnectionException;
use HuaweiCloud\SDK\Core\Exceptions\RequestTimeoutException;
use HuaweiCloud\SDK\Core\Exceptions\ServiceResponseException;
use HuaweiCloud\SDK\Moderation\V2\ModerationClient;
// 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中密文存放，使用时解密，确保安全；
// 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK、HUAWEICLOUD_SDK_PROJECT_ID。
$ak = getenv('HUAWEICLOUD_SDK_AK');
$sk = getenv('HUAWEICLOUD_SDK_SK');
$projectId = getenv('HUAWEICLOUD_SDK_PROJECT_ID');

$endpoint = "https://moderation.cn-north-4.myhuaweicloud.com"; //把cn-north-4替换成服务所在的区
```

```
域。
$credentials = new BasicCredentials($ak,$sk,$projectId);
$config = HttpConfig::getDefaultConfig();
$config->setIgnoreSslVerification(true);

$client = ModerationClient::newBuilder(new ModerationClient)
->withHttpConfig($config)
->withEndpoint($endpoint)
->withCredentials($credentials)
->build();
$request = new RunCheckResultRequest();
$request->setJobId("321cb5b6-3ad3-461a-b258-7ba7f530b91a");

try {
    $response = $client->RunCheckResult($request);
    echo "\n";
    echo $response;
} catch (ConnectionException $e) {
    $msg = $e->getMessage();
    echo "\n". $msg ."\n";
} catch (RequestTimeoutException $e) {
    $msg = $e->getMessage();
    echo "\n". $msg ."\n";
} catch (ServiceResponseException $e) {
    echo "\n";
    echo $e->getHttpStatusCode(). "\n";
    echo $e->getErrorCode() . "\n";
    echo $e->getErrMsg() . "\n";
}
}
```

2. 审核结果输出到控制台表示执行成功，异步批量[查询任务](#)的结果输出到控制台。

```
{
  "httpStatusCode":200,"result":{"createTime":"2021-08-13T03:46:30Z","items":[{"categorySuggestions":
{
  "terrorism":"pass","porn":"pass"},"detail":{"porn":[{"confidence":0.9883,"label":"normal"},
{"confidence":0.0039,"label":"porn"},{"confidence":0.0078,"label":"sexy"}],"terrorism":
[{"confidence":2.0E-4,"label":"terrorist"},{"confidence":2.0E-4,"label":"fascist"},
{"confidence":1.0E-4,"label":"cult"},{"confidence":2.0E-4,"confidence":1.0E-4},
{"confidence":1.0E-4,"label":"kidnap"},{"confidence":6.0E-4,"label":"riot"},
{"confidence":1.0E-4,"label":"parade"},{"confidence":2.0E-4,"label":"corpse"},
{"confidence":1.0E-4,"label":"bloody"},{"confidence":1.0E-4,"label":"knife"},
{"confidence":1.0E-4,"label":"gun"},{"confidence":2.0E-4,"label":"military_weapon"},
{"confidence":0.0,"label":"special_characters"},
{"confidence":0.9994,"label":"normal"}]},"suggestion":"pass","url":"https://sdk-obs-source-save.obs.cn-
north-4.myhuaweicloud.com/
terrorism.jpg"},"jobId":"7f6a5881-7774-40cf-903b-4548914b55a3","status":"finish","updateTime":"202
1-08-13T03:46:30Z"}]}
```

10.3.3 音频审核

10.3.3.1 创建音频内容审核作业

1. 本章节对创建音频审核AK/SK方式使用SDK进行示例说明。

示例代码中可以使用RunCreateAudioModerationJobRequest类的withUrls配置待审核的音频url（仅支持url方式），配置完成后运行即可。

项目ID的获取请参考[获取项目ID](#)。

```
<?php
namespace HuaweiCloud\SDK\Moderation\V3\Model;
require_once "vendor/autoload.php";
use HuaweiCloud\SDK\Core\Auth\BasicCredentials;
use HuaweiCloud\SDK\Core\Http\HttpConfig;
use HuaweiCloud\SDK\Core\Exceptions\ConnectionException;
use HuaweiCloud\SDK\Core\Exceptions\RequestTimeoutException;
use HuaweiCloud\SDK\Core\Exceptions\ServiceResponseException;
use HuaweiCloud\SDK\Moderation\V3\ModerationClient;
// 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中密
```

```
文存放, 使用时解密, 确保安全;
// 本示例以ak和sk保存在环境变量中来实现身份验证为例, 运行本示例前请先在本地环境中设置环境变量
HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK、HUAWEICLOUD_SDK_PROJECT_ID。
$ak = getenv('HUAWEICLOUD_SDK_AK');
$sk = getenv('HUAWEICLOUD_SDK_SK');
$projectId = getenv('HUAWEICLOUD_SDK_PROJECT_ID');

$endpoint = "https://moderation.cn-north-4.myhuaweicloud.com"; //把cn-north-4替换成服务所在的区域。
$credentials = new BasicCredentials($ak,$sk,$projectId);
$config = HttpConfig::getDefaultConfig();
$config->setIgnoreSslVerification(true);

$client = ModerationClient::newBuilder(new ModerationClient)
->withHttpConfig($config)
->withEndpoint($endpoint)
->withCredentials($credentials)
->build();
$request = new RunCreateAudioModerationJobRequest();

$body = new AudioCreateRequest();
$listbodyCategories = array();
array_push($listbodyCategories,"porn");
$databody = new AudioInputBody();
$databody->setUrl("https://**.wav");
$body->setCategories($listbodyCategories);
$body->setEventType("default");
$body->setData($databody);
$request->setBody($body);
try {
    $response = $client->RunCreateAudioModerationJob($request);
    echo "\n";
    echo $response;
} catch (ConnectionException $e) {
    $msg = $e->getMessage();
    echo "\n". $msg ."\n";
} catch (RequestTimeoutException $e) {
    $msg = $e->getMessage();
    echo "\n". $msg ."\n";
} catch (ServiceResponseException $e) {
    echo "\n";
    echo $e->getStatusCode(). "\n";
    echo $e->getErrorCode() . "\n";
    echo $e->getErrMsg() . "\n";
}
}
```

- 控制台输出创建音频内容审核提交的结果, [创建音频内容审核](#)结果输出到控制台。

```
{"request_id": "0b111ce94ec3d563a40d4e51a096d996", "job_id": "xxx"}
```

10.3.3.2 查询音频内容审核作业

- 本章节对查询音频审核AK/SK方式使用SDK进行示例说明。

示例代码中需要配置使用RunQueryAudioModerationJobRequest类的withJobId配置审核任务的id, 配置完成后运行即可。

项目ID的获取请参考[获取项目ID](#)。

```
<?php
namespace HuaweiCloud\SDK\Moderation\V3\Model;
require_once "vendor/autoload.php";
use HuaweiCloud\SDK\Core\Auth\BasicCredentials;
use HuaweiCloud\SDK\Core\Http\HttpConfig;
use HuaweiCloud\SDK\Core\Exceptions\ConnectionException;
use HuaweiCloud\SDK\Core\Exceptions\RequestTimeoutException;
use HuaweiCloud\SDK\Core\Exceptions\ServiceResponseException;
use HuaweiCloud\SDK\Moderation\V3\ModerationClient;
// 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险, 建议在配置文件或者环境变量中密文存放, 使用时解密, 确保安全;
```

```
// 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK、HUAWEICLOUD_SDK_PROJECT_ID。
$sak = getenv('HUAWEICLOUD_SDK_AK');
$sk = getenv('HUAWEICLOUD_SDK_SK');
$projectId = getenv('HUAWEICLOUD_SDK_PROJECT_ID');

$endpoint = "https://moderation.cn-north-4.myhuaweicloud.com"; //把cn-north-4替换成服务所在的区域。
$credentials = new BasicCredentials($sak,$sk,$projectId);
$config = HttpConfig::getDefaultConfig();
$config->setIgnoreSslVerification(true);

$client = ModerationClient::newBuilder(new ModerationClient)
->withHttpConfig($config)
->withEndpoint($endpoint)
->withCredentials($credentials)
->build();
$request = new RunQueryAudioModerationJobRequest();
$request->setJobId("01661702400_cd06d61c16a14565995fcbfb04877bd3");

try {
    $response = $client->RunQueryAudioModerationJob($request);
    echo "\n";
    echo $response;
} catch (ConnectionException $e) {
    $msg = $e->getMessage();
    echo "\n". $msg ."\n";
} catch (RequestTimeoutException $e) {
    $msg = $e->getMessage();
    echo "\n". $msg ."\n";
} catch (ServiceResponseException $e) {
    echo "\n";
    echo $e->getHttpStatusCode(). "\n";
    echo $e->getErrorCode() . "\n";
    echo $e->getErrorMsg() . "\n";
}
```

- 控制台输出查询音频内容审核的结果，[查询音频内容审核](#)结果输出到控制台。

```
{
  "job_id": "xxx",
  "status": "succeeded",
  "result": {
    "suggestion": "block",
    "details": [
      {
        "start_time": 0,
        "suggestion": "block",
        "end_time": 10,
        "label": "porn",
        "audio_text": "xxxx",
        "segments": [
          {
            "segment": "xxx",
            "segment": "xxx",
            "segment": "xxx"
          }
        ]
      },
      {
        "start_time": 30,
        "suggestion": "block",
        "end_time": 40,
        "label": "porn",
        "audio_text": "xxx",
        "segments": [
          {
            "segment": "xx"
          }
        ]
      }
    ],
    "audio_text": "xxxxxx",
    "request_params": {
      "event_type": "default",
      "data": {
        "url": "https://xxx.wav",
        "callback": "http://xxx",
        "categories": ["porn", "ad"],
        "create_time": "2022-06-20 11:05:29",
        "update_time": "2022-06-20 11:07:08",
        "request_id": "2419446b1fe14203f64e4018d12db3dd"
      }
    }
  }
}
```

10.3.4 视频审核

10.3.4.1 创建视频内容审核作业

- 本章节对创建视频审核AK/SK方式使用SDK进行示例说明。

示例代码中可以使用RunCreateVideoModerationJobRequest类的withUrls配置待审核的视频url（仅支持url方式），配置完成后运行即可。

项目ID的获取请参考[获取项目ID](#)。

```
<?php
namespace HuaweiCloud\SDK\Moderation\V3\Model;
require_once "vendor/autoload.php";
use HuaweiCloud\SDK\Core\Auth\BasicCredentials;
use HuaweiCloud\SDK\Core\Http\HttpConfig;
use HuaweiCloud\SDK\Core\Exceptions\ConnectionException;
use HuaweiCloud\SDK\Core\Exceptions\RequestTimeoutException;
use HuaweiCloud\SDK\Core\Exceptions\ServiceResponseException;
use HuaweiCloud\SDK\Moderation\V3\ModerationClient;
// 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中密文存放，使用时解密，确保安全；
```

```
// 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK、HUAWEICLOUD_SDK_PROJECT_ID。
$sak = getenv('HUAWEICLOUD_SDK_AK');
$sk = getenv('HUAWEICLOUD_SDK_SK');
$projectId = getenv('HUAWEICLOUD_SDK_PROJECT_ID');

$endpoint = "https://moderation.cn-north-4.myhuaweicloud.com"; //把cn-north-4替换成服务所在的区域。
$credentials = new BasicCredentials($sak,$sk,$projectId);
$config = HttpConfig::getDefaultConfig();
$config->setIgnoreSslVerification(true);

$client = ModerationClient::newBuilder(new ModerationClient)
->withHttpConfig($config)
->withEndpoint($endpoint)
->withCredentials($credentials)
->build();
$request = new RunCreateVideoModerationJobRequest();

$body = new VideoCreateRequest();
$listbodyAudioCategories = array();
array_push($listbodyAudioCategories,"porn");
$listbodyImageCategories = array();
array_push($listbodyImageCategories,"porn");
$databody = new VideoCreateRequestData();
$databody->setUrl("http://xxx.xx.xxx")
->setFrameInterval(5);
$body->setAudioCategories($listbodyAudioCategories);
$body->setImageCategories($listbodyImageCategories);
$body->setEventType("default");
$body->setData($databody);
$request->setBody($body);
try {
    $response = $client->RunCreateVideoModerationJob($request);
    echo "\n";
    echo $response;
} catch (ConnectionException $e) {
    $msg = $e->getMessage();
    echo "\n". $msg ."\n";
} catch (RequestTimeoutException $e) {
    $msg = $e->getMessage();
    echo "\n". $msg ."\n";
} catch (ServiceResponseException $e) {
    echo "\n";
    echo $e->getStatusCode(). "\n";
    echo $e->getErrorCode() . "\n";
    echo $e->getErrMsg() . "\n";
}
}
```

2. 控制台输出创建视频内容审核作业的结果，[创建视频内容审核结果输出到控制台](#)。

```
{"request_id": "dbbcb75d1a174bfa899a38be38fdd107", "job_id": "xxx"}
```

10.3.4.2 查询视频内容审核作业

1. 本章节对查询视频审核AK/SK方式使用SDK进行示例说明。

示例代码中需要配置使用RunQueryVideoModerationJobRequest类的withJobId配置审核任务的id，配置完成后运行即可。

项目ID的获取请参考[获取项目ID](#)。

```
<?php
namespace HuaweiCloud\SDK\Moderation\V3\Model;
require_once "vendor/autoload.php";
use HuaweiCloud\SDK\Core\Auth\BasicCredentials;
use HuaweiCloud\SDK\Core\Http\HttpConfig;
use HuaweiCloud\SDK\Core\Exceptions\ConnectionException;
use HuaweiCloud\SDK\Core\Exceptions\RequestTimeoutException;
use HuaweiCloud\SDK\Core\Exceptions\ServiceResponseException;
```



```
use HuaweiCloud\SDK\Moderation\V3\ModerationClient;
// 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险, 建议在配置文件或者环境变量中密文存放, 使用时解密, 确保安全;
// 本示例以ak和sk保存在环境变量中来实现身份验证为例, 运行本示例前请先在本地环境中设置环境变量 HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK、HUAWEICLOUD_SDK_PROJECT_ID。
$ak = getenv('HUAWEICLOUD_SDK_AK');
$sk = getenv('HUAWEICLOUD_SDK_SK');
$projectId = getenv('HUAWEICLOUD_SDK_PROJECT_ID');

$endpoint = "https://moderation.cn-north-4.myhuaweicloud.com"; //把cn-north-4替换成服务所在的区域。
$credentials = new BasicCredentials($ak,$sk,$projectId);
$config = HttpConfig::getDefaultConfig();
$config->setIgnoreSslVerification(true);

$client = ModerationClient::newBuilder(new ModerationClient)
->withHttpConfig($config)
->withEndpoint($endpoint)
->withCredentials($credentials)
->build();
$request = new RunQueryVideoModerationJobRequest();
$request->setJobId("xxxx"); //此处需要输入您的jobID

try {
    $response = $client->RunQueryVideoModerationJob($request);
    echo "\n";
    echo $response;
} catch (ConnectionException $e) {
    $msg = $e->getMessage();
    echo "\n". $msg ."\n";
} catch (RequestTimeoutException $e) {
    $msg = $e->getMessage();
    echo "\n". $msg ."\n";
} catch (ServiceResponseException $e) {
    echo "\n";
    echo $e->getHttpStatusCode(). "\n";
    echo $e->getErrorCode() . "\n";
    echo $e->getErrMsg() . "\n";
}
}
```

2. 控制台输出查询视频内容审核作业提交的结果, [查询视频内容审核](#)结果输出到控制台。

```
{
  "request_id": "xxxx", "job_id": "xxx", "status": "succeeded", "request_params": {
    "data": {
      "url": "xxxx", "frame_interval": 5, "event_type": "default", "image_categories": [
        "porn", "terrorism", "image_text", "audio_categories": [
          "porn", "ad", "moan", "abuse", "callback": "xxx", "create_time": "2022-07-30T08:57:11.011Z", "update_time": "2022-07-30T08:57:14.014Z", "result": {
            "suggestion": "block", "image_detail": [
              {
                "suggestion": "block", "category": "porn", "ocr_text": "123", "time": 0, "detail": [
                  {
                    "confidence": 0.676015138626099, "category": "porn", "suggestion": "block", "label": "pornography",
                    "confidence": 0.666015138626099, "category": "porn", "suggestion": "review", "label": "sexy_female",
                    "confidence": 0.666015138626099, "category": "xxx", "suggestion": "block", "label": "leader",
                    "confidence": 0.691423773765564, "category": "xxx", "suggestion": "block", "face_location": {
                      "top_left_x": 12, "top_left_y": 13, "bottom_right_x": 200, "bottom_right_y": 240, "label": "leader",
                      "confidence": 0.691423773765564, "category": "porn", "suggestion": "block", "face_location": {
                        "top_left_x": 12, "top_left_y": 13, "bottom_right_x": 100, "bottom_right_y": 2000, "label": "leader",
                        "confidence": 0.891423773765564, "category": "image_text", "suggestion": "block", "label": "qr_code",
                        "confidence": 0.891423773765564, "category": "image_text", "suggestion": "block", "label": "qr_code", "qr_content": "xxxxxxxx", "qr_location": {
                          "top_left_x": 12, "top_left_y": 13, "bottom_right_x": 100, "bottom_right_y": 2000,
                          "confidence": 0.891423773765564, "category": "image_text", "suggestion": "block", "label": "porn", "segments": [
                            {
                              "segment": "xxx", {"segment": "xx"}]
                            }
                          ], "audio_detail": [
                            {
                              "suggestion": "block", "label": "porn", "audio_text": "xxxx", "end_time": 10, "start_time": 0, "detail": [
                                {
                                  "confidence": 0.999117187582349, "label": "porn", "suggestion": "block", "segments": [
                                    {
                                      "segment": "xxx"}, {"confidence": 0.885117187582349, "label": "porn", "suggestion": "block"}]
                                }
                              ]
                            }
                          ]
                        }
                      ]
                    }
                  ]
                }
              ]
            }
          }
        ]
      }
    }
  }
}
```

10.3.5 音频流审核

10.3.5.1 创建音频流内容审核作业

1. 本章节对创建音频流审核AK/SK方式使用SDK进行示例说明。

示例代码中可以使用AudioStreamCreateRequestData类的withUrls配置待审核的音频流url（仅支持url方式），配置完成后运行即可。

项目ID的获取请参考[获取项目ID](#)。

```
<?php
namespace HuaweiCloud\SDK\Moderation\V3\Model;
require_once "vendor/autoload.php";
use HuaweiCloud\SDK\Core\Auth\BasicCredentials;
use HuaweiCloud\SDK\Core\Http\HttpConfig;
use HuaweiCloud\SDK\Core\Exceptions\ConnectionException;
use HuaweiCloud\SDK\Core\Exceptions\RequestTimeoutException;
use HuaweiCloud\SDK\Core\Exceptions\ServiceResponseException;
use HuaweiCloud\SDK\Moderation\V3\ModerationClient;
// 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中密
文存放，使用时解密，确保安全；
// 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK、HUAWEICLOUD_SDK_PROJECT_ID。
$ak = getenv('HUAWEICLOUD_SDK_AK');
$sk = getenv('HUAWEICLOUD_SDK_SK');
$projectId = getenv('HUAWEICLOUD_SDK_PROJECT_ID');

$endpoint = "https://moderation.cn-north-4.myhuaweicloud.com"; //把cn-north-4替换成服务所在的区
域。
$credentials = new BasicCredentials($ak,$sk,$projectId);
$config = HttpConfig::getDefaultConfig();
$config->setIgnoreSslVerification(true);

$client = ModerationClient::newBuilder(new ModerationClient)
->withHttpConfig($config)
->withEndpoint($endpoint)
->withCredentials($credentials)
->build();
$request = new RunCreateAudioStreamModerationJobRequest();

$body = new AudioStreamCreateRequest();
$listbodyCategories = array();
array_push($listbodyCategories,"ad");
$databody = new AudioStreamCreateRequestData();
$databody->setUrl("http://xxx");
$body->setCallback("http://xxxxxx");
$body->setCategories($listbodyCategories);
$body->setEventType("default");
$body->setData($databody);
$request->setBody($body);
try {
    $response = $client->RunCreateAudioStreamModerationJob($request);
    echo "\n";
    echo $response;
} catch (ConnectionException $e) {
    $msg = $e->getMessage();
    echo "\n". $msg ."\n";
} catch (RequestTimeoutException $e) {
    $msg = $e->getMessage();
    echo "\n". $msg ."\n";
} catch (ServiceResponseException $e) {
    echo "\n";
    echo $e->getStatusCode(). "\n";
    echo $e->getRequestId(). "\n";
    echo $e->getErrorCode(). "\n";
    echo $e->getErrMsg(). "\n";
}
}
```

2. 控制台输出创建音频流内容审核作业的结果。

```
{"request_id": "303fbd70c9813a2476bad125fa436f00", "job_id":
"11690387200_ffa1f5bb9a58451b86816gdfc654610e"}
```

10.3.5.2 关闭音频流内容审核作业

1. 本章节对关闭音频流审核AK/SK方式使用SDK进行示例说明。

示例代码中需要配置使用RunCloseAudioStreamModerationJobRequest类的withJobId配置审核任务的id，配置完成后运行即可。

项目ID的获取请参考[获取项目ID](#)。

```
<?php
namespace HuaweiCloud\SDK\Moderation\V3\Model;
require_once "vendor/autoload.php";
use HuaweiCloud\SDK\Core\Auth\BasicCredentials;
use HuaweiCloud\SDK\Core\Http\HttpConfig;
use HuaweiCloud\SDK\Core\Exceptions\ConnectionException;
use HuaweiCloud\SDK\Core\Exceptions\RequestTimeoutException;
use HuaweiCloud\SDK\Core\Exceptions\ServiceResponseException;
use HuaweiCloud\SDK\Moderation\V3\ModerationClient;
// 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中密
文存放，使用时解密，确保安全；
// 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK、HUAWEICLOUD_SDK_PROJECT_ID。
$sak = getenv('HUAWEICLOUD_SDK_AK');
$sk = getenv('HUAWEICLOUD_SDK_SK');
$projectId = getenv('HUAWEICLOUD_SDK_PROJECT_ID');

$endpoint = "https://moderation.cn-north-4.myhuaweicloud.com"; //把cn-north-4替换成服务所在的区
域。
$credentials = new BasicCredentials($sak,$sk,$projectId);
$config = HttpConfig::getDefaultConfig();
$config->setIgnoreSslVerification(true);

$client = ModerationClient::newBuilder(new ModerationClient)
->withHttpConfig($config)
->withEndpoint($endpoint)
->withCredentials($credentials)
->build();
$request = new RunCloseAudioStreamModerationJobRequest();
$request->setJobId("xxxx");

try {
    $response = $client->RunCloseAudioStreamModerationJob($request);
    echo "\n";
    echo $response;
} catch (ConnectionException $e) {
    $msg = $e->getMessage();
    echo "\n". $msg ."\n";
} catch (RequestTimeoutException $e) {
    $msg = $e->getMessage();
    echo "\n". $msg ."\n";
} catch (ServiceResponseException $e) {
    echo "\n";
    echo $e->getHttpStatusCode(). "\n";
    echo $e->getRequestId(). "\n";
    echo $e->getErrorCode(). "\n";
    echo $e->getErrMsg(). "\n";
}
```

2. 控制台输出关闭音频流内容审核作业提交的结果。

```
{"result": "success"}
```

10.3.6 视频流审核

10.3.6.1 创建视频流内容审核作业

1. 本章节对创建视频流审核AK/SK方式使用SDK进行示例说明。

示例代码中可以使用VideoStreamCreateRequestData类的withUrls配置待审核的视频流url（仅支持url方式），配置完成后运行即可。

项目ID的获取请参考[获取项目ID](#)。

```
<?php
namespace HuaweiCloud\SDK\Moderation\V3\Model;
require_once "vendor/autoload.php";
use HuaweiCloud\SDK\Core\Auth\BasicCredentials;
use HuaweiCloud\SDK\Core\Http\HttpConfig;
use HuaweiCloud\SDK\Core\Exceptions\ConnectionException;
use HuaweiCloud\SDK\Core\Exceptions\RequestTimeoutException;
use HuaweiCloud\SDK\Core\Exceptions\ServiceResponseException;
use HuaweiCloud\SDK\Moderation\V3\ModerationClient;
// 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中密文存放，使用时解密，确保安全；
// 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK、HUAWEICLOUD_SDK_PROJECT_ID。
$ak = getenv('HUAWEICLOUD_SDK_AK');
$sk = getenv('HUAWEICLOUD_SDK_SK');

$endpoint = "https://moderation.cn-north-4.myhuaweicloud.com"; //把cn-north-4替换成服务所在的区域。
$projectId = getenv('HUAWEICLOUD_SDK_PROJECT_ID');
$credentials = new BasicCredentials($ak,$sk,$projectId);
$config = HttpConfig::getDefaultConfig();
$config->setIgnoreSslVerification(true);

$client = ModerationClient::newBuilder(new ModerationClient)
->withHttpConfig($config)
->withEndpoint($endpoint)
->withCredentials($credentials)
->build();
$request = new RunCreateVideoStreamModerationJobRequest();

$body = new VideoStreamCreateRequest();
$listbodyImageCategories = array();
array_push($listbodyImageCategories,"politics");
$databody = new VideoStreamCreateRequestData();
$databody->setUrl("http://xxx");
$body->setCallback("http://xxxxxx");
$body->setImageCategories($listbodyImageCategories);
$body->setEventType("default");
$body->setData($databody);
$request->setBody($body);
try {
    $response = $client->RunCreateVideoStreamModerationJob($request);
    echo "\n";
    echo $response;
} catch (ConnectionException $e) {
    $msg = $e->getMessage();
    echo "\n". $msg ."\n";
} catch (RequestTimeoutException $e) {
    $msg = $e->getMessage();
    echo "\n". $msg ."\n";
} catch (ServiceResponseException $e) {
    echo "\n";
    echo $e->getHttpStatusCode(). "\n";
    echo $e->getRequestId(). "\n";
    echo $e->getErrorCode(). "\n";
    echo $e->getErrorMsg(). "\n";
}
}
```

2. 控制台输出创建视频流内容审核作业的结果。

```
{ "request_id": "fa520ddd6b54af51c8d1e6d19898656", "job_id": "01690387200_09ad14f155bb47e9bf2718ace57c73ed" }
```

10.3.6.2 关闭视频流内容审核作业

1. 本章节对关闭视频流审核AK/SK方式使用SDK进行示例说明。

示例代码中需要配置使用RunCloseVideoStreamModerationJobRequest类的withJobId配置审核任务的id，配置完成后运行即可。

项目ID的获取请参考[获取项目ID](#)。

```
<?php
namespace HuaweiCloud\SDK\Moderation\V3\Model;
require_once "vendor/autoload.php";
use HuaweiCloud\SDK\Core\Auth\BasicCredentials;
use HuaweiCloud\SDK\Core\Http\HttpConfig;
use HuaweiCloud\SDK\Core\Exceptions\ConnectionException;
use HuaweiCloud\SDK\Core\Exceptions\RequestTimeoutException;
use HuaweiCloud\SDK\Core\Exceptions\ServiceResponseException;
use HuaweiCloud\SDK\Moderation\V3\ModerationClient;
// 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中密
文存放，使用时解密，确保安全；
// 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK、HUAWEICLOUD_SDK_PROJECT_ID。
$ak = getenv('HUAWEICLOUD_SDK_AK');
$sk = getenv('HUAWEICLOUD_SDK_SK');
$projectId = getenv('HUAWEICLOUD_SDK_PROJECT_ID');

$endpoint = "https://moderation.cn-north-4.myhuaweicloud.com"; //把cn-north-4替换成服务所在的区
域。
$credentials = new BasicCredentials($ak,$sk,$projectId);
$config = HttpConfig::getDefaultConfig();
$config->setIgnoreSslVerification(true);

$client = ModerationClient::newBuilder(new ModerationClient)
->withHttpConfig($config)
->withEndpoint($endpoint)
->withCredentials($credentials)
->build();
$request = new RunCloseVideoStreamModerationJobRequest();
$request->setJobId("xxxx");

try {
    $response = $client->RunCloseVideoStreamModerationJob($request);
    echo "\n";
    echo $response;
} catch (ConnectionException $e) {
    $msg = $e->getMessage();
    echo "\n". $msg ."\n";
} catch (RequestTimeoutException $e) {
    $msg = $e->getMessage();
    echo "\n". $msg ."\n";
} catch (ServiceResponseException $e) {
    echo "\n";
    echo $e->getHttpStatusCode(). "\n";
    echo $e->getRequestId(). "\n";
    echo $e->getErrorCode(). "\n";
    echo $e->getErrMsg(). "\n";
}
}
```

2. 控制台输出关闭视频流内容审核作业提交的结果。

```
{
  "result": "success"
}
```

10.3.7 文档审核

10.3.7.1 创建文档内容审核作业

1. 本章节对创建文档内容审核作业AK/SK方式使用SDK进行示例说明。

示例代码中可以使用DocumentCreateRequestData类的withUrl配置待审核的文档url，配置完成后运行即可。

项目ID的获取请参考[获取项目ID](#)。

```
<?php
namespace HuaweiCloud\SDK\Moderation\V3\Model;
require_once "vendor/autoload.php";
use HuaweiCloud\SDK\Core\Auth\BasicCredentials;
use HuaweiCloud\SDK\Core\Http\HttpConfig;
use HuaweiCloud\SDK\Core\Exceptions\ConnectionException;
use HuaweiCloud\SDK\Core\Exceptions\RequestTimeoutException;
use HuaweiCloud\SDK\Core\Exceptions\ServiceResponseException;
use HuaweiCloud\SDK\Moderation\V3\ModerationClient;

// 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中密
文存放，使用时解密，确保安全；
// 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK、HUAWEICLOUD_SDK_PROJECT_ID。
$ak = getenv('HUAWEICLOUD_SDK_AK');
$sk = getenv('HUAWEICLOUD_SDK_SK');
$projectId = getenv('HUAWEICLOUD_SDK_PROJECT_ID');

$endpoint = "https://moderation.cn-north-4.myhuaweicloud.com"; //把cn-north-4替换成服务所在的区
域。
$credentials = new BasicCredentials($ak,$sk,$projectId);
$config = HttpConfig::getDefaultConfig();
$config->setIgnoreSslVerification(true);

$client = ModerationClient::newBuilder(new ModerationClient)
->withHttpConfig($config)
->withEndpoint($endpoint)
->withCredentials($credentials)
->build();
$request = new RunCreateDocumentModerationJobRequest();

$body = new DocumentCreateRequest();
$databody = new DocumentCreateRequestData();
$databody->setUrl("http://xxx")
->setFormat("pdf");
$body->setEventType("default");
$body->setData($databody);
$request->setBody($body);
try {
    $response = $client->RunCreateDocumentModerationJob($request);
    echo "\n";
    echo $response;
} catch (ConnectionException $e) {
    $msg = $e->getMessage();
    echo "\n". $msg ."\n";
} catch (RequestTimeoutException $e) {
    $msg = $e->getMessage();
    echo "\n". $msg ."\n";
} catch (ServiceResponseException $e) {
    echo "\n";
    echo $e->getHttpStatusCode(). "\n";
    echo $e->getRequestId(). "\n";
    echo $e->getErrorCode() . "\n";
    echo $e->getErrMsg() . "\n";
}
}
```

2. 控制台输出创建文档内容审核作业的结果。

```
{"request_id": "c083957284ce1f2c0b910115d8232ec6", "job_id":
"01690387200_e3dfb78d9f20463f98aee25e6895203a"}
```

10.3.7.2 查询文档审核作业结果

1. 本章节对查询文档审核作业结果AK/SK方式使用SDK进行示例说明。

示例代码中需要配置使用RunQueryDocumentModerationJobRequest类的withJobId配置审核任务的id，配置完成后运行即可。

项目ID的获取请参考[获取项目ID](#)。

```
<?php
namespace HuaweiCloud\SDK\Moderation\V3\Model;
require_once "vendor/autoload.php";
use HuaweiCloud\SDK\Core\Auth\BasicCredentials;
use HuaweiCloud\SDK\Core\Http\HttpConfig;
use HuaweiCloud\SDK\Core\Exceptions\ConnectionException;
use HuaweiCloud\SDK\Core\Exceptions\RequestTimeoutException;
use HuaweiCloud\SDK\Core\Exceptions\ServiceResponseException;
use HuaweiCloud\SDK\Moderation\V3\ModerationClient;

// 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中密文存放，使用时解密，确保安全；
// 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK、HUAWEICLOUD_SDK_PROJECT_ID。
$ak = getenv('HUAWEICLOUD_SDK_AK');
$sk = getenv('HUAWEICLOUD_SDK_SK');
$projectId = getenv('HUAWEICLOUD_SDK_PROJECT_ID');

$endpoint = "https://moderation.cn-north-4.myhuaweicloud.com"; //把cn-north-4替换成服务所在的区域。
$credentials = new BasicCredentials($ak,$sk,$projectId);
$config = HttpConfig::getDefaultConfig();
$config->setIgnoreSslVerification(true);

$client = ModerationClient::newBuilder(new ModerationClient)
->withHttpConfig($config)
->withEndpoint($endpoint)
->withCredentials($credentials)
->build();
$request = new RunQueryDocumentModerationJobRequest();
$request->setJobId("xxx");

try {
    $response = $client->RunQueryDocumentModerationJob($request);
    echo "\n";
    echo $response;
} catch (ConnectionException $e) {
    $msg = $e->getMessage();
    echo "\n". $msg ."\n";
} catch (RequestTimeoutException $e) {
    $msg = $e->getMessage();
    echo "\n". $msg ."\n";
} catch (ServiceResponseException $e) {
    echo "\n";
    echo $e->getStatusCode(). "\n";
    echo $e->getRequestId(). "\n";
    echo $e->getErrorCode() . "\n";
    echo $e->getErrMsg() . "\n";
}
}
```

2. 控制台输出查询文档审核作业结果。

```
{"job_id": "01693238400_cd7c8fd0a30b4c13b1fc9246c542c6", "status": "succeeded", "result": {"suggestion": "pass", "details": []}, "request_params": {"data": {"url": "http://xxxx", "format": "pdf"}, "event_type": "default"}, "create_time": "2023-08-29T16:33:50.050Z", "update_time": "2023-08-29T16:33:50.050Z", "request_id": "e4eeb972aeb92f4269e8dfbb59ba7e3b"}
```

11 使用 SDK (Node.js)

11.1 Node.js 开发环境配置

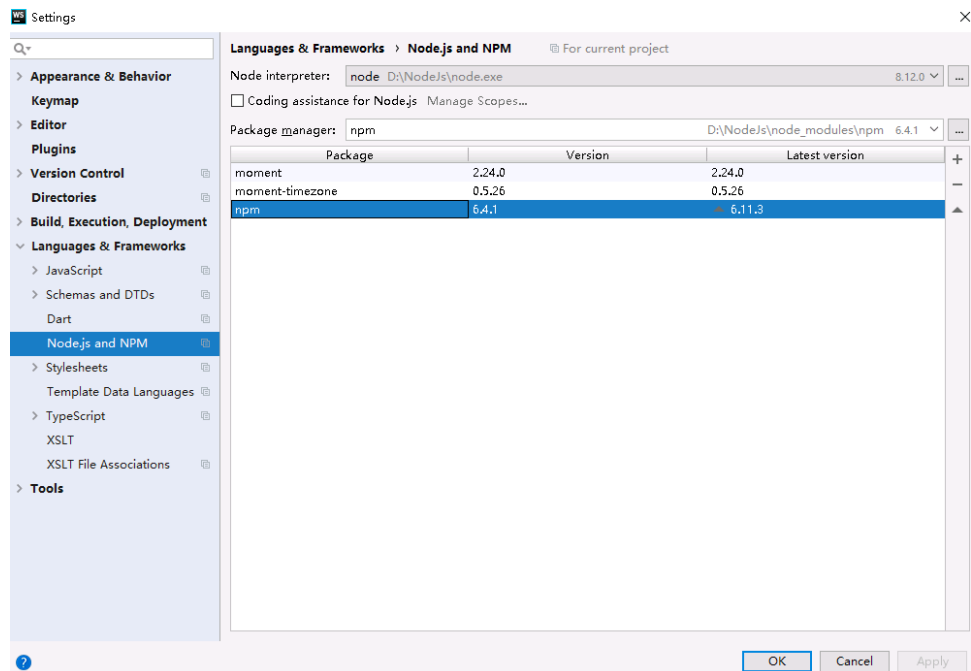
内容审核javascript版本SDK包依赖Node.js环境。

1. 从[Node.js官网](#)下载并安装推荐使用的版本。推荐使用Node.js 8.x及以上版本，如下以Node.js 8.12.0 版本为例进行说明。
2. 从[WebStorm官网](#)下载并安装WebStorm 最新版本。
3. 在WebStorm界面配置Node.js环境，在菜单依次选择“File > Settings > Languages & Frameworks > Node.js and NPM”。
4. 在页面上方选择您的“Node.js and NPM”安装路径，如[图11-1](#)所示。选择好目标“Node.js and NPM”之后单击页面下方“Apply”完成配置。

说明

内容审核javascript版本SDK需要依赖moment-timezone以及moment等模块，可使用npm install命令安装或单击[图11-1](#)中的“+”号通过工具类安装。

图 11-1 配置 Node.js 环境



11.2 Node.js 环境 SDK 导入

使用SDK前，您需要安装“@huaweicloud/huaweicloud-sdk-core”和“@huaweicloud/huaweicloud-sdk-moderation”。

推荐您使用npm安装SDK，在命令行输入以下命令：

```
npm install @huaweicloud/huaweicloud-sdk-core  
npm i @huaweicloud/huaweicloud-sdk-moderation
```

11.3 SDK 调用示例

11.3.1 文本审核

11.3.1.1 文本内容审核（V3）

1. 本章节对文本内容审核AK/SK方式使用SDK进行示例说明。

示例代码中可以将TextDetectionDataReq的text配置待检测的文本信息，配置完成后运行即可。

```
const core = require('@huaweicloud/huaweicloud-sdk-core');  
const moderation = require("@huaweicloud/huaweicloud-sdk-moderation");  
// 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中密  
文存放，使用时解密，确保安全；  
// 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量  
HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK、HUAWEICLOUD_SDK_PROJECT_ID。  
const ak = process.env.HUAWEICLOUD_SDK_AK;  
const sk = process.env.HUAWEICLOUD_SDK_SK;  
const projectId = process.env.HUAWEICLOUD_SDK_PROJECT_ID;  
  
const endpoint = "https://moderation.cn-north-4.myhuaweicloud.com"; //把cn-north-4替换成服务所在
```

```
的区域。  
const credentials = new core.BasicCredentials()  
    .withAk(ak)  
    .withSk(sk)  
    .withProjectId(project_id)  
  
const client = moderation.v3.ModerationClient.newBuilder()  
    .withCredential(credentials)  
    .withEndpoint(endpoint)  
    .build();  
const request = new moderation.v3.RunTextModerationRequest();  
const body = new moderation.v3.TextDetectionReq();  
const databody = new moderation.v3.TextDetectionDataReq();  
databody.withText("test");  
body.withData(databody);  
body.withEventType("comment");  
request.withBody(body);  
const result = client.runTextModeration(request);  
result.then(result => {  
    console.log("JSON.stringify(result)::" + JSON.stringify(result));  
}).catch(ex => {  
    console.log("exception:" + JSON.stringify(ex));  
});
```

2. 审核结果输出到控制台表示执行成功，[文本内容审核结果](#)输出到控制台。
JSON.stringify(result)::{"request_id": "d81702cfd3599e103030c3d52d6f955f", "result": {"suggestion": "pass", "label": "normal", "details": []}}

11.3.1.2 文本内容审核 (V2)

1. 本章节对文本内容审核AK/SK方式使用SDK进行示例说明。

示例代码中可以将TextDetectionItemsReq的text配置待检测的文本信息，配置完成后运行即可。

```
const core = require('@huaweicloud/huaweicloud-sdk-core');  
const moderation = require("@huaweicloud/huaweicloud-sdk-moderation");  
// 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中密  
文存放，使用时解密，确保安全；  
// 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量  
HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK、HUAWEICLOUD_SDK_PROJECT_ID。  
const ak = process.env.HUAWEICLOUD_SDK_AK;  
const sk = process.env.HUAWEICLOUD_SDK_SK;  
const projectId = process.env.HUAWEICLOUD_SDK_PROJECT_ID;  
  
const endpoint = "https://moderation.cn-north-4.myhuaweicloud.com"; //把cn-north-4替换成服务所在  
的区域。  
const credentials = new core.BasicCredentials()  
    .withAk(ak)  
    .withSk(sk)  
const client = moderation.ModerationClient.newBuilder()  
    .withCredential(credentials)  
    .withEndpoint(endpoint)  
    .build();  
const request = new moderation.RunTextModerationRequest();  
const body = new moderation.TextDetectionReq();  
const listbodyItems = new Array();  
listbodyItems.push(  
    new moderation.TextDetectionItemsReq()  
        .withText("text") //此处输入待检测文本  
        .withType("content")  
);  
const listbodyCategories = new Array();  
listbodyCategories.push("ad");  
listbodyCategories.push("abuse");  
listbodyCategories.push("porn");  
listbodyCategories.push("contraband");  
listbodyCategories.push("flood");  
body.withItems(listbodyItems);
```

```
body.withCategories(listbodyCategories);
request.withBody(body);
const result = client.runTextModeration(request);
result.then(result => {
  console.log("JSON.stringify(result)::" + JSON.stringify(result));
}).catch(ex => {
  console.log("exception:" + JSON.stringify(ex));
});
```

2. 审核结果输出到控制台表示执行成功，**文本内容审核结果**输出到控制台。
JSON.stringify(result)::{"result": {"suggestion": "block", "detail": {"contraband": ["亚砷酸钾"], "porn": ["luo聊"]}}}

11.3.2 图像审核

11.3.2.1 图像内容审核 (V3)

1. 本章节对图像内容审核AK/SK方式使用SDK进行示例说明。

示例代码中可以使用Url或Image配置图像信息(代表使用url或者图像的base64)，示例中以url为例配置完成后运行即可。

```
const core = require('@huaweicloud/huaweicloud-sdk-core');
const moderation = require("@huaweicloud/huaweicloud-sdk-moderation");
// 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中密文存放，使用时解密，确保安全；
// 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量 HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK、HUAWEICLOUD_SDK_PROJECT_ID。
const ak = process.env.HUAWEICLOUD_SDK_AK;
const sk = process.env.HUAWEICLOUD_SDK_SK;
const projectId = process.env.HUAWEICLOUD_SDK_PROJECT_ID;

const endpoint = "https://moderation.cn-north-4.myhuaweicloud.com"; //把cn-north-4替换成服务所在的区域。
const credentials = new core.BasicCredentials()
  .withAk(ak)
  .withSk(sk)
  .withProjectId(project_id)
const client = moderation.v3.ModerationClient.newBuilder()
  .withCredential(credentials)
  .withEndpoint(endpoint)
  .build();

const request = new moderation.v3.CheckImageModerationRequest();
const body = new moderation.v3.ImageDetectionReq();
const listbodyCategories = new Array();
listbodyCategories.push("porn");
body.withUrl("http://www.xxxx.xxx");
body.withCategories(listbodyCategories);
body.withEventType("head_image");
request.withBody(body);
const result = client.checkImageModeration(request);
result.then(result => {
  console.log("JSON.stringify(result)::" + JSON.stringify(result));
}).catch(ex => {
  console.log("exception:" + JSON.stringify(ex));
});
```

2. 审核结果输出到控制台表示执行成功，**图像内容审核结果**输出到控制台。
JSON.stringify(result)::{"request_id": "69b424920668cb15e6d25eef879a023b", "result": {"suggestion": "block", "category": "image_text", "details": [{"suggestion": "block", "category": "image_text", "confidence": 1.0, "qr_location": {"top_left_x": 74, "top_left_y": 66, "bottom_right_x": 151, "bottom_right_y": 142}, "qr_content": "http://xxxxx/xxxxx", "label": "qr_code"}, {"suggestion": "block", "category": "image_text", "confidence": 0.98779297, "label": "ad"}, {"suggestion": "block", "category": "image_text", "confidence": 1.0, "label": "qr_code"}], "ocr_text": "xxxxxxxxx"}}

11.3.2.2 图像内容审核 (V2)

1. 本章节对图像内容审核AK/SK方式使用SDK进行示例说明。

示例代码中可以使用Url或Image配置图像信息(代表使用url或者图像的base64)，示例中以url为例配置完成后运行即可。

```
const core = require('@huaweicloud/huaweicloud-sdk-core');
const moderation = require("@huaweicloud/huaweicloud-sdk-moderation");
// 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中密文存放，使用时解密，确保安全；
// 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK、HUAWEICLOUD_SDK_PROJECT_ID。
const ak = process.env.HUAWEICLOUD_SDK_AK;
const sk = process.env.HUAWEICLOUD_SDK_SK;
const projectId = process.env.HUAWEICLOUD_SDK_PROJECT_ID;

const endpoint = "https://moderation.cn-north-4.myhuaweicloud.com"; //把cn-north-4替换成服务所在的区域。
const credentials = new core.BasicCredentials()
    .withAk(ak)
    .withSk(sk)
const client = moderation.ModerationClient.newBuilder()
    .withCredential(credentials)
    .withEndpoint(endpoint)
    .build();
const request = new moderation.RunImageModerationRequest();
const body = new moderation.ImageDetectionReq();
const listbodyCategories = new Array();
listbodyCategories.push("terrorism");
listbodyCategories.push("porn");
body.withCategories(listbodyCategories);
body.withModerationRule("default");
body.withUrl("https://XXX.jpg"); //此处替换为公网可以访问的图片地址
request.withBody(body);
const result = client.runImageModeration(request);
result.then(result => {
    console.log("JSON.stringify(result)::" + JSON.stringify(result));
}).catch(ex => {
    console.log("exception:" + JSON.stringify(ex));
});
```

2. 审核结果输出到控制台表示执行成功，**图像内容审核结果**输出到控制台。

```
JSON.stringify(result)::{"result": {"detail": { "porn": [{"confidence": 0.9883, "label": "normal"}, {"confidence": 0.0039, "label": "porn"}, {"confidence": 0.0078, "label": "sexy"}], "terrorism": [{"confidence": 0.0002, "label": "terrorist"}, {"confidence": 0.0002, "label": "fascist"}, {"confidence": 0.0001, "label": "cult"}, {"confidence": 0.0002}, {"confidence": 0.0001, "label": "kidnap"}, {"confidence": 0.0006, "label": "riot"}, {"confidence": 0.0001, "label": "parade"}, {"confidence": 0.0002, "label": "corpse"}, {"confidence": 0.0001, "label": "bloody"}, {"confidence": 0.0001, "label": "knife"}, {"confidence": 0.0001, "label": "gun"}, {"confidence": 0.0002, "label": "military_weapon"}, {"confidence": 0.0, "label": "special_characters"}, {"confidence": 0.9994, "label": "normal"}], "ad": [{"confidence": 0.0, "label": "ad"}, {"confidence": 1.0, "label": "normal"}]}, "suggestion": "pass", "category_suggestions": { "ad": "pass", "terrorism": "pass", "porn": "pass"}}
```

11.3.2.3 图像内容审核 (批量) (V2)

1. 本章节对图像内容审核 (批量) AK/SK方式使用SDK进行示例说明。

示例代码中修改ImageBatchModerationReq的Urls配置待审核的图像url (仅支持url方式)，配置完成后运行即可。

```
const core = require('@huaweicloud/huaweicloud-sdk-core');
const moderation = require("@huaweicloud/huaweicloud-sdk-moderation");
// 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中密文存放，使用时解密，确保安全；
// 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK、HUAWEICLOUD_SDK_PROJECT_ID。
const ak = process.env.HUAWEICLOUD_SDK_AK;
const sk = process.env.HUAWEICLOUD_SDK_SK;
const projectId = process.env.HUAWEICLOUD_SDK_PROJECT_ID;
```

```
const endpoint = "https://moderation.cn-north-4.myhuaweicloud.com"; //把cn-north-4替换成服务所在
的区域。
const credentials = new core.BasicCredentials()
    .withAk(ak)
    .withSk(sk)
const client = moderation.ModerationClient.newBuilder()
    .withCredential(credentials)
    .withEndpoint(endpoint)
    .build();
const request = new moderation.RunImageBatchModerationRequest();
const body = new moderation.ImageBatchModerationReq();
const listbodyCategories = new Array();
listbodyCategories.push("terrorism");
listbodyCategories.push("porn");
listbodyCategories.push("ad");
const listbodyUrls = new Array();
listbodyUrls.push("https://XXX.jpg"); //此处替换为公网可以访问的图片地址
body.withThreshold(0);
body.withCategories(listbodyCategories);
body.withUrls(listbodyUrls);
request.withBody(body);
const result = client.runImageBatchModeration(request);
result.then(result => {
    console.log("JSON.stringify(result)::" + JSON.stringify(result));
}).catch(ex => {
    console.log("exception:" + JSON.stringify(ex));
});
```

2. 审核结果输出到控制台表示执行成功，[图像内容审核批量任务](#)提交的结果输出到控制台。

```
JSON.stringify(result)::{"statusCode":200,"result":{"categorySuggestions":
{"terrorism":"pass","porn":"pass"},"detail":{"porn":{"confidence":0.9883,"label":"normal"},
{"confidence":0.0039,"label":"porn"},"confidence":0.0078,"label":"sexy"},"terrorism":
[{"confidence":2.0E-4,"label":"terrorist"},{"confidence":2.0E-4,"label":"fascist"},
{"confidence":1.0E-4,"label":"cult"},{"confidence":2.0E-4},{"confidence":1.0E-4,"label":"kidnap"},
{"confidence":6.0E-4,"label":"riot"},{"confidence":1.0E-4,"label":"parade"},
{"confidence":2.0E-4,"label":"corpse"},{"confidence":1.0E-4,"label":"bloody"},
{"confidence":1.0E-4,"label":"knife"},{"confidence":1.0E-4,"label":"gun"},
{"confidence":2.0E-4,"label":"military_weapon"},{"confidence":0.0,"label":"special_characters"},
{"confidence":0.9994,"label":"normal"}],"suggestion":"pass","url":"https://sdk-obs-source-save.obs.cn-
north-4.myhuaweicloud.com/terrorism.jpg"}}
```

11.3.2.4 内容审核-图像（异步批量）（V2）

11.3.2.4.1 提交任务

1. 本章节对图像内容审核（异步批量）-提交任务AK/SK方式使用SDK进行示例说明。

示例代码中可以修改TaskSubmitReq的Urls配置待审核的图像url（仅支持url方式），配置完成后运行即可。

```
const core = require('@huaweicloud/huaweicloud-sdk-core');
const moderation = require("@huaweicloud/huaweicloud-sdk-moderation");
// 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中密
文存放，使用时解密，确保安全；
// 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK、HUAWEICLOUD_SDK_PROJECT_ID。
const ak = process.env.HUAWEICLOUD_SDK_AK;
const sk = process.env.HUAWEICLOUD_SDK_SK;
const projectId = process.env.HUAWEICLOUD_SDK_PROJECT_ID;

const endpoint = "https://moderation.cn-north-4.myhuaweicloud.com"; //把cn-north-4替换成服务所在
的区域。
const credentials = new core.BasicCredentials()
    .withAk(ak)
    .withSk(sk)
```

```
const client = moderation.ModerationClient.newBuilder()
    .withCredential(credentials)
    .withEndpoint(endpoint)
    .build();
const request = new moderation.RunTaskSubmitRequest();
const body = new moderation.TaskSubmitReq();
const listbodyCategories = new Array();
listbodyCategories.push("porn");
listbodyCategories.push("terrorism");
const listbodyUrls = new Array();
listbodyUrls.push("https://XXX.jpg"); //此处替换为公网可以访问的图片地址
body.withCategories(listbodyCategories);
body.withUrls(listbodyUrls);
request.withBody(body);
const result = client.runTaskSubmit(request);
result.then(result => {
    console.log("JSON.stringify(result)::" + JSON.stringify(result));
}).catch(ex => {
    console.log("exception:" + JSON.stringify(ex));
});
```

2. 提交任务结果输出到控制台表示执行成功，异步批量[任务提交](#)的结果输出到控制台。

```
JSON.stringify(result)::{"statusCode":200,"result":
{"jobId":"7f6a5881-7774-40cf-903b-4548914b55a3"}}
```

11.3.2.4.2 查询任务

1. 本章节对图像内容审核异步批量-查询任务AK/SK方式使用SDK进行示例说明。
示例代码中需要配置使用RunCheckResultRequest的JobId配置审核任务的id，配置完成后运行即可。

```
const core = require('@huaweicloud/huaweicloud-sdk-core');
const moderation = require("@huaweicloud/huaweicloud-sdk-moderation");
// 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中密文存放，使用时解密，确保安全；
// 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK、HUAWEICLOUD_SDK_PROJECT_ID。
const ak = process.env.HUAWEICLOUD_SDK_AK;
const sk = process.env.HUAWEICLOUD_SDK_SK;
const projectId = process.env.HUAWEICLOUD_SDK_PROJECT_ID;

const endpoint = "https://moderation.cn-north-4.myhuaweicloud.com"; //把cn-north-4替换成服务所在的区域。
const credentials = new core.BasicCredentials()
    .withAk(ak)
    .withSk(sk);
const client = moderation.ModerationClient.newBuilder()
    .withCredential(credentials)
    .withEndpoint(endpoint)
    .build();
const request = new moderation.RunCheckResultRequest();
request.jobId = "321cb5b6-3ad3-461a-b258-7ba7f530b91a";
const result = client.runCheckResult(request);
result.then(result => {
    console.log("JSON.stringify(result)::" + JSON.stringify(result));
}).catch(ex => {
    console.log("exception:" + JSON.stringify(ex));
});
```

2. 审核结果输出到控制台表示执行成功，异步批量[查询任务](#)的结果输出到控制台。
JSON.stringify(result)::{"statusCode":200,"result":{"createTime":"2021-08-13T03:46:30Z","items":[{"categorySuggestions":{"terrorism":"pass","porn":"pass"},"detail":{"porn":[{"confidence":0.9883,"label":"normal"}, {"confidence":0.0039,"label":"porn"}, {"confidence":0.0078,"label":"sexy"}],"terrorism":[{"confidence":2.0E-4,"label":"terrorist"}, {"confidence":2.0E-4,"label":"fascist"}, {"confidence":1.0E-4,"label":"cult"}, {"confidence":2.0E-4,"label":"confidence":1.0E-4}, {"confidence":1.0E-4,"label":"kidnap"}, {"confidence":6.0E-4,"label":"riot"}, {"confidence":1.0E-4,"label":"parade"}, {"confidence":2.0E-4,"label":"corpse"}]}

```
{"confidence":1.0E-4,"label":"bloody"},{"confidence":1.0E-4,"label":"knife"},
{"confidence":1.0E-4,"label":"gun"},{"confidence":2.0E-4,"label":"military_weapon"},
{"confidence":0.0,"label":"special_characters"},
{"confidence":0.9994,"label":"normal"}], "suggestion":"pass", "url":"https://sdk-obs-source-save.obs.cn-north-4.myhuaweicloud.com/terrorism.jpg"}, {"jobId":"7f6a5881-7774-40cf-903b-4548914b55a3", "status":"finish", "updateTime":"2021-08-13T03:46:30Z"}}
```

11.3.3 音频审核

11.3.3.1 创建音频内容审核作业

1. 本章节对创建音频审核AK/SK方式使用SDK进行示例说明。

创建音频审核示例代码只需将AK/SK信息替换为实际AK/SK，代码中可以使用RunCreateAudioModerationJobRequest类的withUrls配置待审核的音频url（仅支持url方式），配置完成后运行即可。

```
const core = require('@huaweicloud/huaweicloud-sdk-core');
const moderation = require("@huaweicloud/huaweicloud-sdk-moderation");
// 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中密文存放，使用时解密，确保安全；
// 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK、HUAWEICLOUD_SDK_PROJECT_ID。
const ak = process.env.HUAWEICLOUD_SDK_AK;
const sk = process.env.HUAWEICLOUD_SDK_SK;
const projectId = process.env.HUAWEICLOUD_SDK_PROJECT_ID;

const endpoint = "https://moderation.cn-north-4.myhuaweicloud.com";
const credentials = new core.BasicCredentials()
    .withAk(ak)
    .withSk(sk)
    .withProjectId(project_id)

const client = moderation.v3.ModerationClient.newBuilder()
    .withCredential(credentials)
    .withEndpoint(endpoint)
    .build();

const request = new moderation.v3.RunCreateAudioModerationJobRequest();
const body = new moderation.v3.AudioCreateRequest();
const listbodyCategories = new Array();
listbodyCategories.push("porn");
const databody = new moderation.v3.AudioInputBody();
databody.withUrl("https://*.wav");
body.withCategories(listbodyCategories);
body.withEventType("default");
body.withData(databody);
request.withBody(body);
const result = client.runCreateAudioModerationJob(request);
result.then(result => {
    console.log("JSON.stringify(result)::" + JSON.stringify(result));
}).catch(ex => {
    console.log("exception:" + JSON.stringify(ex));
});
```

2. 控制台输出创建音频内容审核提交的结果，[创建音频内容审核结果输出到控制台](#)。

```
JSON.stringify(result)::{"request_id": "0b111ce94ec3d563a40d4e51a096d996", "job_id": "xxx"}
```

11.3.3.2 查询音频内容审核作业

1. 本章节对查询音频审核AK/SK方式使用SDK进行示例说明。

查询音频审核示例代码只需将AK/SK信息替换为实际AK/SK，代码中需要配置使用RunQueryAudioModerationJobRequest类的withJobId配置审核任务的id，配置完成后运行即可。

```
const core = require('@huaweicloud/huaweicloud-sdk-core');
const moderation = require("@huaweicloud/huaweicloud-sdk-moderation");
// 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中密
文存放，使用时解密，确保安全；
// 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK、HUAWEICLOUD_SDK_PROJECT_ID。
const ak = process.env.HUAWEICLOUD_SDK_AK;
const sk = process.env.HUAWEICLOUD_SDK_SK;
const projectId = process.env.HUAWEICLOUD_SDK_PROJECT_ID;

const endpoint = "https://moderation.cn-north-4.myhuaweicloud.com";
const credentials = new core.BasicCredentials()
    .withAk(ak)
    .withSk(sk)
    .withProjectId(project_id)
const client = moderation.v3.ModerationClient.newBuilder()
    .withCredential(credentials)
    .withEndpoint(endpoint)
    .build();
const request = new moderation.v3.RunQueryAudioModerationJobRequest();
request.jobId = "xxxxxx"; //此处需要输入您的jobID
const result = client.runQueryAudioModerationJob(request);
result.then(result => {
    console.log("JSON.stringify(result)::" + JSON.stringify(result));
}).catch(ex => {
    console.log("exception:" + JSON.stringify(ex));
});
```

- 控制台输出查询音频内容审核的结果，[查询音频内容审核](#)结果输出到控制台。

```
JSON.stringify(result)::{"job_id":"xxx","status":"succeeded","result":{"suggestion":"block","details":
[{"start_time":0,"suggestion":"block","end_time":10,"label":"porn","audio_text":"xxxx","segments":
[{"segment":"xxx"},{"segment":"xxx"},{"segment":"xxx"}]},
{"start_time":30,"suggestion":"block","end_time":40,"label":"porn","audio_text":"xxx","segments":
[{"segment":"xx"}]}],"audio_text":"xxxxxx"},"request_params":{"event_type":"default","data":
{"url":"https://xxx.wav"},"callback":"http://xxx","categories":["porn","ad"],"create_time":"2022-06-20
11:05:29","update_time":"2022-06-20 11:07:08","request_id":"2419446b1fe14203f64e4018d12db3dd"}
```

11.3.4 视频审核

11.3.4.1 创建视频内容审核作业

- 本章节对创建视频审核AK/SK方式使用SDK进行示例说明。

示例代码中可以使用RunCreateVideoModerationJobRequest类的withUrls配置待审核的视频url（仅支持url方式），配置完成后运行即可。

```
const core = require('@huaweicloud/huaweicloud-sdk-core');
const moderation = require("@huaweicloud/huaweicloud-sdk-moderation");
// 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中密
文存放，使用时解密，确保安全；
// 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK、HUAWEICLOUD_SDK_PROJECT_ID。
const ak = process.env.HUAWEICLOUD_SDK_AK;
const sk = process.env.HUAWEICLOUD_SDK_SK;
const projectId = process.env.HUAWEICLOUD_SDK_PROJECT_ID;

const endpoint = "https://moderation.cn-north-4.myhuaweicloud.com"; //把cn-north-4替换成服务所在
的区域。
const credentials = new core.BasicCredentials()
    .withAk(ak)
    .withSk(sk)
    .withProjectId(project_id)
const client = moderation.v3.ModerationClient.newBuilder()
    .withCredential(credentials)
    .withEndpoint(endpoint)
    .build();
const request = new moderation.v3.RunCreateVideoModerationJobRequest();
const body = new moderation.v3.VideoCreateRequest();
```



```
const listbodyAudioCategories = new Array();
listbodyAudioCategories.push("porn");
const listbodyImageCategories = new Array();
listbodyImageCategories.push("porn");
const databody = new moderation.v3.VideoCreateRequestData();
databody.withUrl("http://xxx.xx.xxxx")
.withFrameInterval(5);
body.withAudioCategories(listbodyAudioCategories);
body.withImageCategories(listbodyImageCategories);
body.withEventType("default");
body.withData(databody);
request.withBody(body);
const result = client.runCreateVideoModerationJob(request);
result.then(result => {
  console.log("JSON.stringify(result)::" + JSON.stringify(result));
}).catch(ex => {
  console.log("exception:" + JSON.stringify(ex));
});
```

2. 控制台输出创建视频内容审核作业的结果，[创建视频内容审核](#)结果输出到控制台。

```
JSON.stringify(result)::{"request_id": "dbbcb75d1a174bfa899a38be38fdd107", "job_id": "xxx"}
```

11.3.4.2 查询视频内容审核作业

1. 本章节对查询视频审核AK/SK方式使用SDK进行示例说明。

示例代码中需要配置使用RunQueryVideoModerationJobRequest类的withJobId配置审核任务的id，配置完成后运行即可。

```
const core = require('@huaweicloud/huaweicloud-sdk-core');
const moderation = require("@huaweicloud/huaweicloud-sdk-moderation");
// 认证的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中密文存放，使用时解密，确保安全；
// 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK、HUAWEICLOUD_SDK_PROJECT_ID。
const ak = process.env.HUAWEICLOUD_SDK_AK;
const sk = process.env.HUAWEICLOUD_SDK_SK;
const projectId = process.env.HUAWEICLOUD_SDK_PROJECT_ID;

const endpoint = "https://moderation.cn-north-4.myhuaweicloud.com"; //把cn-north-4替换成服务所在的区域。
const credentials = new core.BasicCredentials()
.withAk(ak)
.withSk(sk)
const client = moderation.v3.ModerationClient.newBuilder()
.withCredential(credentials)
.withEndpoint(endpoint)
.build();

const request = new moderation.v3.RunQueryVideoModerationJobRequest();
request.jobId = "xxxx"; //此处需要输入您的jobID
const result = client.runQueryVideoModerationJob(request);
result.then(result => {
  console.log("JSON.stringify(result)::" + JSON.stringify(result));
}).catch(ex => {
  console.log("exception:" + JSON.stringify(ex));
});
```

2. 控制台输出查询视频内容审核作业提交的结果，[查询视频内容审核](#)结果输出到控制台。

```
JSON.stringify(result)::{"request_id":"xxxx","job_id":"xxx","status":"succeeded","request_params":
{"data":{"url":"xxxx","frame_interval":5},"event_type":"default","image_categories":
["porn","terrorism","image_text"],"audio_categories":
["porn","ad","moan","abuse"],"callback":"xxx"},"create_time":"2022-07-30T08:57:11.011Z","update_time":
"2022-07-30T08:57:14.014Z","result":{"suggestion":"block","image_detail":
[{"suggestion":"block","category":"porn","ocr_text":"123","time":0,"detail":
[{"confidence":0.676015138626099,"category":"porn","suggestion":"block","label":"pornography"},
{"confidence":0.666015138626099,"category":"porn","suggestion":"review","label":"sexy_female"},
{"confidence":0.666015138626099,"category":"xxx","suggestion":"block","label":"leader"}]}
```

```
    {"confidence":0.691423773765564,"category":"xxx","suggestion":"block","face_location":
    {"top_left_x":12,"top_left_y":13,"bottom_right_x":200,"bottom_right_y":240},"label":"leader"},
    {"confidence":0.691423773765564,"category":"porn","suggestion":"block","face_location":
    {"top_left_x":12,"top_left_y":13,"bottom_right_x":100,"bottom_right_y":2000},"label":"leader"},
    {"confidence":0.891423773765564,"category":"image_text","suggestion":"block","label":"qr_code"},
    {"confidence":0.891423773765564,"category":"image_text","suggestion":"block","label":"qr_code","qr_c
    ontent":"xxxxxxx","qr_location":
    {"top_left_x":12,"top_left_y":13,"bottom_right_x":100,"bottom_right_y":2000}},
    {"confidence":0.891423773765564,"category":"image_text","suggestion":"block","label":"porn","segme
    nts":[{"segment":"xxx"},{"segment":"xx"}]}], "audio_detail":
    [{"suggestion":"block","label":"porn","audio_text":"xxxxx","end_time":10,"start_time":0,"detail":
    [{"confidence":0.999117187582349,"label":"porn","suggestion":"block","segments":
    [{"segment":"xxx"}]},{"confidence":0.885117187582349,"label":"porn","suggestion":"block"}]}]}
```

11.3.5 音频流审核

11.3.5.1 创建音频流内容审核作业

1. 本章节对创建音频流审核AK/SK方式使用SDK进行示例说明。

示例代码中可以使用AudioStreamCreateRequestData类的withUrls配置待审核的音频流url（仅支持url方式），配置完成后运行即可。

```
const core = require('@huaweicloud/huaweicloud-sdk-core');
const moderation = require("@huaweicloud/huaweicloud-sdk-moderation/v3/public-api");
// 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中密
文存放，使用时解密，确保安全；
// 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量
HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK、HUAWEICLOUD_SDK_PROJECT_ID。
const ak = process.env.HUAWEICLOUD_SDK_AK;
const sk = process.env.HUAWEICLOUD_SDK_SK;
const projectId = process.env.HUAWEICLOUD_SDK_PROJECT_ID;

const endpoint = "https://moderation.cn-north-4.myhuaweicloud.com"; //把cn-north-4替换成服务所在
的区域。
const credentials = new core.BasicCredentials()
    .withAk(ak)
    .withSk(sk)
    .withProjectId(project_id)
const client = moderation.ModerationClient.newBuilder()
    .withCredential(credentials)
    .withEndpoint(endpoint)
    .build();

const request = new moderation.RunCreateAudioStreamModerationJobRequest();
const body = new moderation.v3.AudioStreamCreateRequest();
const listbodyCategories = new Array();
listbodyCategories.push("ad");
const databody = new moderation.v3.AudioStreamCreateRequestData();
databody.withUrl("http://xxx");
body.withCallback("http://xxxxxx");
body.withCategories(listbodyCategories);
body.withEventType("default");
body.withData(databody);
request.withBody(body);
const result = client.runCreateAudioStreamModerationJob(request);
result.then(result => {
    console.log("JSON.stringify(result)::" + JSON.stringify(result));
}).catch(ex => {
    console.log("exception:" + JSON.stringify(ex));
});
```

2. 控制台输出创建音频流内容审核作业的结果。

```
JSON.stringify(result):: {"request_id": "303fbd70c9813a2476bad125fa436f00", "job_id":
"11690387200_ffa1f5bb9a58451b86816gdfc654610e"}
```

11.3.5.2 关闭音频流内容审核作业

1. 本章节对关闭音频流审核AK/SK方式使用SDK进行示例说明。

示例代码中需要配置使用RunCloseAudioStreamModerationJobRequest类的withJobId配置审核任务的id，配置完成后运行即可。

```
const core = require('@huaweicloud/huaweicloud-sdk-core');
const moderation = require("@huaweicloud/huaweicloud-sdk-moderation/v3/public-api");
// 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中密文存放，使用时解密，确保安全；
// 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK、HUAWEICLOUD_SDK_PROJECT_ID。
const ak = process.env.HUAWEICLOUD_SDK_AK;
const sk = process.env.HUAWEICLOUD_SDK_SK;
const projectId = process.env.HUAWEICLOUD_SDK_PROJECT_ID;

const endpoint = "https://moderation.cn-north-4.myhuaweicloud.com"; //把cn-north-4替换成服务所在的区域。
const credentials = new core.BasicCredentials()
    .withAk(ak)
    .withSk(sk)
    .withProjectId(project_id)
const client = moderation.ModerationClient.newBuilder()
    .withCredential(credentials)
    .withEndpoint(endpoint)
    .build();
const request = new moderation.RunCloseAudioStreamModerationJobRequest();
request.jobId = "xxxxx";
const result = client.runCloseAudioStreamModerationJob(request);
result.then(result => {
    console.log("JSON.stringify(result)::" + JSON.stringify(result));
}).catch(ex => {
    console.log("exception:" + JSON.stringify(ex));
});
```

2. 控制台输出关闭音频流内容审核作业提交的结果。
JSON.stringify(result)::{"result": "success"}

11.3.6 视频流审核

11.3.6.1 创建视频流内容审核作业

1. 本章节对创建视频流审核AK/SK方式使用SDK进行示例说明。

示例代码中可以使用VideoStreamCreateRequestData类的withUrls配置待审核的视频流url（仅支持url方式），配置完成后运行即可。

```
const core = require('@huaweicloud/huaweicloud-sdk-core');
const moderation = require("@huaweicloud/huaweicloud-sdk-moderation/v3/public-api");

// 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中密文存放，使用时解密，确保安全；
// 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK、HUAWEICLOUD_SDK_PROJECT_ID。
const ak = process.env.HUAWEICLOUD_SDK_AK;
const sk = process.env.HUAWEICLOUD_SDK_SK;
const projectId = process.env.HUAWEICLOUD_SDK_PROJECT_ID;

const endpoint = "https://moderation.cn-north-4.myhuaweicloud.com"; //把cn-north-4替换成服务所在的区域。
const credentials = new core.BasicCredentials()
    .withAk(ak)
    .withSk(sk)
    .withProjectId(project_id)
const client = moderation.ModerationClient.newBuilder()
    .withCredential(credentials)
    .withEndpoint(endpoint)
```

```
        .build();
const request = new moderation.RunCreateVideoStreamModerationJobRequest();
const body = new moderation.VideoStreamCreateRequest();
const listbodyImageCategories = new Array();
listbodyImageCategories.push("politics");
const databody = new moderation.VideoStreamCreateRequestData();
databody.withUrl("http://xxx");
body.withCallback("http://xxxxxx");
body.withImageCategories(listbodyImageCategories);
body.withEventType("default");
body.withData(databody);
request.withBody(body);
const result = client.runCreateVideoStreamModerationJob(request);
result.then(result => {
    console.log("JSON.stringify(result)::" + JSON.stringify(result));
}).catch(ex => {
    console.log("exception:" + JSON.stringify(ex));
});
```

2. 控制台输出创建视频流内容审核作业的结果。

```
JSON.stringify(result)::{"request_id": "fa520ddd6b54af51c8d1e6d19898656", "job_id": "01690387200_09ad14f155bb47e9bf2718ace57c73ed"}
```

11.3.6.2 关闭视频流内容审核作业

1. 本章节对关闭视频流审核AK/SK方式使用SDK进行示例说明。

示例代码中需要配置使用RunCloseVideoStreamModerationJobRequest类的withJobId配置审核任务的id，配置完成后运行即可。

```
const core = require('@huaweicloud/huaweicloud-sdk-core');
const moderation = require("@huaweicloud/huaweicloud-sdk-moderation/v3/public-api");

// 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中密文存放，使用时解密，确保安全；
// 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK、HUAWEICLOUD_SDK_PROJECT_ID。
const ak = process.env.HUAWEICLOUD_SDK_AK;
const sk = process.env.HUAWEICLOUD_SDK_SK;
const projectId = process.env.HUAWEICLOUD_SDK_PROJECT_ID;

const endpoint = "https://moderation.cn-north-4.myhuaweicloud.com";//把cn-north-4替换成服务所在的区域。
const credentials = new core.BasicCredentials()
    .withAk(ak)
    .withSk(sk)
    .withProjectId(project_id)
const client = moderation.ModerationClient.newBuilder()
    .withCredential(credentials)
    .withEndpoint(endpoint)
    .build();
const request = new moderation.RunCloseVideoStreamModerationJobRequest();
request.jobId = "xxxxx";
const result = client.runCloseVideoStreamModerationJob(request);
result.then(result => {
    console.log("JSON.stringify(result)::" + JSON.stringify(result));
}).catch(ex => {
    console.log("exception:" + JSON.stringify(ex));
});
```

2. 控制台输出关闭视频流内容审核作业提交的结果。

```
JSON.stringify(result)::{"result": "success"}
```

11.3.7 文档审核

11.3.7.1 创建文档内容审核作业

1. 本章节对创建文档内容审核作业AK/SK方式使用SDK进行示例说明。

示例代码中可以使用DocumentCreateRequestData类的withUrl配置待审核的文档url，配置完成后运行即可。

```
const core = require('@huaweicloud/huaweicloud-sdk-core');
const moderation = require("@huaweicloud/huaweicloud-sdk-moderation/v3/public-api");
// 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中密文存放，使用时解密，确保安全；
// 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK、HUAWEICLOUD_SDK_PROJECT_ID。
const ak = process.env.HUAWEICLOUD_SDK_AK;
const sk = process.env.HUAWEICLOUD_SDK_SK;
const projectId = process.env.HUAWEICLOUD_SDK_PROJECT_ID;

const endpoint = "https://moderation.cn-north-4.myhuaweicloud.com"; //把cn-north-4替换成服务所在的区域。
const credentials = new core.BasicCredentials()
    .withAk(ak)
    .withSk(sk)
    .withProjectId(project_id)
const client = moderation.ModerationClient.newBuilder()
    .withCredential(credentials)
    .withEndpoint(endpoint)
    .build();
const request = new moderation.RunCreateDocumentModerationJobRequest();
const body = new moderation.DocumentCreateRequest();
const databody = new moderation.DocumentCreateRequestData();
databody.withUrl("http://xxx")
    .withFormat("pdf");
body.withEventType("default");
body.withData(databody);
request.withBody(body);
const result = client.runCreateDocumentModerationJob(request);
result.then(result => {
    console.log("JSON.stringify(result)::" + JSON.stringify(result));
}).catch(ex => {
    console.log("exception:" + JSON.stringify(ex));
});
```

2. 控制台输出创建文本内容审核作业的结果。

```
JSON.stringify(result)::{"request_id": "c083957284ce1f2c0b910115d8232ec6", "job_id": "01690387200_e3dfb78d9f20463f98aee25e6895203a"}
```

11.3.7.2 查询文档审核作业结果

1. 本章节对查询文档审核作业结果AK/SK方式使用SDK进行示例说明。

示例代码中需要配置使用RunQueryDocumentModerationJobRequest类的withJobId配置审核任务的id，配置完成后运行即可。

```
const core = require('@huaweicloud/huaweicloud-sdk-core');
const moderation = require("@huaweicloud/huaweicloud-sdk-moderation/v3/public-api");

// 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险，建议在配置文件或者环境变量中密文存放，使用时解密，确保安全；
// 本示例以ak和sk保存在环境变量中来实现身份验证为例，运行本示例前请先在本地环境中设置环境变量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK、HUAWEICLOUD_SDK_PROJECT_ID。
const ak = process.env.HUAWEICLOUD_SDK_AK;
const sk = process.env.HUAWEICLOUD_SDK_SK;
const projectId = process.env.HUAWEICLOUD_SDK_PROJECT_ID;

const endpoint = "https://moderation.cn-north-4.myhuaweicloud.com"; //把cn-north-4替换成服务所在的区域。
const credentials = new core.BasicCredentials()
    .withAk(ak)
    .withSk(sk)
    .withProjectId(project_id)
```

```
const client = moderation.ModerationClient.newBuilder()
    .withCredential(credentials)
    .withEndpoint(endpoint)
    .build();
const request = new moderation.RunQueryDocumentModerationJobRequest();
request.jobId = "xxxx";
const result = client.runQueryDocumentModerationJob(request);
result.then(result => {
    console.log("JSON.stringify(result)::" + JSON.stringify(result));
}).catch(ex => {
    console.log("exception:" + JSON.stringify(ex));
});
```

2. 控制台输出查询文档审核作业结果。

```
JSON.stringify(result)::{"job_id": "01693238400_cd7c8fd0a30b4c13b1fc9246c542c6", "status":
"succeeded", "result": {"suggestion": "pass", "details": []}, "request_params": {"data": {"url": "http://
xxxx", "format": "pdf"}, "event_type": "default"}, "create_time": "2023-08-29T16:33:50.050Z",
"update_time": "2023-08-29T16:33:50.050Z", "request_id": "e4eeb972aeb92f4269e8dfbb59ba7e3b"}
```