

代码检查

# 用户指南

文档版本 01  
发布日期 2024-10-09



版权所有 © 华为云计算技术有限公司 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云计算技术有限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，华为云计算技术有限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。

---

# 目录

---

<b>1 代码检查服务（CodeArts Check）使用流程</b>	<b>1</b>
<b>2 购买 CodeArts Check</b>	<b>3</b>
2.1 购买 CodeArts Check	3
<b>3 配置 CodeArts Check 项目级角色权限</b>	<b>4</b>
<b>4 创建代码检查任务</b>	<b>6</b>
<b>5 配置代码检查任务</b>	<b>10</b>
5.1 配置代码检查任务规则集	10
5.1.1 配置代码检查任务预置规则集	10
5.1.2 配置代码检查任务自定义规则集	13
5.2 配置代码检查任务质量门禁	14
5.3 配置代码检查任务定时执行	15
5.4 配置代码检查任务消息通知	16
5.5 配置代码检查任务检查模式	16
5.6 查看徽标状态	18
5.7 配置代码检查任务集成服务	19
5.8 配置代码检查任务自定义环境	20
5.9 配置代码检查任务高级选项	20
<b>6 执行代码检查任务</b>	<b>22</b>
<b>7 查看代码检查结果</b>	<b>23</b>
<b>8 查询审计日志</b>	<b>28</b>

# 1 代码检查服务（CodeArts Check）使用流程

代码检查（CodeArts Check）是基于云端实现的代码检查服务。建立在多年自动化源代码静态检查技术积累与企业级应用经验的沉淀之上，为用户提供代码风格、通用质量与网络安全风险等丰富的检查能力，提供全面的质量报告、便捷的问题闭环处理流程，帮助企业有效管控代码质量，助力企业成功。

## 使用流程

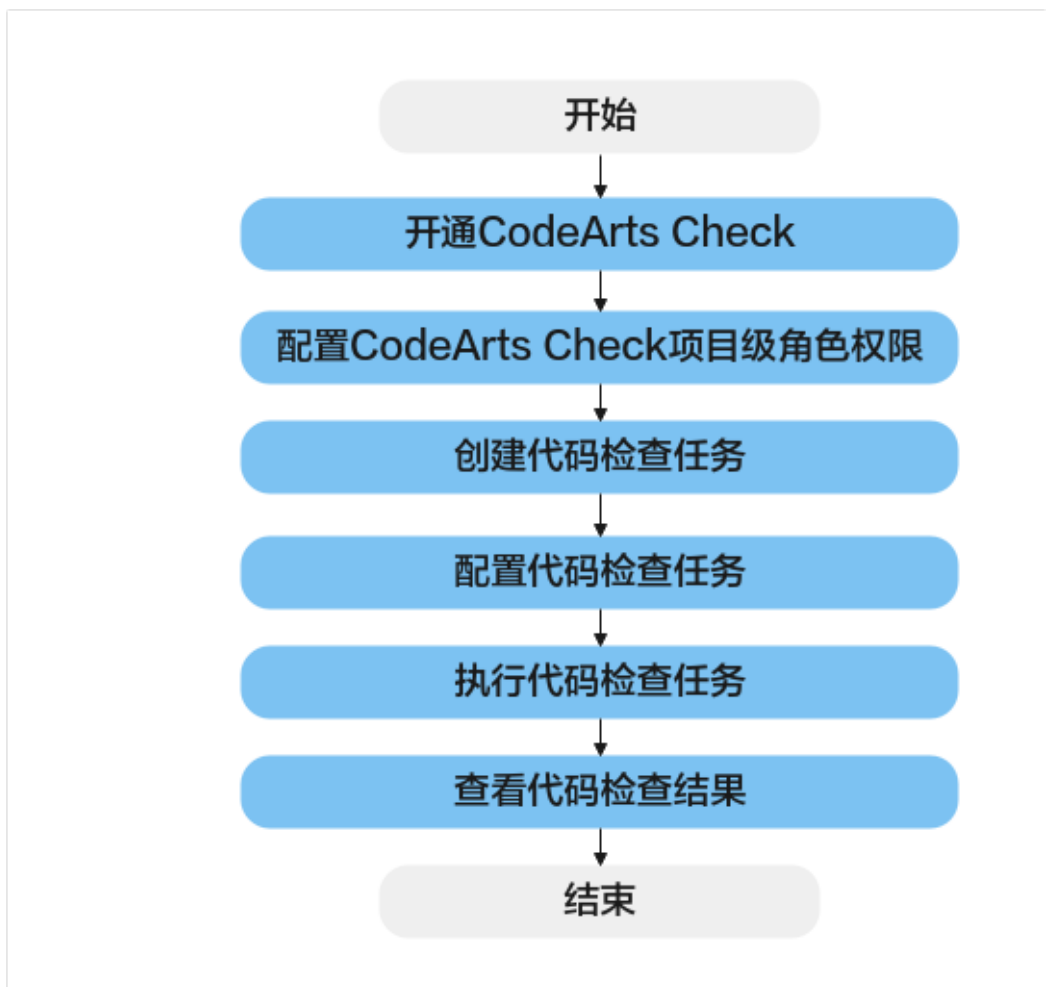


表 1-1 使用流程说明

流程	说明
<a href="#">购买 CodeArts Check</a>	指导您如何开通代码检查服务。
<a href="#">配置 CodeArts Check 项目级角色权限</a>	为您介绍如何配置 CodeArts Check 服务项目级权限以及访问代码检查服务首页的方式。
<a href="#">创建代码检查任务</a>	您可创建不同代码源的代码检查任务。
<a href="#">配置代码检查任务</a>	您可对已有的检查任务进行修改或配置。
<a href="#">执行代码检查任务</a>	您可对目标代码检查任务执行检查，及时发现源代码中的代码问题。
<a href="#">查看代码检查结果</a>	代码检查任务执行完成后，您可查看检查结果并修改代码问题。

# 2 购买 CodeArts Check

---

## 2.1 购买 CodeArts Check

### 前提条件

已注册华为云并实名认证，如果还没有华为账号，请参考以下步骤创建。

1. 打开[华为云网站](#)。
2. 单击“注册”，根据提示信息完成注册。  
注册成功后，系统会自动跳转至您的个人信息界面。
3. 参考[实名认证](#)完成个人或企业账号实名认证。

### 购买 CodeArts Check

可参考[购买CodeArts套餐](#)。

# 3 配置 CodeArts Check 项目级角色权限

新增的成员需赋予指定的角色，不同角色具备的默认权限不同。各角色具备的默认权限如表3-1。

表 3-1 代码检查服务默认角色权限


资源权限/角色		项目 管理员	项目 经理	开发 人员	测试 经理	测试 人员	参与 者	浏览 者	产 品 经 理	系 统 工 程 师	Co mm itter
检查 任务	创建	√	√	√	×	×	×	×	×	√	√
	执行	√	√	√	×	×	×	×	√	√	√
	查看	√	√	√	×	×	×	×	√	√	√
	编辑	√	√	√	×	×	×	×	√	√	√
	删除	√	√	×	×	×	×	×	√	√	√
规则 集	设置 默认 规则 集	√	√	×	×	×	×	×	√	√	√
缺陷 报告	查看	√	√	√	×	×	×	×	√	√	√
	编辑	√	√	√	×	×	×	×	√	√	√

## 前提条件

- 已[购买CodeArts Check](#)。
- 已参考软件开发生产线(CodeArts)的“用户指南 > 软件开发生产线 (CodeArts) 使用前准备 > 添加CodeArts项目成员”章节添加成员，并为新增的成员[赋予角色](#)。

## 访问 CodeArts Check 服务首页

**步骤1** 使用华为云账号[登录华为云控制台页面](#)。

**步骤2** 单击页面左上角 ，在服务列表中选择“开发与运维 > 代码检查 CodeArts Check”。

**步骤3** 代码检查服务页面有两种访问方式：首页入口和项目入口。

- **首页入口**

单击“立即使用”，进入代码检查服务首页。该页面展示的是与当前用户相关的代码检查任务列表。



- **项目入口**

- 单击“立即使用”，进入代码检查服务首页。
- 单击导航栏“首页”。
- 单击需要查看的项目名称。
- 选择“代码 > 代码检查”，进入指定项目下代码检查任务列表页。

单击页面左上角 ，可根据需要选择区域。

----结束

## 配置项目级角色权限

- 通过项目入口方式[访问CodeArts Check服务首页](#)。
- 选择导航栏“设置 > 权限管理”。
- 在配置权限页面，为不同的角色配置代码检查服务的各个资源权限。





# 4 创建代码检查任务

代码检查服务可检查不同代码仓的代码质量，在创建代码检查任务前，需满足以下条件：

- 已新建CodeArts项目。
- 如果使用的是代码托管服务（CodeArts Repo）的代码仓库，则可以参考代码托管服务的“用户指南 > 新建Repo代码仓库”，完成[新建代码仓库](#)。
- 如果使用的是第三方代码仓库，需已创建可用的代码仓库。

## 创建 Repo 代码源检查任务

在代码托管中创建云端仓库时，若勾选了“自动创建代码检查任务”，则代码仓库创建完成后，在代码检查任务列表可查看对应仓库的检查任务。

以下示例为在代码托管服务中创建云端仓库不勾选“自动创建代码检查任务”时，Repo代码源代码检查任务的创建步骤。

**步骤1** 基于项目入口[访问CodeArts Check服务首页](#)。

**步骤2** 在代码检查页面，单击“新建任务”，在“新建任务”页面，参考[表4-1](#)配置参数后单击“新建任务”。

表 4-1 Repo 代码源检查任务参数说明

参数名称	参数说明	是否必填
归属项目	任务所属项目。 <ul style="list-style-type: none"><li>• 以项目入口方式访问CodeArts Check服务时默认填写，无需手动填写。</li><li>• 以服务入口访问时需根据实际情况选择<a href="#">新建项目</a>中创建的项目。</li></ul>	是
代码源	选择Repo，检查代码托管服务中的代码质量。	是
任务名称	代码检查任务名称，可自定义。 <ul style="list-style-type: none"><li>• 支持中英文，数字，点，下划线“_”和连接符“-”。</li><li>• 字符长度范围为1~128。</li></ul>	是

参数名称	参数说明	是否必填
仓库	选择需要检查的代码仓库。	是
分支	选择需要检查的仓库分支。	是
检查语言	选择需要检查的代码语言。	否

### 📖 说明

- 创建代码检查任务后，如果您需要修改任务名称或默认分支，可在“代码检查详情”页面，选择“设置 > 基本信息”，修改任务名称和默认分支。
- 对于不再继续使用的代码检查任务，可进入“代码检查详情”页面，选择“设置 > 基本信息”，删除当前代码检查任务。也可以单击代码检查任务所在行 **\*\*\***，在下拉框中，选择“删除”。

若删除的代码检查任务其代码仓为多分支代码仓，分支数量小于200时可删除成功，大于等于200会删除失败。

----结束

## 创建通用 Git 代码源检查任务

**步骤1** 基于项目入口[访问CodeArts Check服务首页](#)。

**步骤2** 在代码检查页面，单击“新建任务”，在“新建任务”页面，参考表4-2配置参数后单击“新建任务”。

表 4-2 通用 Git 代码源检查任务参数说明

参数项	描述	是否必填
归属项目	任务所属项目。 <ul style="list-style-type: none"> <li>• 以项目入口方式访问CodeArts Check服务时默认填写，无需手动填写。</li> <li>• 以服务入口访问时需根据实际情况选择<b>新建项目</b>中创建的项目。</li> </ul>	是
代码源	选择通用Git。对于托管在其他服务上的代码，可以使用通用Git连接实现代码拉取。	是
任务名称	代码检查任务名称，可自定义。 <ul style="list-style-type: none"> <li>• 支持中英文，数字，点，下划线“_”和连接符“-”。</li> <li>• 字符长度范围为1~128。</li> </ul>	是
Endpoint实例	选择连接到通用Git代码仓库的服务扩展点。如果没有，可参考 <b>新建服务扩展点</b> 中“新建通用Git服务扩展点”创建。	是

参数项	描述	是否必填
仓库	选择需要检查的代码仓库。	是
分支	输入需要检查的仓库分支。	是
检查语言	选择需要检查的代码语言。	否

### 📖 说明

- 创建代码检查任务后，如果您需要修改任务名称或默认分支，可在“代码检查详情”页面，选择“设置 > 基本信息”，修改任务名称和默认分支。
- 对于不再继续使用的代码检查任务，可进入“代码检查详情”页面，选择“设置 > 基本信息”，删除当前代码检查任务。也可以单击代码检查任务所在行<sup>\*\*\*</sup>，在下拉框中，选择“删除”。

若删除的代码检查任务其代码仓为多分支代码仓，分支数量小于200时可删除成功，大于等于200会删除失败。

### ----结束

## 新建服务扩展点

代码源选择第三仓库时需要配置。

服务扩展点（Endpoint）是软件开发生产线的一种扩展插件，为软件开发生产线提供链接第三方服务的能力。

CodeArts Check默认从代码托管服务拉取代码进行构建，同时也可使用服务扩展点连接第三方代码仓库获取项目源码。

### 📖 说明

- 使用第三方代码仓库可能出现网络不稳定或其他问题，具体使用体验取决于第三方代码仓库网络环境和服务状态。
- 建议使用代码托管的代码导入功能，将代码导入到代码托管，实现安全、稳定、高效下载与构建。

### 新建通用Git服务扩展点

- 单击“Endpoint实例”输入框中的▼，单击“新建扩展点”。
- 在弹出的页面中，单击“新建服务扩展点”，选择“通用Git”
- 在弹出的对话框中，参考表4-3配置参数。

表 4-3 通用 Git 服务扩展点参数说明

参数	说明	是否必填
连接名称	自定义连接名称。支持中文、英文、数字、“-”、“_”、“.”、空格，不超过256个字符。	是
Git仓库Url	填写用户实际使用的Git仓库的Url（https协议地址）。	是
用户名	填写用户实际使用的Git仓库用户名。	否
密码或Access Token	填写用户实际使用的Git仓库密码。	否

- 单击“确定”，返回创建代码检查任务页面。

# 5 配置代码检查任务

## 5.1 配置代码检查任务规则集

### 5.1.1 配置代码检查任务预置规则集

规则集则是一组用于做代码检查的规则组成的集合。一般根据不同的检查目标 and 需求，规则集也有相应的分类，比如对安全类、安卓应用、编程风格，使用这些类别进行检查的时候，规则集内的规则也会有所不同。

用户可根据实际需要更换代码检查任务使用的规则集，新启用的规则集将在下一次执行检查时生效。

代码检查服务支持多种语言的规则集，且每种语言类型对应多种不同级别的规则集，系统规则集列表请参见[表5-1](#)。

#### 须知

多个编译语言的规则集不能选择一起同时检查。例如：C#语言规则集不可与其它语言规则集进行混合检查。

表 5-1 预置规则集

语言	系统规则集类型	默认规则集	支持扫描文件后缀
C++	<ul style="list-style-type: none"><li>Generic Inspection Rule Set</li></ul>	Generic Inspection Rule Set	.c/.cc/.cpp/.cxx/.c p/.c+ +/.inc/.inl/.mm/ h/.hh/.hpp/.hxx/ h++
Java	<ul style="list-style-type: none"><li>Generic Inspection Rule Set</li></ul>	Generic Inspection Rule Set	.java

语言	系统规则集类型	默认规则集	支持扫描文件后缀
Go	<ul style="list-style-type: none"> <li>Generic Inspection Rule Set</li> </ul>	Generic Inspection Rule Set	.go
Python	<ul style="list-style-type: none"> <li>Generic Inspection Rule Set</li> </ul>	Generic Inspection Rule Set	.py
JavaScript	<ul style="list-style-type: none"> <li>Generic Inspection Rule Set</li> </ul>	Generic Inspection Rule Set	.js/.jsx
C#	<ul style="list-style-type: none"> <li>Generic Inspection Rule Set</li> </ul>	Generic Inspection Rule Set	.cs
TypeScript	<ul style="list-style-type: none"> <li>Generic Inspection Rule Set</li> </ul>	Generic Inspection Rule Set	.ts/.tsx
CSS	<ul style="list-style-type: none"> <li>Generic Inspection Rule Set</li> </ul>	Generic Inspection Rule Set	.css
HTML	<ul style="list-style-type: none"> <li>Generic Inspection Rule Set</li> </ul>	Generic Inspection Rule Set	.html
PHP	<ul style="list-style-type: none"> <li>Generic Inspection Rule Set</li> </ul>	Generic Inspection Rule Set	.php
LUA	通用检查规则集	通用检查规则集	.lua
RUST	通用检查规则集	通用检查规则集	.rs
Shell	通用检查规则集	通用检查规则集	.sh
KOTLIN	通用检查规则集	通用检查规则集	.kt

d  
e  
t  
e  
k  
t  
o  
u  
t

## 查看规则集中规则详情

用户可优先了解各个规则集中的规则详情，选择符合检查要求的规则集。可查看的规则详情包含：问题级别、正确示例、错误示例和修复建议等信息。

**步骤1** 基于项目入口[访问CodeArts Check服务首页](#)。

**步骤2** 单击“规则集”页签，进入检查规则集列表页面。

**步骤3** 单击规则集名称，进入到该规则集的规则详情页面可在线查看规则详情。



您也可以单击“导出”，导出该规则集中已启用的规则到本地查看规则详情。

----结束

## 修改默认规则集

初次使用代码检查服务时，每种编程语言都设置一个默认的检查规则集，如表5-1所示。如果用户需要修改各个语言的默认检查规则集，可参考以下指导操作。

### 说明

以项目入口访问代码检查服务，规则集列表中才会为每种检查语言预置默认规则集。

**步骤1** 在“规则集”页面的“语言”列筛选需要配置的语言。

**步骤2** 单击规则集所在行的\*\*\*，选择“默认规则集”，即可将当前的规则集设置为默认规则集。

----结束

## 配置代码检查任务使用的规则集

如果用户需要在代码检查任务中使用其他的预置规则集，可参考以下指导操作。如果预置规则集不满足用户的检查需求，可参考配置代码检查任务自定义规则集。

**步骤1** 进入代码检查页面，在任务列表中，单击任务名称。

**步骤2** 进入“代码检查详情”页面，选择“设置 > 规则集”。

**步骤3** 在“已包含语言”区域，打开目标语言开关。

**步骤4** 在“启用规则集”区域单击 ，选择需要使用的规则集。

当选中的规则集需配置扩展参数（此处以Java为例）时，单击“检查参数”，配置代码检查扩展参数并开启配置开关，完成后单击“确认”。

## 检查参数

代码检查扩展参数配置：请您在使用中保护好自己的敏感信息。

编译工具 

maven

编译工具版本

maven3.5.3-jdk1.8

编译命令 

mvn clean compile -Dmaven.test.skip=true -B -U -q

确认

取消

### 说明

用户购买了代码安全检查增强包后，使用代码安全检查增强包规则集必须设置编译脚本检查参数。

----结束

## 5.1.2 配置代码检查任务自定义规则集

代码检查服务除了可以使用预置规则集类型，用户还可以根据实际需要自定义代码检查规则集，每个规则集中最少设置一条规则。

目前仅支持配置单语言的规则集，即一种规则集只能配置同一种语言类型的检查规则。

### 说明

代码检查规则集提供了代码安全检查增强包功能，使用前请务必按需购买。购买了代码安全检查增强包后，Java、C++、Go和Python语言可以使用安全度更高的缺陷扫描规则。

## 创建自定义检查规则集

**步骤1** 基于项目入口[访问CodeArts Check服务首页](#)。

**步骤2** 单击“规则集”页签，进入规则集列表页面。

**步骤3** 单击“新建规则集”，在弹出的“新建规则集”窗口设置规则集名称、描述以及检查语言。

您也可以单击“复制于”下拉框选择复制于已有的规则集，基于复制的规则集配置使用的规则。

**步骤4** 单击“确定”。

**步骤5** 根据需要勾选规则名称、设置规则级别，单击右上角“保存”。



- 单击规则集名称，进入规则集详情页面可在线查看规则集中使用的规则详情。
- 单击规则集名称，进入规则集详情页面，单击“导出”按钮可导出该规则集中已启用的规则到本地查看规则详情。
- 单击规则集名称，进入规则集详情页面，单击“修改基本信息”，可修改规则集名称或描述信息。

#### 📖 说明

- “检查语言”不支持修改。
- 系统预置规则集无法修改。
- 只有创建者可以修改自定义规则集。
- 单击规则集所在行的  $\dots$ ，选择“删除”即可删除检查规则集。


#### 📖 说明

- 只有创建者可以删除自定义规则集。
- 系统规则集和已被使用的自定义规则集无法删除。
- 如果需要删除已被使用的规则集，需将关联使用该规则集的代码检查任务删除或将代码检查任务关联到其它规则集。

---结束

## 配置代码检查服务使用自定义规则集

**步骤1** 在代码检查详情页选择“设置 > 规则集”。

**步骤2** 如果创建代码检查任务后，代码仓的代码有修改，则需单击“已包含语言”所在行  重新获取代码仓语言，将目标语言的开关设置为开启状态。

**步骤3** 单击 ，选择已创建的自定义规则集。

---结束

## 5.2 配置代码检查任务质量门禁

根据质量要求设置门禁阈值，即可了解您的项目是否已准备好投入生产。例如，实际的检查结果超出设置的门禁阈值，则表示当前项目还不具备生产条件。

门禁质量设置支持租户级、项目级以及任务级，优先级为租户级 > 项目级 > 任务级。入口分别如下：

- 租户级：进入代码检查服务首页，选择“配置中心”，默认显示质量门禁详情。
- 项目级：进入项目详情页面，选择“代码 > 代码检查 > 配置中心 > 质量门禁”。
- 任务级：进入代码检查详情页面，选择“设置 > 质量门禁”。

#### 📖 说明

项目级和任务级支持事件[通知设置](#)。

本节以任务级门禁设置为例介绍操作步骤。

**步骤1** 基于项目入口[访问CodeArts Check服务首页](#)。

**步骤2** 进入代码检查页面，在任务列表中，单击任务名称。

**步骤3** 进入“代码检查详情”页面，选择“设置 > 质量门禁”。

**步骤4** 在“质量门禁”区域，根据质量要求开启门禁开关，并设置门禁阈值。

图 5-1 配置质量门禁

门禁项	运算逻辑	门禁阈值	门禁开关
致命问题数	≤	0	<input checked="" type="checkbox"/>
严重问题数	≤	0	<input checked="" type="checkbox"/>
一般问题数	≤	0	<input type="checkbox"/>
提示问题数	≤	0	<input type="checkbox"/>

保存

**步骤5** 完成后，单击“保存”。

您可单击“开始检查”执行检查任务并查看检查结果。

若问题级别数超过设定阈值，任务概览页面提示门禁结果为“不通过”，需要您根据检查结果优化代码。

----结束

## 5.3 配置代码检查任务定时执行

根据实际需要，设置代码检查任务的执行计划，定时执行默认分支的检查任务。

### 说明

- 建议在非流水线使用情况下进行配置。
- 配置当天时间间隔必须在5分钟以上。

### 配置执行计划

**步骤1** 基于项目入口[访问CodeArts Check服务首页](#)。

**步骤2** 进入代码检查页面，在任务列表中，单击任务名称。

**步骤3** 进入“代码检查详情”页面，选择“设置 > 执行计划”。

**步骤4** 开启“定时执行”开关，并设置自动执行时间。

图 5-2 配置执行计划

执行计划

定时执行

\* 自动执行时间:

日 一 二 三 四 五 六 00:00

保存

**步骤5** 完成后，单击“保存”。

----结束

## 5.4 配置代码检查任务消息通知

您可以根据实际需求，设置代码检查任务事件类型的通知方式。

例如，当检查任务的门禁失败时，可以设置以服务动态或邮件方式通知用户，并可配置通知的接收者角色和自定义成员。若不配置接收者角色及成员，门禁失败时向任务执行者发送通知；若配置，则向任务执行者、配置的接收者角色以及自定义成员发送通知。

通知设置支持项目级和任务级，入口分别如下：

- 项目级：进入项目详情页面，选择“代码 > 代码检查 > 配置中心 > 通知管理”。
- 任务级：进入代码检查详情页面，选择“设置 > 通知管理”。本节以任务级通知设置为例介绍操作步骤。

### 配置代码检查任务消息通知

**步骤1** 基于项目入口[访问CodeArts Check服务首页](#)。

**步骤2** 进入代码检查页面，在任务列表中，单击任务名称。

**步骤3** 进入“代码检查详情”页面，选择“设置 > 通知管理”。

**步骤4** 根据需要开启或关闭事件类型的通知状态。

- 开启服务动态通知，即通知当前任务所属项目的特定成员，可在消息中心查看消息。
- 开启邮件通知，即通知当前任务所属项目的特定成员，可在邮件中查看消息。

----结束

## 5.5 配置代码检查任务检查模式

您可以根据实际需求设置检查模式和检查目录，提高检查效率。

---

### 须知

C#语言和secbrella引擎均不支持MR合入。

---

### 设置检查模式

**步骤1** 基于项目入口[访问CodeArts Check服务首页](#)。

**步骤2** 进入代码检查页面，在任务列表中，单击任务名称。

**步骤3** 进入“代码检查详情”页面，选择“设置 > 检查模式”。

**步骤4** 在“MR检查模式”区域，选择“扫描变更文件”或“全量扫描”。默认选择“扫描变更文件”。




**步骤5** 完成后，单击“保存”。

----结束

## 设置检查目录

支持检查代码仓指定目录。

- 不配置检查目录时，检查代码仓中所有目录下的代码文件。
- 配置检查目录时，只检查配置目录中的文件。

检查目录 

输入需要检查的目录，多个目录使用逗号隔开

保存

### 说明

支持配置多个检查目录，多个目录需要用英文逗号分隔。例如：

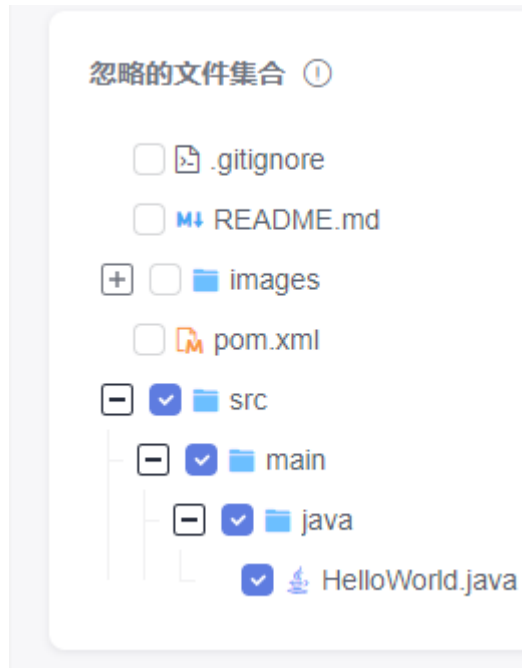
- 如果要检查dir1和dir2，检查目录配置为“dir1,dir2”。
- 如果要检查dir1目录下的dir3，检查目录配置为“dir1/dir3”。
- 配置项为目录且不支持文件，默认检查代码仓中所有代码。
- 配置的目录层级中不能包含"."开头的目录。
- 如果检查目录输入框路径为空，则默认全部检查。如果有指定检查目录，则只检查检查目录中的文件，如果指定的检查目录中有勾选[设置忽略文件](#)，则被忽略的文件不参与检查。
- 检查目录只生效于不需要编译的文件。例如，A文件编译后产生了B文件，则不会输出B文件上的告警。

## 设置忽略的文件集合

代码检查服务支持用户以任务为单元，设置代码检查的文件范围。

- 不勾选文件路径时，则检查所选仓库分支的所有代码。
- 勾选指定文件路径时，则该指定路径下的文件将排除检查。

在“忽略的文件集合”区域，勾选无需进行检查的文件路径。设置完成后，执行代码检查任务，将按照设置排除选中的文件路径执行检查。



#### 说明

勾选的忽略文件对需要编译的文件和不需要编译的文件均生效。

- 不需要编译的文件：勾选忽略文件后，被忽略的文件不参与检查。
- 需要编译的文件：例如A文件与B文件互为依赖，即使把A文件忽略，但A文件依然会参与检查前的编译过程，编译完成后不参与检查。

## 5.6 查看徽标状态

执行完代码检查任务后，您可通过徽标状态查看代码健康度，并标识在代码仓库中。

代码检查服务中设置各个告警级别的影响度如下：

致命：5

严重：3

一般：1

提示：0.1

健康度 =  $\text{SUM} \sum (\text{告警数量} * \text{告警级别影响度}) / \text{代码量}$ ，各个健康度对应的标识如下：

A: 健康度  $\leq 0.001$

B:  $0.001 < \text{健康度} \leq 0.005$

C:  $0.005 < \text{健康度} \leq 0.01$

D: 健康度  $> 0.01$

### 查看徽标状态

**步骤1** 基于项目入口[访问CodeArts Check服务首页](#)。

**步骤2** 在任务列表中，单击任务名称。

**步骤3** 进入“代码检查详情”页面，选择“设置 > 徽标状态”。

当前仅支持“HTML”和“Markdown”两种查看方式。



----结束

## 5.7 配置代码检查任务集成服务

当代码托管中有分支合并时，在代码检查服务中可以设置自动检查代码检查任务，并生成一个子任务，即代码仓库中有分支合并，就会重新生成一个代码检查子任务。

### 配置 MR 检查状态


**步骤1** 基于项目入口[访问CodeArts Check服务首页](#)。

**步骤2** 进入代码检查页面，在任务列表中，单击任务名称。

**步骤3** 进入“代码检查详情”页面，选择“设置 > 集成服务 > CodeArts Repo”。


**步骤4** 选择是否勾选“MR检查状态”，勾选即表示开启代码评审（MR）的自动化检查。

**步骤5** 当所在任务的代码仓库中有分支合并时，代码检查服务会自动生成一个代码检查子任务。

1. 进入代码检查任务详情页面。
2. 单击，在代码托管中新建MR请求。

#### 说明

合并分支后，根据您已选择的[检查模式](#)，对代码分支进行差异部分或全量的检查。

3. 单击 master ▾，通过下拉框中，可查看所有的MR触发代码检查的历史记录。

----结束

### 配置代码提交时执行

**步骤1** 勾选“代码提交时执行”，表示对代码仓提交代码时会触发执行代码检查任务。

#### 说明

仅对已创建任务的分支有效。

**步骤2** 单击“保存”。

----结束

## 5.8 配置代码检查任务自定义环境

当使用代码检查服务时，若常用的编译构建环境缺少您需要的依赖包、工具，无法满足您的使用需求时，您可通过自定义构建环境提供的基础镜像执行代码检查任务。

### 前提条件

已将自定义环境推送到镜像仓中。

### 配置自定义镜像

- 步骤1** 基于项目入口[访问CodeArts Check服务首页](#)。
- 步骤2** 进入代码检查页面，在任务列表中，单击任务名称。
- 步骤3** 在代码检查详情页面，选择“设置 > 自定义环境”。
- 步骤4** 在“自定义镜像”中，输入以“域名称/组织名称/仓名称:tag名称”格式的内容或容器镜像服务SWR内镜像链接。
- 步骤5** 完成后，单击“保存”。


----结束

## 5.9 配置代码检查任务高级选项

当前“亚太-新加坡”、“土耳其-伊斯坦布尔”和“中东-利雅得”局点不支持该功能。

### 配置新问题起始时间

在每个任务的基础上可以设置新问题起始时间。起始时间设置之后要重新扫描，问题的发现时间大于设置时间将被划分为新问题。如果没有设置起始时间，连续两次的检查结果对比，新增的就是新问题数。

- 步骤1** 基于项目入口[访问CodeArts Check服务首页](#)。
- 步骤2** 进入代码检查页面，在任务列表中，单击任务名称。
- 步骤3** 进入“代码检查详情”页面，选择“设置 > 高级选项”。
- 步骤4** 在“新问题起始时间”区域，单击设置新问题起始时间。

#### 说明

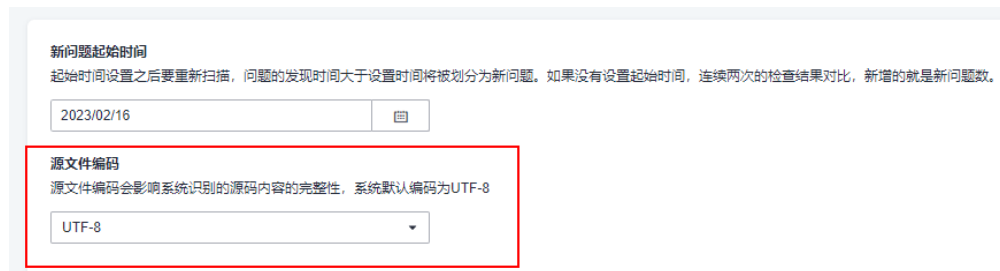
默认起始时间为当前任务第一次扫描成功的时间，起始时间之后扫描出的所有问题都将被划分为新问题。

----结束

### 配置源文件编码

设置文件的编码方式，源文件编码会影响系统识别的源码内容的完整性，系统默认编码为UTF-8。

**步骤1** 在“源文件编码”区域，单击下拉框选择源文件编码。



**新问题起始时间**  
起始时间设置之后要重新扫描，问题的发现时间大于设置时间将被划分为新问题。如果没有设置起始时间，连续两次的检查结果对比，新增的就是新问题数。

2023/02/16

**源文件编码**  
源文件编码会影响系统识别的源码内容的完整性，系统默认编码为UTF-8

UTF-8

**步骤2** 完成后，提示修改成功。

----结束



# 6 执行代码检查任务

用户可以对目标代码检查任务执行检查，及时发现源代码中的代码问题。

执行代码检查有两种方式：

- 手动执行：创建完代码检查任务后手动执行代码检查任务。本章节介绍的为手动执行代码检查任务的操作指导。
- 定时执行：多用于日常执行代码检查，配置详情可参考[配置代码检查任务定时执行](#)。

## 前提条件

- 已[创建代码检查任务](#)。

## 执行代码检查任务

**步骤1** 基于项目入口[访问CodeArts Check服务首页](#)。



**步骤2** 单击代码检查任务所在行的 ，根据页面提示等待任务执行完成。

----结束


## 检查代码仓其它分支的代码

在代码检查详情页面可以手动切换“已检查分支列表”和“未检查分支列表”进行检查。

**步骤1** 单击代码检查任务名称。

**步骤2** 在“代码检查详情”页面，默认显示“概览”页签，单击  master  可以切换不同分支。

**步骤3** 切换目标分支后，单击“开始检查”，等待代码检查任务执行完成。

再次单击代码检查任务所在行的  执行检查任务时，检查的代码分支会自动变更为创建检查任务时设置的默认分支。

----结束

# 7 查看代码检查结果

## 前提条件

您已[创建并执行代码检查任务](#)。

## 查看代码检查详情

**步骤1** 基于项目入口[访问CodeArts Check服务首页](#)。

**步骤2** 进入代码检查页面，搜索目标任务。在任务列表页，可以查看检查任务的状态。

- 新问题：表示本次代码检查任务执行后新增的问题。
- 未解决：表示该代码检查任务未解决的问题。
- 已解决：表示该代码检查任务已解决的问题。

**步骤3** 单击检查任务名称，进入代码检查详情概览页，可查看的信息如下：

- 原始代码行数。
- 门禁结果：检查结果分为“通过”和“不通过”，同时会展示“致命问题数”和“严重问题数”。

代码检查是否通过是由设置的质量门禁值判断，门禁值设置请参考[配置代码检查任务质量门禁](#)。

- 问题数：检查出来的代码质量问题数。
- 延迟上线问题数：由于检查引擎的能力提升，出现的新问题数。
- 未解决新问题数：当前的代码检查结果中，未解决的问题个数。
- 已解决问题数：当前的代码检查结果中，已解决的问题个数
- 代码平均圈复杂度：代码的平均[圈复杂度](#)。
- 代码重复率：代码的[重复率](#)。
- NBNC代码行：非空非注释的代码行数。
- 问题数最多Top10检查规则：问题数Top10的检查规则以及每条规则对应的问题数。
- 待处理问题严重程度：待处理的各个等级问题数量分布情况。
- 问题指派分配：检查出来的代码问题修改分配情况。如果已指派的数量与问题数不一致，说明存在某代码问题还未分配修改责任人。

- 历史趋势：默认展示近一个月内该代码检查任务的检查结果变化趋势，包括问题数、圈复杂度和重复率。

----结束

## 查看代码问题

可查看代码问题详情有两种查看方式：导出至本地和在线查看。

- 导出至本地。

方式一：

- 返回至代码检查任务列表页，勾选代码检查任务名称前的复选框，可选择的任务数为1~20。
- 选择“更多操作 > 导出任务告警”。
- 在弹出的窗口中选择需要导出的选项，单击“开始导出”。
- 单击“查看下载”，即可查看导出进度。
- 待导出完成后，单击“下载”，可将代码检查问题信息导出至本地。


方式二：

- 单击“代码问题”页签。
- 选择“全量操作 > 全量导出”，可将代码检查问题信息导出至本地。

- 在线查看。

- 单击“代码问题”页签，可以在“过滤器”区域，根据过滤条件筛选问题，可设置的过滤条件如下表。

过滤条件	说明
问题级别	分为致命、严重、一般、提示四个级别。
问题状态	包括待处理、已解决、已忽略、已修复四种状态。
延迟问题	查看当前检查出的问题中正式版本问题总数和延迟上线版本问题总数。
检测时间	根据检测时间区分新问题和存量问题。 <b>说明</b> 设置新问题起始时间，问题发现时间在起始时间之后的则为新问题。
文件名	按照源文件名称筛选代码问题。
语言	按照检查的语言筛选代码问题。
规则	按照问题命中的规则筛选代码问题，便于用户一次性解决同一类问题。
标签	按照规则的用户标签筛选代码问题。
负责人	按照问题分配负责人筛选代码问题。 <b>说明</b> 代码检查过程中，开启精准匹配自动将新问题分配给问题代码行的最后一次提交者。
CWE	按照安全类规则名筛选代码问题。

- b. 设置完筛选条件后，在右侧区域对检查问题可进行以下操作。
- 如果该问题已解决，单击“待处理”选择“已解决”；如果该问题无需处理，单击“待处理”选择“忽略问题”，忽略问题时需提交评论。
  - 单击“负责人”，可将此问题提交至其他负责人。
  - 单击“问题帮助”，可查看该问题对应的规则详情。
  - 单击“查看上下文”，在问题详情页面中，可修改问题状态、转移负责人和根据“问题帮助”单击  前往代码仓对代码进行修改。
  - 单击“创建问题单”，提交问题单给指定的责任人。该功能仅Scrum类型项目支持。
  - 单击“生成报告”，可将代码问题生成PDF格式的文件报告。
  - 单击“全量操作 > 全量修改问题状态”，可修改筛选出来的全部问题的问题状态。

#### 说明

如果代码问题数是0，则不显示该内容。

- 单击“全量操作 > 全量转移责任人”，可将筛选出来的全部问题转移给指定责任人。


## 查看代码度量

代码度量可准确的反映提交代码的质量，助力用户及时发现并修改问题，提升研发效能。代码度量包括已下两种类型的问题：

- 圈复杂度：代码的复杂度。

#### 说明

扫描Shell语言代码时，度量结果不支持圈复杂度计算。

单击文件名称进入文件详情页面，单击  可进入到对应的代码仓修改代码。

```
worker/src/main/java/worker/Worker.java
1 package worker;
2
3 import redis.clients.jedis.Jedis;
4 import redis.clients.jedis.exceptions.JedisConnectionException;
5 import java.sql.*;
6 import org.json.JSONObject;
7
8 class Worker {
9     public static void main(String[] args) {
10        try {
11            Jedis redis = connectToRedis("redis");
12            Connection dbConn = connectToDB("db");
13
14            System.err.println("Watching vote queue");
15
16            while (true) {
17                String voteJSON = redis.blpop(0, "votes").get(1);
18                JSONObject voteData = new JSONObject(voteJSON);
19                String voterID = voteData.getString("voter_id");
20                String vote = voteData.getString("vote");
21
22                System.err.printf("Processing vote for '%s' by '%s'\n", vote, voterID);
23                updateVote(dbConn, voterID, vote);
24            }
25        }
26    }
27 }
```

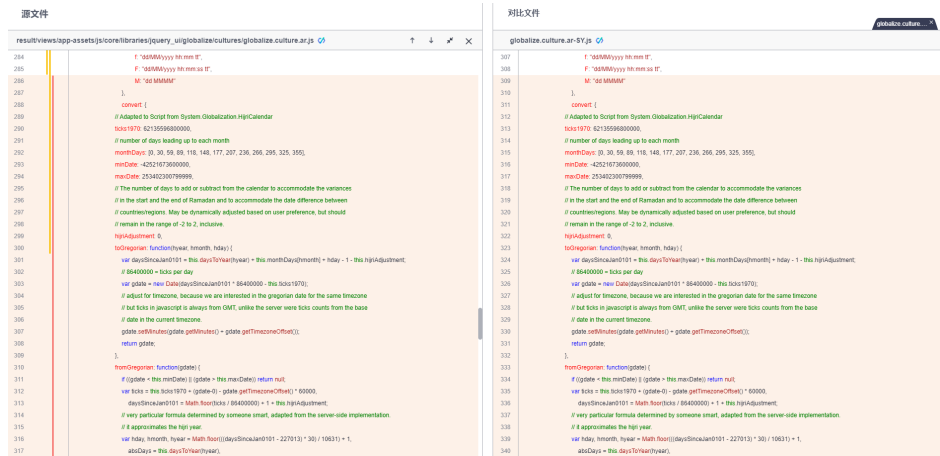
- 重复率：代码检查服务通过重复块、重复行、重复率来识别被测代码的重复情况。


修改重复率问题操作指导如下：

- a. 单击文件名称进入文件详情页面。
- b. 文件详情页面左侧的竖线标识了重复的代码块，单击标识竖线，将会显示与该段代码重复的文件列表。

```
result/views/app-assets/js/core/libraries/jquery_ui/globalize/cultures/globalize.culture.ar.js
319 // year is just approximate, it may need adjustment up or down by 1.
320 if (daysSinceJan0101 < absDays) {
321     hyear--;
322     absDays -= daysInYear;
323 }
324 else if (daysSinceJan0101 === absDays) {
325     hyear--;
326     absDays = this.daysToYear(hyear);
327 }
328 else {
329     if (daysSinceJan0101 > (absDays + daysInYear)) {
330
331         Duplicated By
332         result/views/app-assets/js/core/libraries/jquery_ui/globalize/cultures/globalize.culture.a...
333         Lines: 309 - 382
334
335         result/views/app-assets/js/core/libraries/jquery_ui/globalize/cultures/globalize.culture.a...
336         Lines: 309 - 382
337
338         result/views/app-assets/js/core/libraries/jquery_ui/globalize/cultures/globalize.culture.a...
339         Lines: 92 - 165
340
341         result/views/app-assets/js/core/libraries/jquery_ui/globalize/cultures/globalize.culture.a...
342         Lines: 92 - 165
343
344         hmonth++;
345     }
346     hmonth--;
347     hday = daysInYear - this.monthDays(hmonth);
348     return [hyear, hmonth, hday];
349 },
350 daysToYear: function(year) {
351     // calculates how many days since Jan 1, 0001
352     var yearsToYear30 = Math.floor((year - 1) / 30) * 30,
353         yearsInto30 = year - yearsToYear30 - 1,
354         days = Math.floor((yearsToYear30 * 10631) / 30) + 227013;
```

- c. 单击文件名称，可对比查看两个文件的重复代码块。



d. 单击  可进入到对应的代码仓修改代码。

## 查看检查日志

单击“检查日志”页签，可查看检查步骤日志和检查参数。

图 7-1 步骤日志

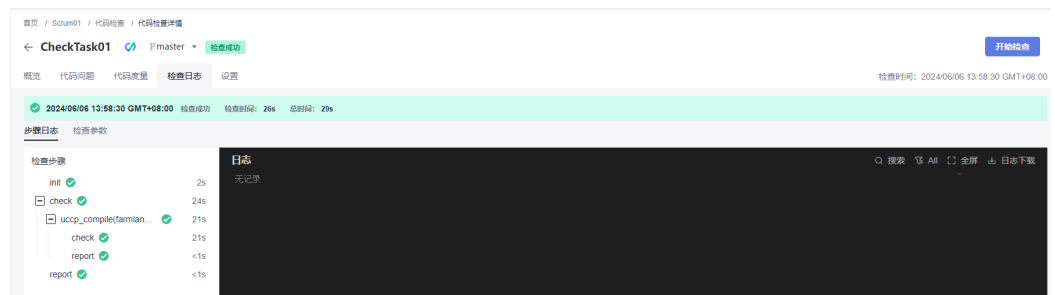
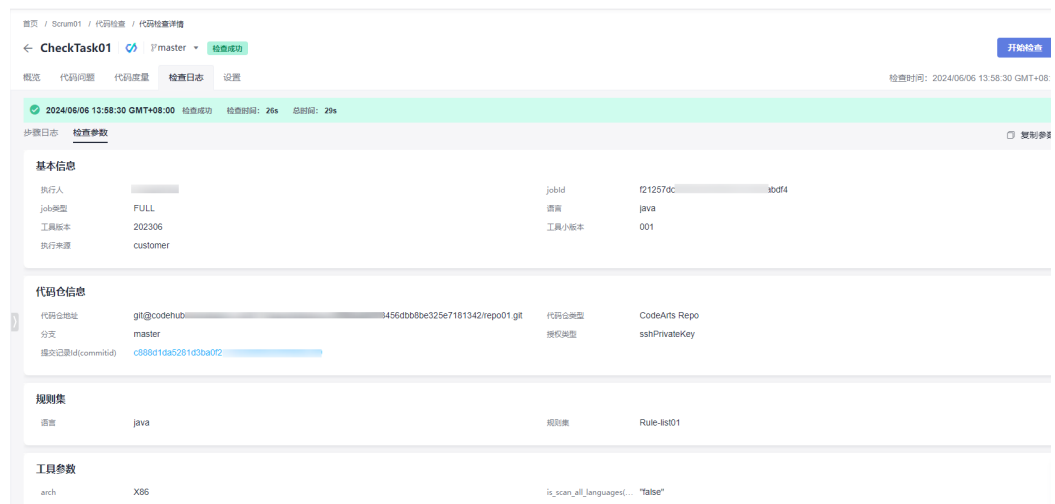


图 7-2 检查参数



# 8 查询审计日志

云审计服务是安全解决方案中专业的日志审计服务，记录了CodeArts Check的相关操作事件，方便您日后的查询、审计和回溯。

## 支持审计日志的操作

表 8-1 云审计服务支持 CodeArts Check 服务操作

操作名称	资源类型	事件名称
创建代码检查任务	task	create
修改代码检查任务	task	modify
删除代码检查任务	task	delete
执行代码检查任务	job	create
取消执行代码检查任务	job	modify_job
关注代码检查任务	followingTask	create_followingTask
取消关注代码检查任务	followingTask	delete_followingTask
修改规则集信息	ruleset	modify_ruleset
修改规则集的规则	ruleset	modify_ruleset
删除规则集	ruleset	delete_ruleset
配置默认规则集	ruleset	set_default_ruleset
导出规则集表格	ruleset	export_ruleset
导出问题表格	defectExcel	export_defectExcel
下载问题表格	defectExcel	downloadAndExport_defectExcel
修改问题状态	issueStatus	update_issueStatus
修改问题责任人	issueOwner	update_issueOwner

操作名称	资源类型	事件名称
修改问题评论	issueComment	update_issueComment

## 查看审计日志

用户需要在云审计服务CTS的管理控制台查询CodeArts Check服务的事件列表。详情请参考[查看审计事件](#)。