CodeArts IDE Online

最佳实践

文档版本	07
发布日期	2023-02-16

华为云计算技术有限公司

版权所有 © 华为云计算技术有限公司 2023。 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

1 基于 CodeArts IDE Online 快速开发、部署微服务.....1
1.1 概要.....1
1.2 准备工作.....1
1.3 开发调试微服务.....2
1.4 制作微服务容器镜像.....4
1.5 部署微服务容器镜像.....6
2 基于 CodeArts IDE Online、TensorFlow 和 Jupyter Notebook 开发深度学习模型.....9
2.1 概要.....9
2.2 准备工作.....9
2.3 导入和预处理训练数据集.....11
2.4 创建和训练模型.....13
2.5 使用模型.....13

1 基于 CodeArts IDE Online 快速开发、部署微服务

1.1 概要

本章节主要讲述在CodeArts IDE Online中如何快速将开发好的微服务部署到华为云容器服务并进行快速验证的过程。

1.2 准备工作

开通容器镜像服务

登录华为云**容器镜像服务**并申请开通获取使用权限。

创建容器镜像组织

1. 进入容器镜像服务控制台。
2. 在左侧导航栏选择“组织管理”，单击右上角“创建组织”。在弹出的页面中填写“组织名称”，然后单击“确定”完成组织创建。

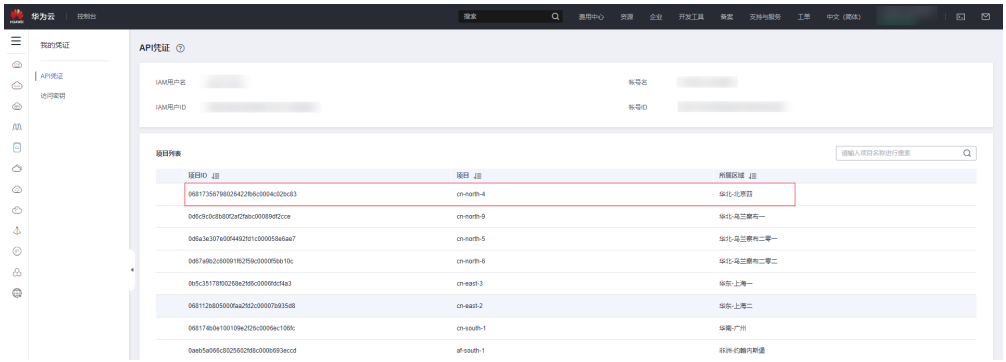
获取镜像仓库访问地址、区域项目名称

操作步骤如下：

1. 访问**我的凭证**。
2. 在“项目列表”区域查找当前区域对应的项目。

版权所有 © 华为云计算技术有限公司

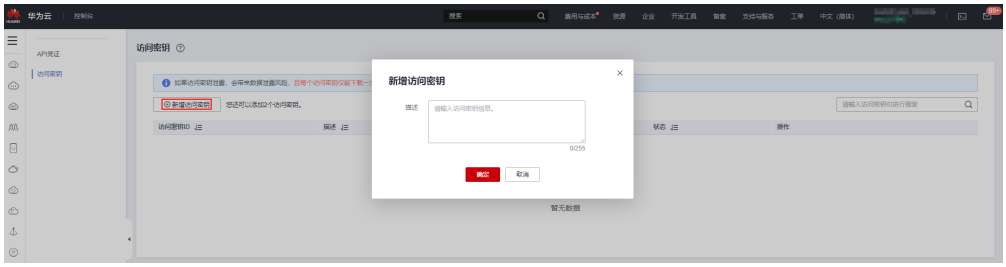
镜像仓库地址为：**swr.区域项目名称.myhuaweicloud.com**，如中国华北区4对应的镜像仓库地址为：**swr.cn-north-4.myhuaweicloud.com**。



获取 AK/SK 访问密钥

如果已有无需重复获取，操作步骤如下：

1. 访问**我的凭证**。
2. 在左侧导航栏选择“访问密钥”，单击“新增访问密钥”。在弹出的页面中填写描述信息，然后单击“确定”创建新的访问密钥。
3. 在“管理访问密钥”页签，单击“新增访问密钥”，创建新的访问密钥。



4. 输入当前用户的登录密码，并通过邮箱或手机进行验证。

说明

在统一身份认证服务中创建的用户，如果创建时未填写邮箱或者手机号，则只需校验登录密码。

5. 创建成功后，单击“立即下载”，下载访问密钥excel文件。

1.3 开发调试微服务

创建并启动 IDE 实例

1. **登录CodeArts IDE Online**，如果提示未开通则单击“OK”跳转至开通页面完成服务开通，成功登录后的页面如下：

版权所有 © 华为云计算技术有限公司

2. 单击“新建实例”，进入IDE实例模板配置页面并完成配置，详细操作可参见快速入门，已有实例并完成载入了项目工程则可以跳过该步骤。

   动图演示如下：

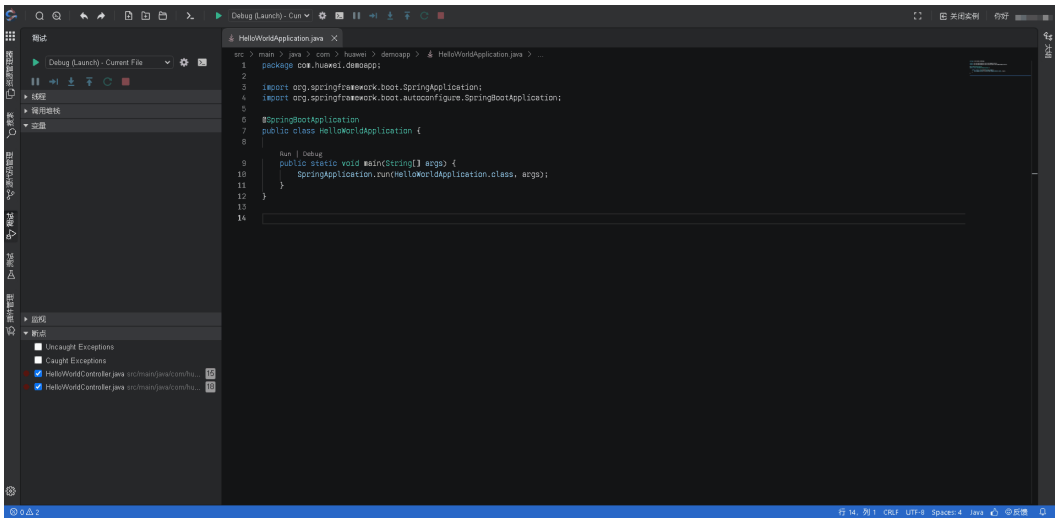
   针对已有实例，则直接单击实例列表的启动按钮进入实例页面。

版权所有 © 华为云计算技术有限公司

开发调试微服务代码

代码编写完毕，单击左侧Debug页签中的“Start Debug”或Main函数中的“Debug”启动工程，在代码文件行号前鼠标左键单击打上断点，完成监听端口的外部访问权开通即可边调试边在浏览器中查看代码修改的应用效果。

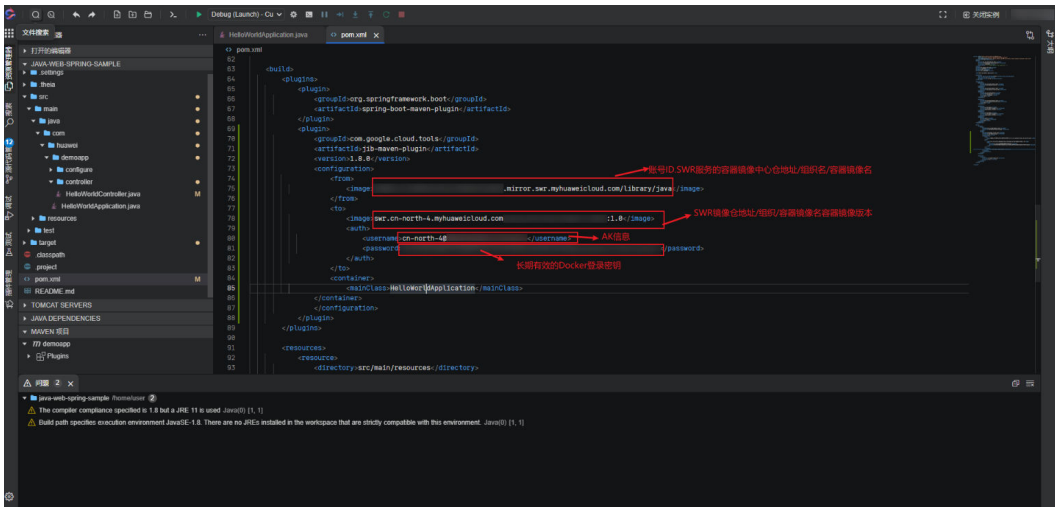
动图演示如下：



1.4 制作微服务容器镜像

编辑 pom 文件增加谷歌 jib 插件配置

打开微服务工程中的“pom.xml”文件，在plugins标签下增加如下图内容：



jib的maven配置格式：

```
<plugin>
    <groupId>com.google.cloud.tools</groupId>
    <artifactId>jib-maven-plugin</artifactId>
    <version>1.8.0</version>
    <configuration>
        <from>
            <image>用户账户ID.mirror.swr.myhuaweicloud.com/library/java</image>
```

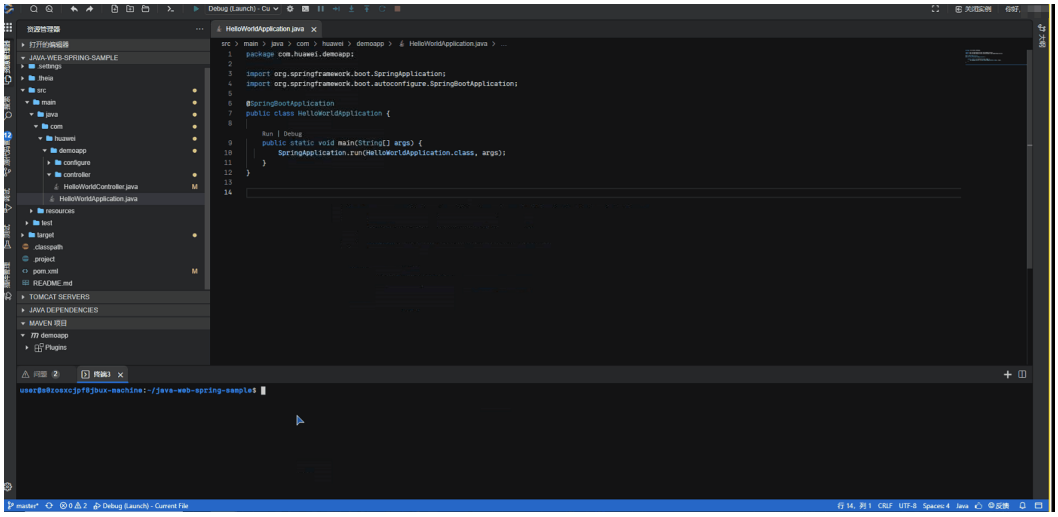

版权所有 © 华为云计算技术有限公司

```
        </from>
        <to>
            <image>swr.区域项目名称.myhuaweicloud.com/个人在SWR中创建的组织/springboot-demo:1.0</
image>
            <auth>
                <username>区域项目名称@申请的AK</username>
                <password>获取到的长期有效的Docker登录密钥</password>
            </auth>
        </to>
        <container>
            <mainClass>微服务入口类</mainClass>
        </container>
    </configuration>
</plugin>
```

- 用户帐号ID、区域项目名称、AK/SK信息可通过访问**我的凭证**获取。
- 长期有效的Docker登录密钥：可通过在Terminal中执行命令如下获取，其中，*$AK*、*$SK*替换成**准备工作**章节获取到的真实AK/SK。
  printf "*$AK*" | openssl dgst -binary -sha256 -hmac "*$SK*" | od -An -vtx1 | sed 's/[\n]//g' | sed 'N;s/\n//'
- 个人在SWR中创建的组织：需要个人**登录容器镜像服务**事先完成创建。
- “from”标签中的image标签内容为源镜像的地址，其中，“library”为容器镜像服务中镜像中心的组织名，该组织下分享了很多官方容器镜像，“java”为源镜像名称。

编译打包并向 SWR 推送微服务镜像

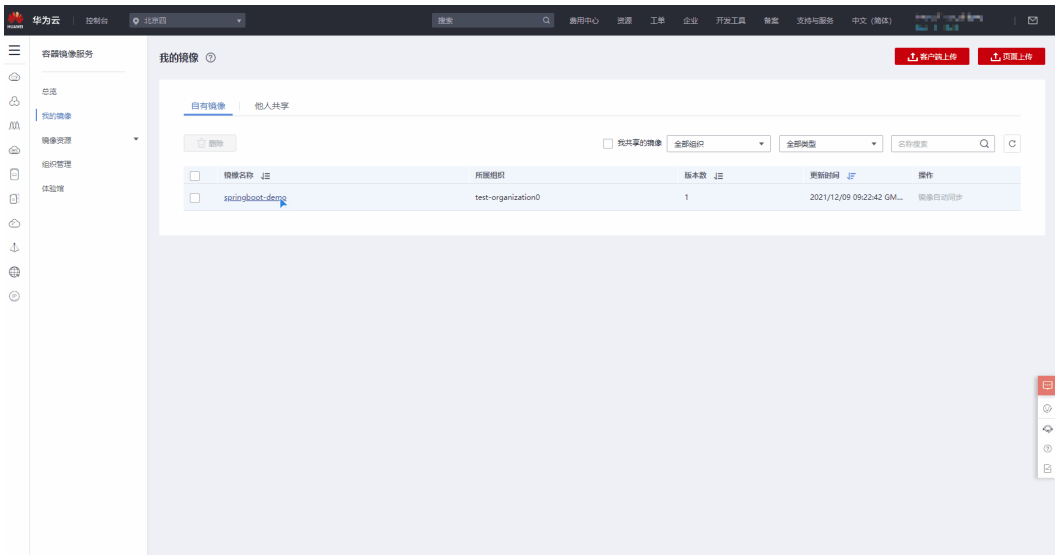
完成jib配置后，在Terminal中输入“mvn package -Dmaven.test.skip=true jib:build”并回车，动图演示如下：



登录 SWR 验证微服务镜像推送结果

登录SWR服务，验证查询容器镜像是否完成推送，动图演示如下：

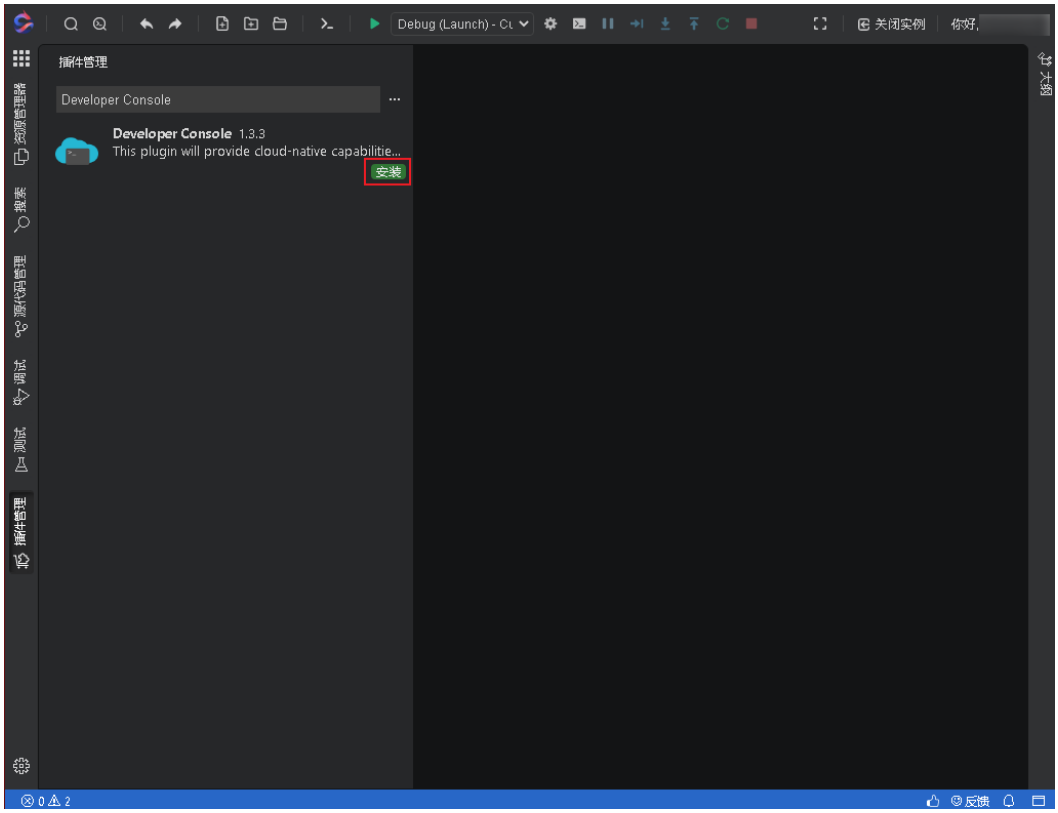
版权所有 © 华为云计算技术有限公司



1.5 部署微服务容器镜像

安装 Developer Console 插件

打开插件市场，搜索到“Developer Console”后，单击“Install”即可完成安装，动图演示如下：



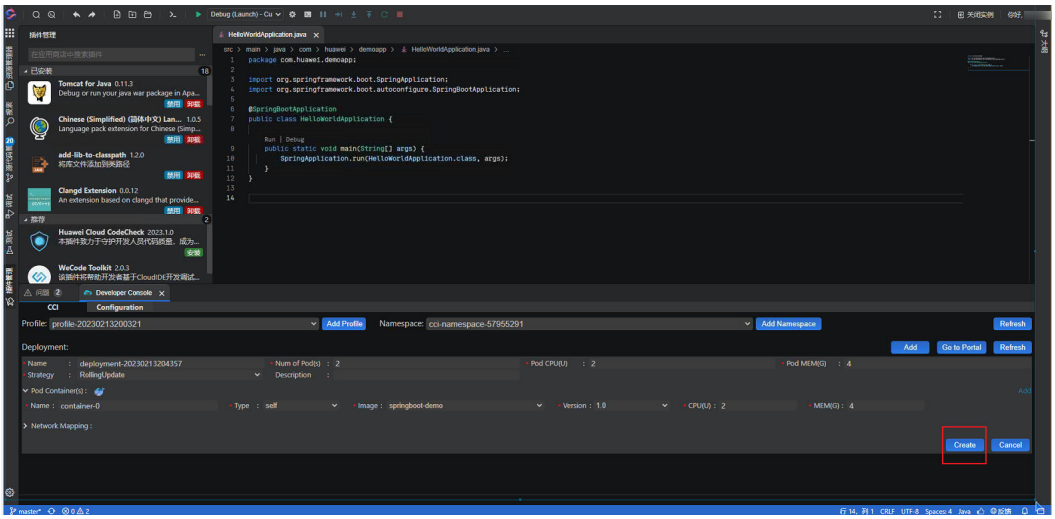
版权所有 © 华为云计算技术有限公司

部署微服务到 CCI 容器实例

步骤1 配置Profile

Profile是一组API调用鉴权所需配置的抽象，Profile中的相关信息可通过访问**我的凭证**获取，有了这些信息就可以获取到镜像列表、vpc列表、cci负载/命名空间等数据。

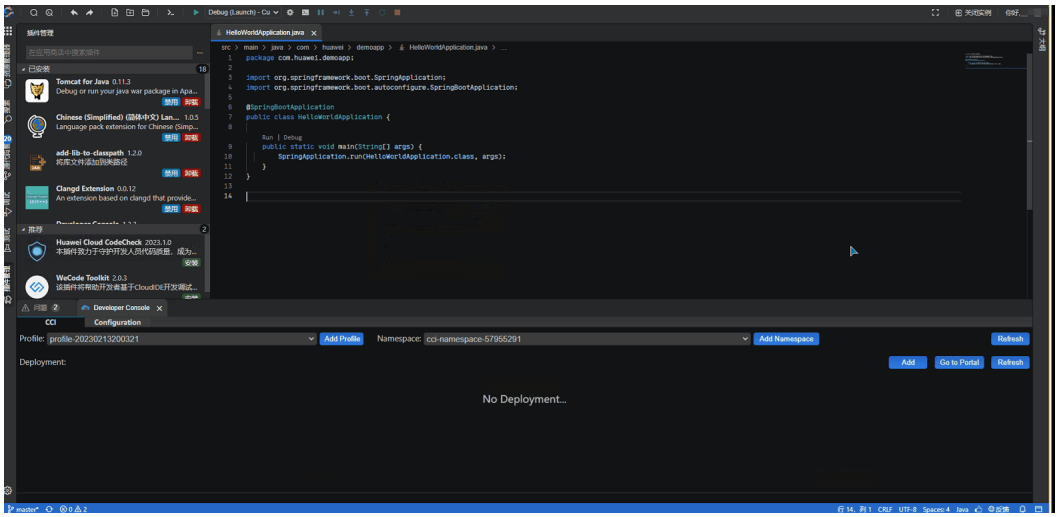
- name：用于标识区分不同的Profile，只能输入大小写英文字母、下划线、中划线。
- ProjectID：对应项目ID。
- ProjectName：对应项目名称的英文。
- Region：将控制台切换到区域项目所在region，网页URL中的region字段，如：cn-north-4。
- UserID：用户ID。
- UserName：用户名。
- AK/SK：下载管理访问密钥页面中密钥excel文件，注意该文件的下载只有首次添加可以下载。



步骤2 添加Namespace

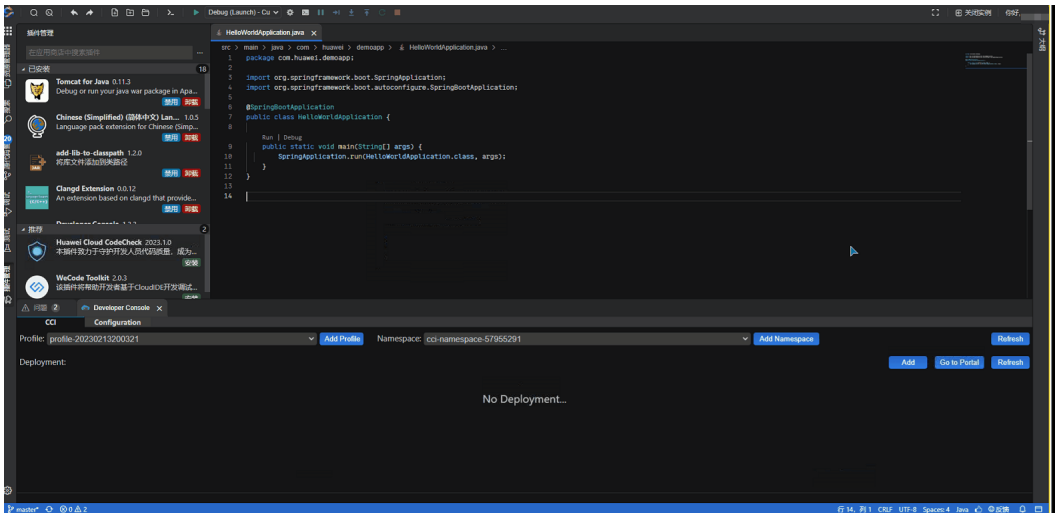
- 选择不同的Profile，Namspace列表不尽相同，每一组profile对应不同的鉴权信息。
- **CCI命名空间参考文档**。

版权所有 © 华为云计算技术有限公司



步骤3 创建Deployment

1. 选择Profile和Namespace后，单击“Add”会出现deployment的配置输入，如果需要配置更多，则可以单击“Go to Portal”跳转至CCI页面进行创建。
   - 如果Pod中的镜像启动需要监听端口并且需要开放外部访问权时，则需要增加容器端口跟elb端口的映射关系，其中elb的端口一定要填写未被占用的。
   - 如果Pod中的镜像启动无监听端口或者不需要开发外部访问权时，则网络映射配置中的“Type”选择“Not enable”。
2. 创建完成后，单击“Upgrade”将在IDE中直接升级负载，放弃升级则单击“Cancel”，如果希望跳转到CCI页面进行升级，则只需要单击与“Upgrade”平级的“Go to Portal”即可。
3. 单击“Delete”，当前deployment则会被立即删除并附带删除配置的service数据。



----结束

版权所有 © 华为云计算技术有限公司

2 基于 CodeArts IDE Online、TensorFlow 和 Jupyter Notebook 开发深度学习模型

2.1 概要

本章节主要讲解如何在CodeArts IDE Online中使用TensorFlow和Jupyter Notebook完成神经网络模型的训练，并利用该模型完成简单的图像分类。

2.2 准备工作

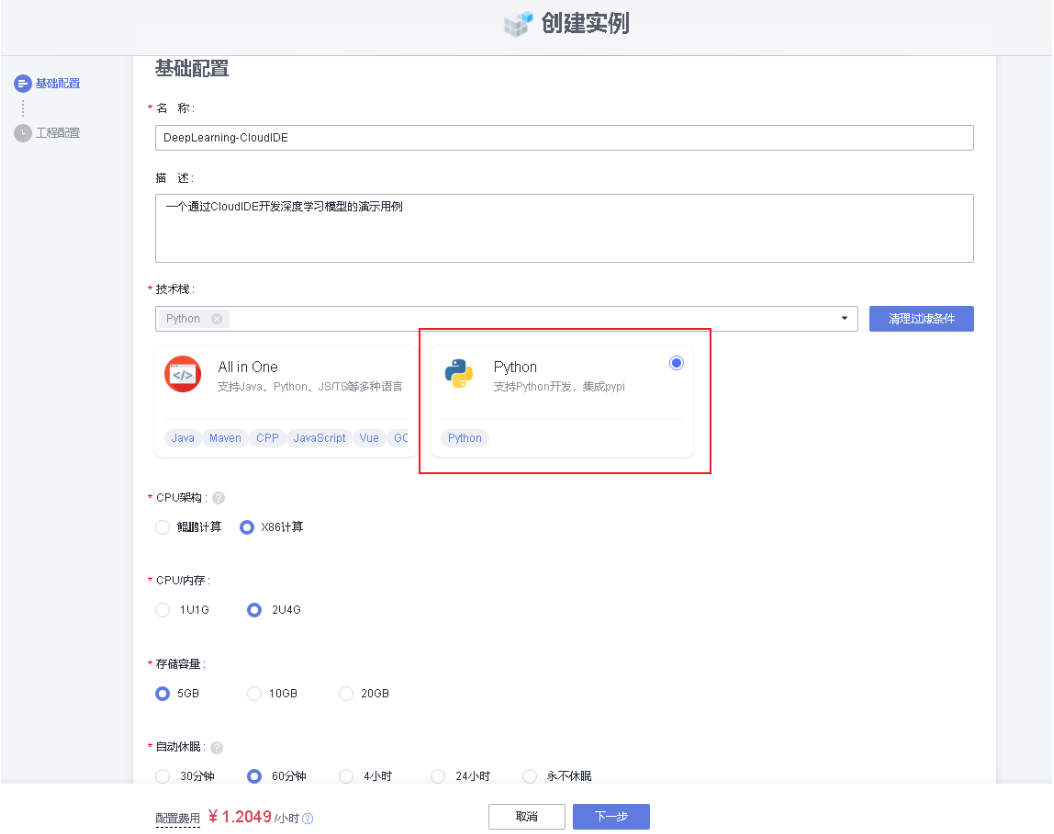
打开 IDE 实例

步骤1 登录CodeArts IDE Online，单击“创建实例”。

如果提示未开通则根据提示跳转至开通页面完成服务开通。

步骤2 进入“基础配置”页面，选择Python技术栈，CPU架构选择X86计算，CPU/内存选择2U4G，单击“下一步”。

版权所有 © 华为云计算技术有限公司



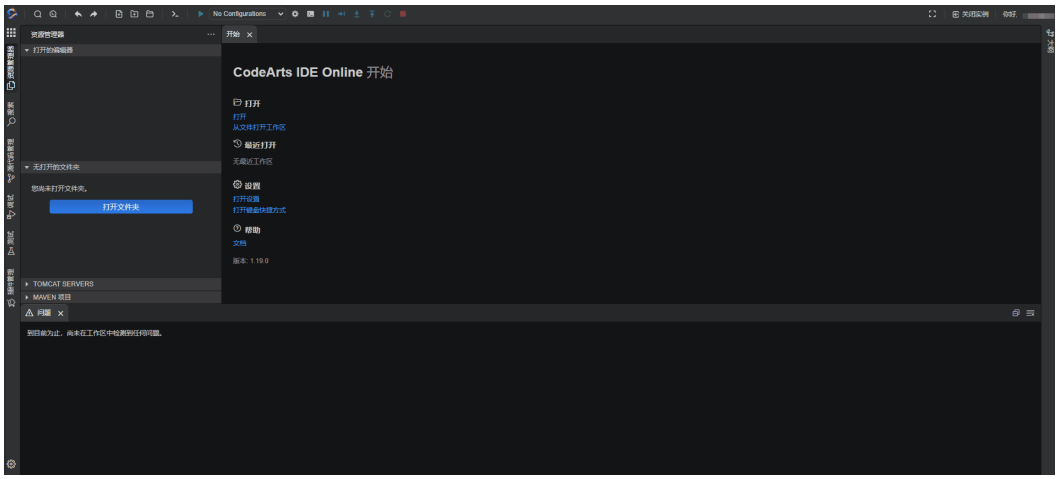
步骤3 进入“工程配置”页面，选择不创建工程，然后单击“确定”，完成实例创建。

----结束

安装 TensorFlow

步骤1 进入CodeArts IDE Online实例，创建并打开一个空白工作目录，命令如下。

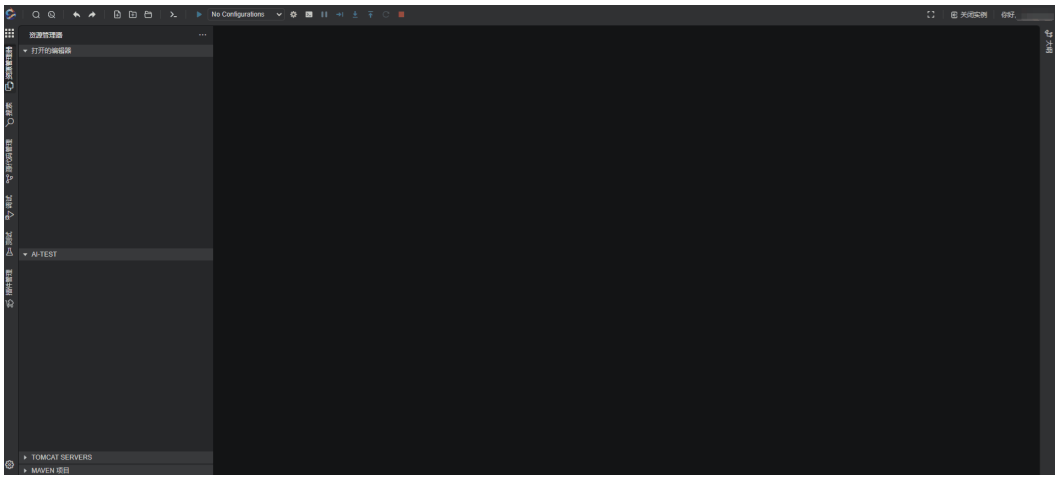
```
mkdir ai-test
```



步骤2 使用pip安装TensorFlow等依赖包，为加快安装速度此处安装的是tensorflow-cpu，命令如下。

```
python3 -m pip install tensorflow-cpu matplotlib numpy
```

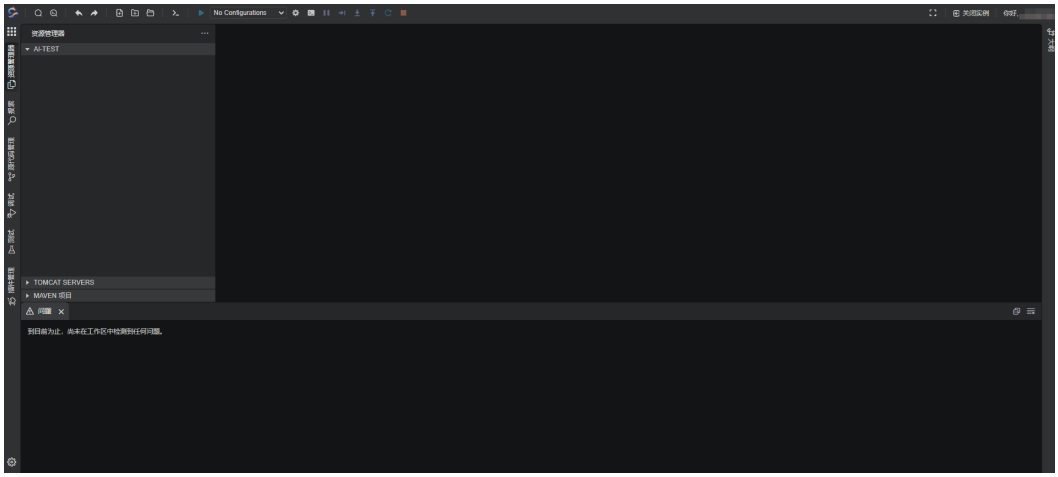

版权所有 © 华为云计算技术有限公司



----结束

进入 Notebook Editor

打开Jupyter Notebook（文件后缀名为“ipynb”），打开该文件后会出现一个Notebook Editor，可以在里面编辑和运行cell。



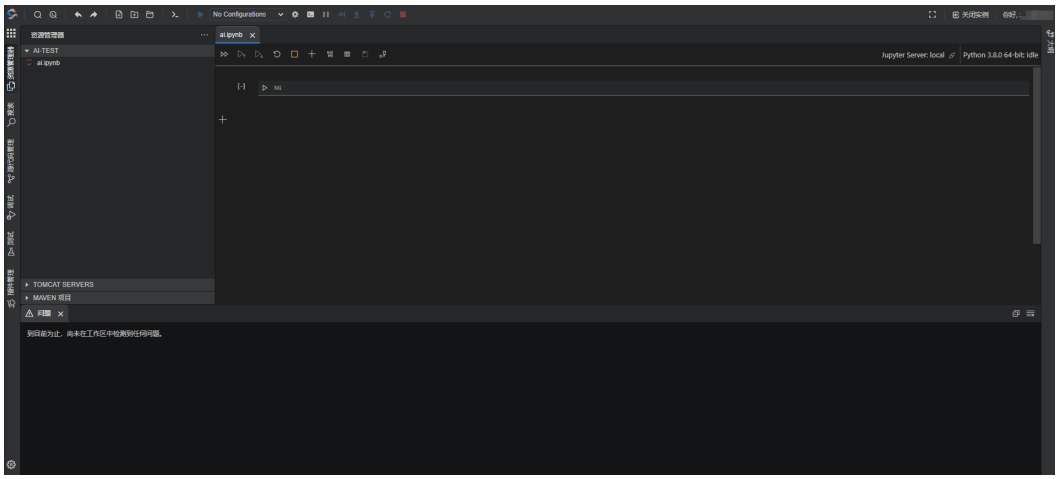
2.3 导入和预处理训练数据集

参考**TensorFlow官网的教程**，创建一个简单的图片分类模型。

查看当前TensorFlow版本，单击或者敲击Shift+Enter运行cell。

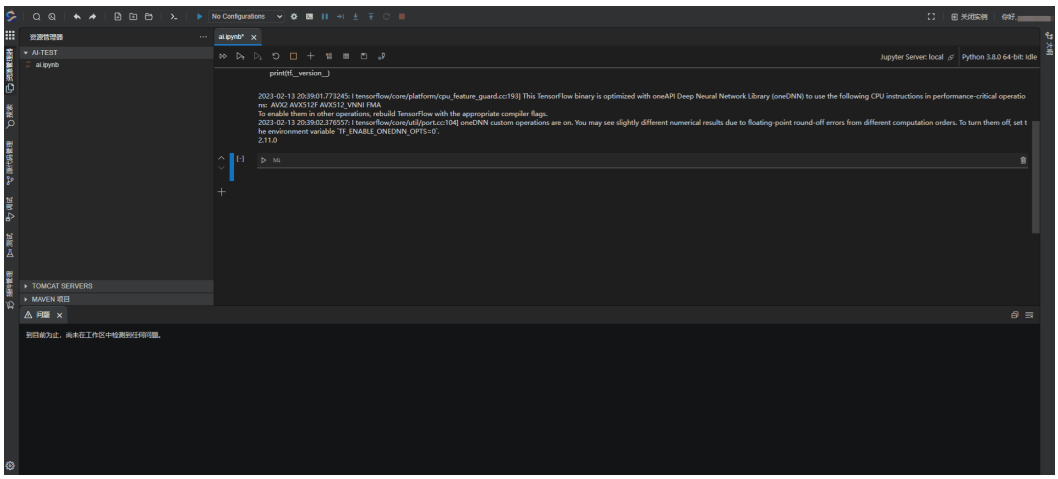
```
from __future__ import absolute_import, division, print_function, unicode_literals

# TensorFlow and tf.keras
import tensorflow as tf
from tensorflow import keras
# Helper libraries
import numpy as np
import matplotlib.pyplot as plt
# print tensorflow version
print(tf.__version__)
```


版权所有 © 华为云计算技术有限公司



下载Fashion MNIST图片数据集，该数据集包含了10个类型共60000张训练图片以及10000张测试图片。

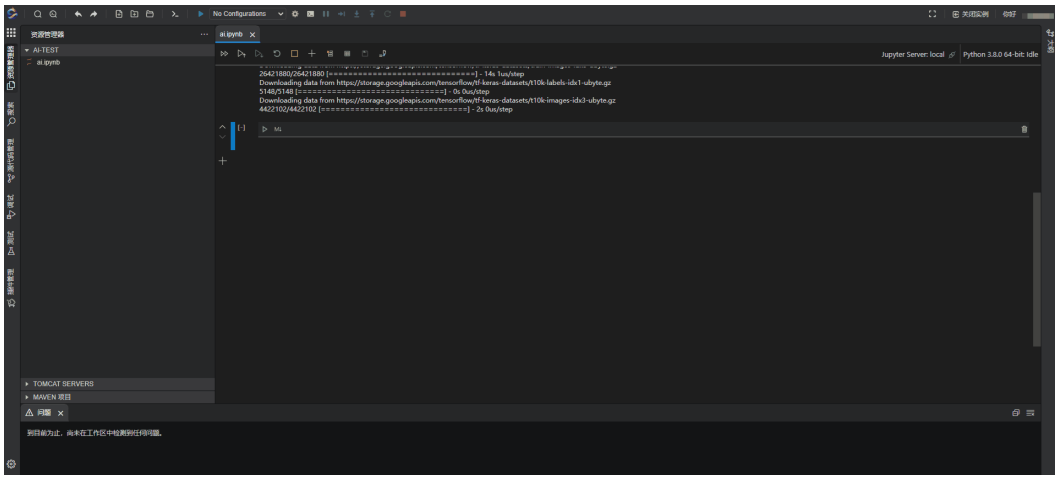
```
# download Fashion MNIST dataset
fashion_mnist = keras.datasets.fashion_mnist
(train_images, train_labels), (test_images, test_labels) = fashion_mnist.load_data()
```



对训练数据做预处理，并查看训练集中最开始的25个图片。

```
class_names = ['T-shirt/top', 'Trouser', 'Pullover', 'Dress', 'Coat',
               'Sandal', 'Shirt', 'Sneaker', 'Bag', 'Ankle boot']
# preprocessing
train_images = train_images / 255.0
test_images = test_images / 255.0
# display first 25 images
plt.figure(figsize=(10,10))
for i in range(25):
    plt.subplot(5,5,i+1)
    plt.xticks([])
    plt.yticks([])
    plt.grid(False)
    plt.imshow(train_images[i], cmap=plt.cm.binary)
    plt.xlabel(class_names[train_labels[i]])
plt.show()
```

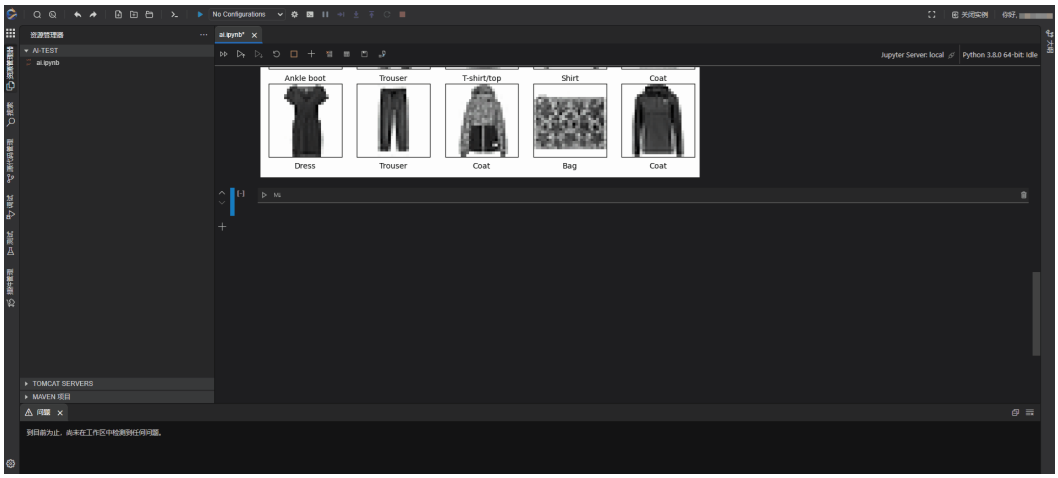

版权所有 © 华为云计算技术有限公司



2.4 创建和训练模型

命令如下：

```
# create model
model = keras.Sequential([
    keras.layers.Flatten(input_shape=(28, 28)),
    keras.layers.Dense(128, activation='relu'),
    keras.layers.Dense(10)
])
model.compile(optimizer='adam',
              loss=tf.keras.losses.SparseCategoricalCrossentropy(from_logits=True),
              metrics=['accuracy'])
# training
model.fit(train_images, train_labels, epochs=10)
```



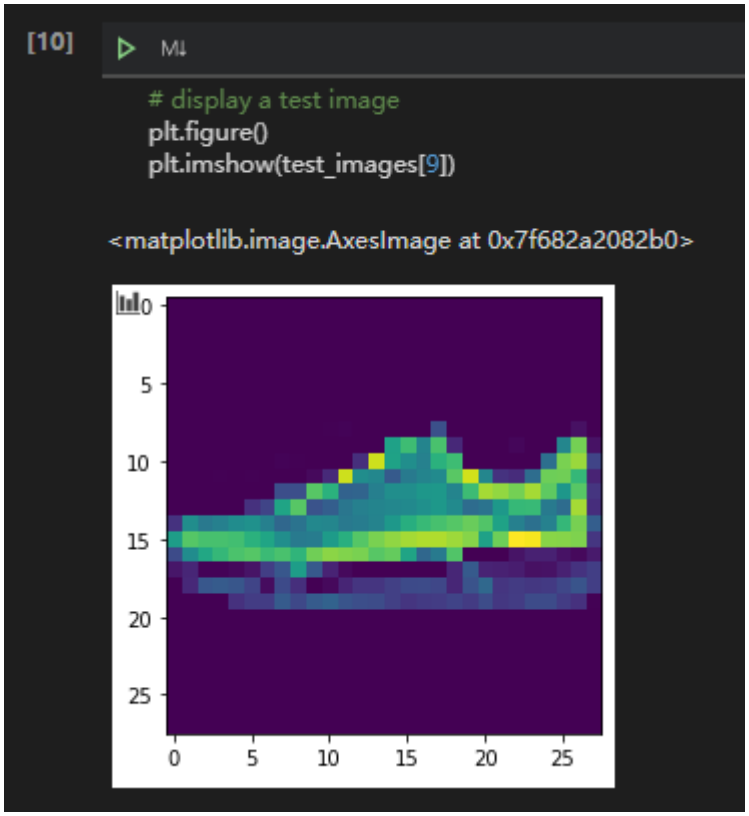
2.5 使用模型

步骤1 用训练好的模型预测测试集中的某个图片属于什么类别，先显示这个图片，命令如下。

```
# display a test image
plt.figure()
plt.imshow(test_images[9])
```


版权所有 © 华为云计算技术有限公司

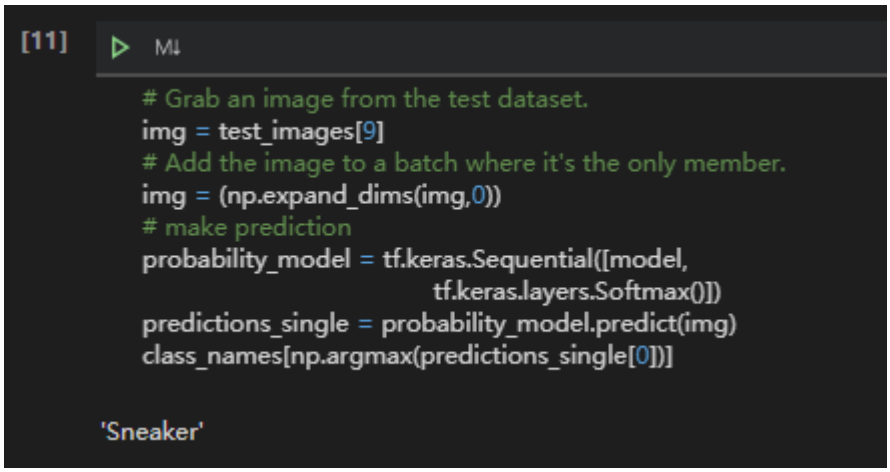
图 2-1 显示用以测试的图片



步骤2 查看预测结果，命令如下。

```
# Grab an image from the test dataset.
img = test_images[9]
# Add the image to a batch where it's the only member.
img = (np.expand_dims(img,0))
# make prediction
probability_model = tf.keras.Sequential([model,
                                         tf.keras.layers.Softmax()])
predictions_single = probability_model.predict(img)
class_names[np.argmax(predictions_single[0])]
```

图 2-2 查看预测结果



版权所有 © 华为云计算技术有限公司

说明

- 当前只对python3做了优化，python2下无法直接运行Jupyter Notebook。
- CodeArts IDE Online暂不支持GPU加速，建议安装tensorflow-cpu减小磁盘占用，并加快安装速度。
- 鲲鹏镜像暂时无法安装TensorFlow，敬请期待后续更新。

----结束

版权所有 © 华为云计算技术有限公司