|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | | |  |
|  | | |
|  | | |
| 方圆ok1 | | | |
|  | **实验手册**  **实验手册** | | 附件1-16K |
|  | |
| **文档版本** | **01** |
| **发布日期** | **2017-08-16** |
| **华为技术有限公司** | |

|  |
| --- |
| 版权所有 © 华为技术有限公司2017。 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  附件3-版权声明页图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默示的声明或保证。  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： | <http://www.huawei.com> |
| 客户服务邮箱： | [support@huawei.com](mailto:support@huawei.com) |
| 客户服务电话： | 4008302118 |

目 录

[1 概述 1](#_Toc490667964)

[2 课程介绍 3](#_Toc490667965)

[2.1 学习对象 3](#_Toc490667966)

[2.2 基本要求 3](#_Toc490667967)

[2.3 学习目标 3](#_Toc490667968)

[2.4 实验内容 4](#_Toc490667969)

[3 实验一：项目管理 5](#_Toc490667970)

[3.1 实验介绍 5](#_Toc490667971)

[3.1.1 关于本实验 5](#_Toc490667972)

[3.1.2 实验目的 5](#_Toc490667973)

[3.1.3 前提条件 5](#_Toc490667974)

[3.1.4 实验任务列表 5](#_Toc490667975)

[3.2 实验任务 6](#_Toc490667976)

[3.2.1 项目及成员的管理 6](#_Toc490667977)

[3.2.2 产品需求分析 7](#_Toc490667978)

[3.2.3 制定迭代计划 8](#_Toc490667979)

[3.2.4 任务分配和进展跟踪 9](#_Toc490667980)

[4 实验二：代码仓库 11](#_Toc490667981)

[4.1 实验介绍 11](#_Toc490667982)

[4.1.1 关于本实验 11](#_Toc490667983)

[4.1.2 实验目的 11](#_Toc490667984)

[4.1.3 前提条件 11](#_Toc490667985)

[4.1.4 实验任务列表 11](#_Toc490667986)

[4.2 实验任务 13](#_Toc490667987)

[4.2.1 环境准备 13](#_Toc490667988)

[4.2.2 创建代码仓库 14](#_Toc490667989)

[4.2.3 Web编写代码 15](#_Toc490667990)

[4.2.4 客户端IDE编写代码 16](#_Toc490667991)

[4.2.5 Git客户端代码分支管理 17](#_Toc490667992)

[4.2.6 Web代码分支管理 18](#_Toc490667993)

[5 实验三：代码检查 21](#_Toc490667994)

[5.1 实验介绍 21](#_Toc490667995)

[5.1.1 关于本实验 21](#_Toc490667996)

[5.1.2 实验目的 21](#_Toc490667997)

[5.1.3 前提条件 21](#_Toc490667998)

[5.1.4 实验任务列表 21](#_Toc490667999)

[5.2 实验任务 22](#_Toc490668000)

[5.2.1 创建代码检查任务 22](#_Toc490668001)

[5.2.2 评估代码质量 23](#_Toc490668002)

[5.2.3 修复代码问题 24](#_Toc490668003)

[6 实验四：编译构建 28](#_Toc490668004)

[6.1 实验介绍 28](#_Toc490668005)

[6.1.1 关于本实验 28](#_Toc490668006)

[6.1.2 实验目的 28](#_Toc490668007)

[6.1.3 前提条件 28](#_Toc490668008)

[6.1.4 实验任务列表 28](#_Toc490668009)

[6.2 实验任务 29](#_Toc490668010)

[6.2.1 创建并配置构建任务 29](#_Toc490668011)

[6.2.2 修改构建任务的配置 29](#_Toc490668012)

[6.2.3 执行编译构建任务并查看构建日志 31](#_Toc490668013)

[6.2.4 查看构建历史并下载构建包 31](#_Toc490668014)

[7 实验五：测试管理 32](#_Toc490668015)

[7.1 实验介绍 32](#_Toc490668016)

[7.1.1 关于本实验 32](#_Toc490668017)

[7.1.2 实验目的 32](#_Toc490668018)

[7.1.3 前提条件 32](#_Toc490668019)

[7.1.4 实验任务列表 32](#_Toc490668020)

[7.2 实验任务 33](#_Toc490668021)

[7.2.1 用例管理 33](#_Toc490668022)

[7.2.2 测试用例设计 34](#_Toc490668023)

[7.2.3 执行测试用例 35](#_Toc490668024)

[7.2.4 查看验收报告 36](#_Toc490668025)

[8 实验七：部署 38](#_Toc490668026)

[8.1 实验介绍 38](#_Toc490668027)

[8.1.1 关于本实验 38](#_Toc490668028)

[8.1.2 实验目的 38](#_Toc490668029)

[8.1.3 前提条件 38](#_Toc490668030)

[8.1.4 实验任务列表 38](#_Toc490668031)

[8.2 实验任务 39](#_Toc490668032)

[8.2.1 主机管理 39](#_Toc490668033)

[8.2.2 服务部署 39](#_Toc490668034)

[9 实验八：发布服务 42](#_Toc490668035)

[9.1 实验介绍 42](#_Toc490668036)

[9.1.1 关于本实验 42](#_Toc490668037)

[9.1.2 实验目的 42](#_Toc490668038)

[9.1.3 前提条件 42](#_Toc490668039)

[9.1.4 实验任务列表 42](#_Toc490668040)

[9.2 实验任务 43](#_Toc490668041)

[9.2.1 环境准备 43](#_Toc490668042)

[9.2.2 包文件上传 43](#_Toc490668043)

[9.2.2.1 页面包文件上传 43](#_Toc490668044)

[9.2.2.2 Maven CLI上传包文件 44](#_Toc490668045)

[9.2.3 依赖包文件下载 45](#_Toc490668046)

[9.2.3.1 页面下载 45](#_Toc490668047)

[9.2.3.2 Mirror仓和私有仓（Release）依赖下载配置 46](#_Toc490668048)

[10 实验九：流水线 48](#_Toc490668049)

[10.1 实验介绍 48](#_Toc490668050)

[10.1.1 关于本实验 48](#_Toc490668051)

[10.1.2 实验目的 48](#_Toc490668052)

[10.1.3 前提条件 48](#_Toc490668053)

[10.1.4 实验任务列表 48](#_Toc490668054)

[10.2 实验任务 49](#_Toc490668055)

[10.2.1 创建流水线 49](#_Toc490668056)

[10.2.2 执行流水线 49](#_Toc490668057)

[10.2.3 修改流水线参数 50](#_Toc490668058)

# 概述

华为软件开发云（DevCloud）是面向开发者提供的云服务，即开即用，随时随地在云端进行项目管理、配置管理（代码托管）、代码检查、流水线、编译、构建、部署、测试、发布等，让开发者快速而又轻松地开启云端开发之旅。

本手册围绕“商品上架”和“商品下架”相关实例介绍如何通过软件开发云进行项目管理、代码托管、代码检查、编译构建、部署、测试、发布等，内容总览如表1-1所示。

总览

| 标题 | 内容 |
| --- | --- |
| 2 课程介绍 | 介绍实验手册的课程，包括：学习对象、基本要求、学习目标和实验内容。 |
| 3 实验一：项目管理 | 介绍项目管理的实验任务，包括：项目及成员的管理、产品需求分析、制定迭代计划和任务分配和进展跟踪。 |
| 4 实验二：代码仓库 | 介绍代码仓库的实验任务，包括：环境准备、创建代码仓库、Web编写代码、客户端IDE编写代码、Git客户端代码分支管理和Web代码分支管理。 |
| 5 实验三：代码检查 | 介绍代码检查的实验任务，包括：创建代码检查任务、评估代码质量和修复代码问题。 |
| 6 实验四：编译构建 | 介绍编译构建的实验任务，包括：创建并配置构建任务、修改构建任务的配置、执行编译构建任务并查看构建日志和查看构建历史。 |
| 7 实验五：测试管理 | 介绍测试用例的实验任务，包括：用例管理、测试用例设计、执行测试用例和查看验收报告。 |
| 8 实验七：部署 | 介绍部署服务的实验任务，包括：主机管理、后台环境部署和前台服务部署。 |
| 9 实验八：发布服务 | 介绍发布服务的实验任务，包括：环境准备、包文件上传和依赖包文件下载。 |
| 10 实验九：流水线 | 介绍流水线的实验任务，包括：创建流水线、执行流水线和修改流水线参数。 |

# 课程介绍

[2.1 学习对象](#_ZH-CN_TOPIC_0069562479)

[2.2 基本要求](#_ZH-CN_TOPIC_0069562490)

[2.3 学习目标](#_ZH-CN_TOPIC_0069562501)

[2.4 实验内容](#_ZH-CN_TOPIC_0069562512)

## 学习对象

希望掌握软件开发云基础知识，成为DevOps工程师的人员。

## 基本要求

* 熟悉PC操作系统，具备基本的计算机知识
* 具备基本的Java开发技能
* 具备Git、Maven工具的使用经验
* 已参加HCDP DevOps系列课程学习

## 学习目标

完成该项目培训后，您将能够：

* 掌握项目及成员的管理。
* 掌握如何使用项目管理服务制定迭代计划。
* 掌握如何使用项目管理服务完成任务分配和进展跟踪。
* 掌握代码仓库的创建。
* 掌握代码仓库的克隆。
* 掌握代码提交。
* 掌握代码推送到远程仓库。
* 掌握代码分支管理。
* 掌握如何创建和配置代码检查任务。
* 掌握执行代码检查任务和质量评估。
* 掌握如何修复代码问题。
* 掌握如何创建和配置构建任务。
* 掌握构件包归档至软件仓库的方法。
* 通过构建日志定位构建问题。
* 掌握构建包的下载。
* 掌握创建和分配测试用例。
* 掌握用例的设计与验收。
* 掌握如何通过测试验收报告评估产品质量。
* 掌握如何管理部署环境的授信主机。
* 掌握如何管理自定义部署模板。
* 掌握如何创建和执行部署任务。
* 掌握发布仓库的基本使用方法。
* 掌握如何新建和配置流水线任务。
* 掌握如何设置流水线执行参数。
* 掌握如何执行流水线。

## 实验内容

本项目的实验内容，详见每个实验的任务列表。

# 实验一：项目管理

[3.1 实验介绍](#_ZH-CN_TOPIC_0069562534)

[3.2 实验任务](#_ZH-CN_TOPIC_0069562331)

## 实验介绍

### 关于本实验

本实验介绍了项目的基本操作。

### 实验目的

* 掌握项目及成员的管理。
* 掌握如何使用项目管理服务制定迭代计划。
* 掌握如何使用项目管理服务完成任务分配和进展跟踪。

### 前提条件

拥有华为云账号，且已经开通项目管理服务。

### 实验任务列表

| **任务** | **子任务** | **使用场景** |
| --- | --- | --- |
| 3.2.1 项目及成员的管理 | 创建项目 | 在华为软件开发云上创建一个项目，项目团队围绕该项目开展需求分析、设计、编码、构建、测试、部署和发布等项目活动。 |
| 添加项目成员 | 只有成为项目的成员，才能拥有该项目的访问权限。项目创建者或者项目经理，可以将本租户或者其他租户的用户加入该项目中。 |
| 3.2.2 产品需求分析 | Backlog管理 | 产品经理根据市场、产品规划、客户需求和客户问题反馈等需求来源，创建Story/任务/Bug，放入产品Backlog中。 |
| 3.2.3 制定迭代计划 | 创建迭代（Sprint） | 产品经理根据产品业务目标、可用资源等，制定Sprint。 |
| 确定迭代（Sprint）内容 | 产品经理确定本次Sprint的范围，确定该Sprint需要交付的Story/任务/Bug，并确定工作项的优先级。 |
| 3.2.4 任务分配和进展跟踪 | Story澄清和讨论 | Scrum Master组织敏捷团队讨论本次Sprint的Story/任务/Bug。 |
| Story交付和验收 | 开发人员交付本次Sprint的Story/任务/Bug，测试人员验收Sory。 |

## 实验任务

### 项目及成员的管理

创建项目，项目名称为“HCDP-DevOps”，项目流程采用Scrum流程。



添加项目成员。

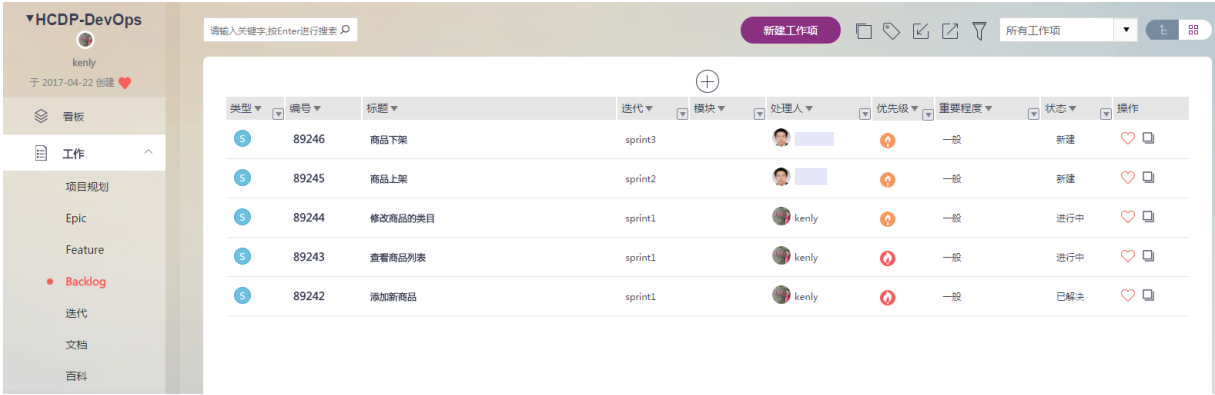
从软件开发云首页进入该项目，分别进入“设置 > 成员”模块，根据项目需要添加项目成员。



----结束

### 产品需求分析

从软件开发云首页进入该项目，分别进入“任务 > Backlog”模块。



创建“添加新商品”的Story。

创建“查看商品列表”的Story。

创建“修改商品的类目”的Story。

创建“商品上架”的Story。

创建“商品下架”的Story。

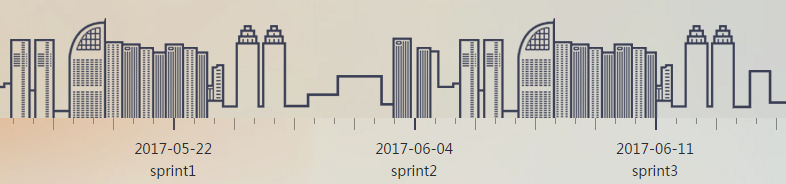


----结束

### 制定迭代计划

在项目左侧导航菜单栏中选择“看板”。

单击“新建迭代”，创建3个迭代：“sprint1”、“sprint2”和“sprint3”。



确定迭代“sprint1”的内容。将“添加新商品”、“查看商品列表”和 “修改商品的类目”3个Story的迭代修改为“sprint1”即可。





该迭代有“添加新商品”、 “查看商品列表”和 “修改商品的类目”3个Story。

----结束

### 任务分配和进展跟踪

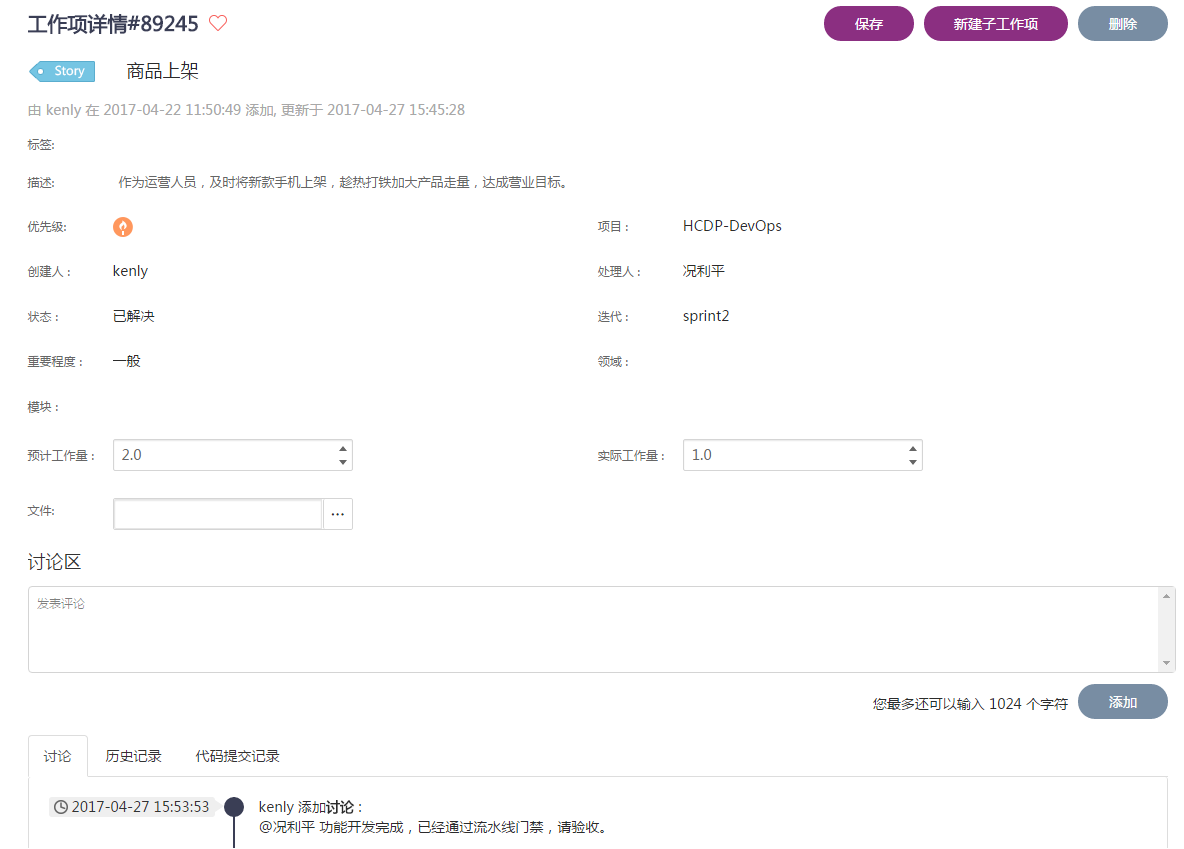
Story澄清&讨论。

产品经理在迭代会议上向交付团队、测试团队等讲解用户故事，对用户使用场景展开讨论。Story详情页面如下：



Story交付&验收。

开发人员编码实现“商品上架”的功能，并通过代码检视、自动化单元测试和基本的功能测试，提交产品经理验收。Story详情页面如下：



----结束

# 实验二：代码仓库

[4.1 实验介绍](#_ZH-CN_TOPIC_0069562458)

[4.2 实验任务](#_ZH-CN_TOPIC_0069562463)

## 实验介绍

### 关于本实验

本实验介绍代码仓库的操作，如何使用配置管理服务作代码版本管理。

### 实验目的

* 掌握代码仓库的创建。
* 掌握代码仓库的克隆。
* 掌握代码提交。
* 掌握代码推送到远程仓库。
* 掌握代码分支管理。

### 前提条件

拥有华为云账号，已开通配置管理服务，并创建有项目。

### 实验任务列表

| **任务** | **子任务** | **使用场景** |
| --- | --- | --- |
| 4.2.1 环境准备 | 生成SSH秘钥 | 使用Git客户端生成SSH密钥。 |
| 添加SSH秘钥至配置管理服务 | 将Git客户端生成SSH密钥添加到配置管理服务，使用SSH密钥在开发者环境和服务端间建立安全的数据传输通道。 |
| 4.2.2 创建代码仓库 | 后台服务代码库 | 创建商品管理的后台微服务的代码仓库。本示例采用springboot框架、使用maven构建。 |
| 前台服务代码库 | 在配置管理服务中创建商品管理的AngularJS2/maven代码仓库。 |
| 4.2.3 Web编写代码 | 在线查看代码文件 | 使用Web浏览器在配置管理服务中查看代码文件。 |
| 在线修改并提交代码文件 | 使用Web浏览器在配置管理服务中根据实际需要修改并提交代码文件。 |
| 4.2.4 客户端IDE编写代码 | 使用Git客户端克隆代码仓库 | 使用Git客户端将配置管理服务中远端代码仓库克隆到本地。 |
| master分支上编写商品上架的代码并commit | 使用Git客户端签出master分支，并在该分支上编写商品列表和添加、删除的代码，并使用Git客户端commit代码。 |
| 将master分支push到远端仓库 | 使用Git客户端将master分支push到远端代码仓库。 |
| 4.2.5 Git客户端代码分支管理 | 创建并切换develop的分支 | 使用Git客户端创建并切换develop分支。 |
| develop分支上编写商品下架的代码并commit | 使用Git客户端签出develop分支，并在该分支上编写删除商品的代码，并使用Git客户端commit代码。 |
| 将develop分支push到远端仓库 | 使用Git客户端将develop分支push到远端仓库。 |
| develop分支合入master分支 | 使用Git客户端将develop分支合入master分支。 |
| 4.2.6 Web代码分支管理 | 新建合并分支请求 | 使用Web浏览器在配置管理服务中新建合并分支请求，将develop分支合入master分支。 |
| 审批分支合并请求 | 使用Web浏览器在配置管理服务中审批分支合并请求。 |
| 查看仓库分支网络 | 使用Web浏览器在配置管理服务中查看仓库分支网络。 |

## 实验任务

### 环境准备

下载并安装Git客户端：

下载地址：<https://git-scm.com/downloads>

Git全局配置：

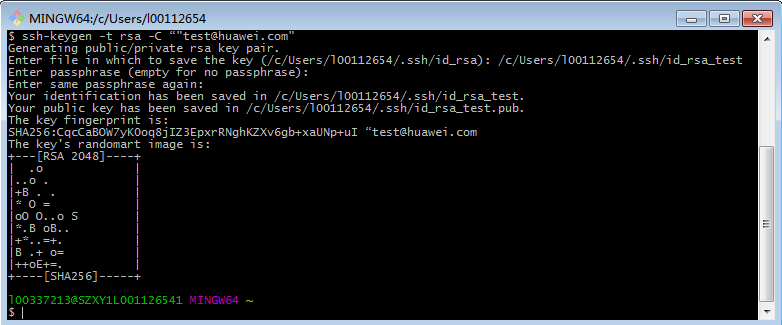
运行**Git Bash**，在终端配置如下用户信息：

git config --global user.name "您的名字"   
git config --global user.email "您的email"

生成SSH密钥：

生成SSH公钥命令：**ssh-keygen -t rsa -C "您的email"**

私钥保存路径为/c/Users/*l00112654*/.ssh/id\_rsa\_test，如下图所示：

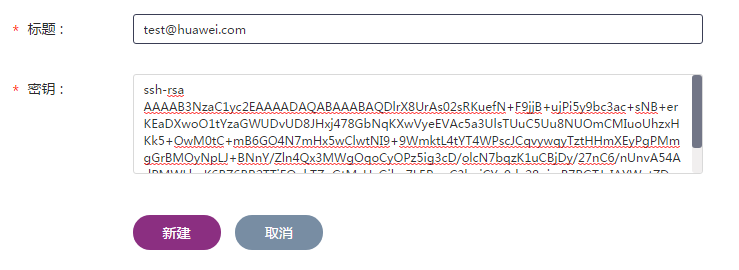


公钥文件路径：C:\Users\*l00112654*\.ssh\id\_rsa\_test.pub

私钥文件路径：C:\Users\*l00112654*\.ssh\id\_rsa\_test

添加SSH密钥，步骤[3](#li7175221)中生成的公钥拷贝至如下图所示的“添加SSH密钥”页面密钥栏中。

在配置管理首页单击“密钥设置 > 添加SSH密钥”进入：

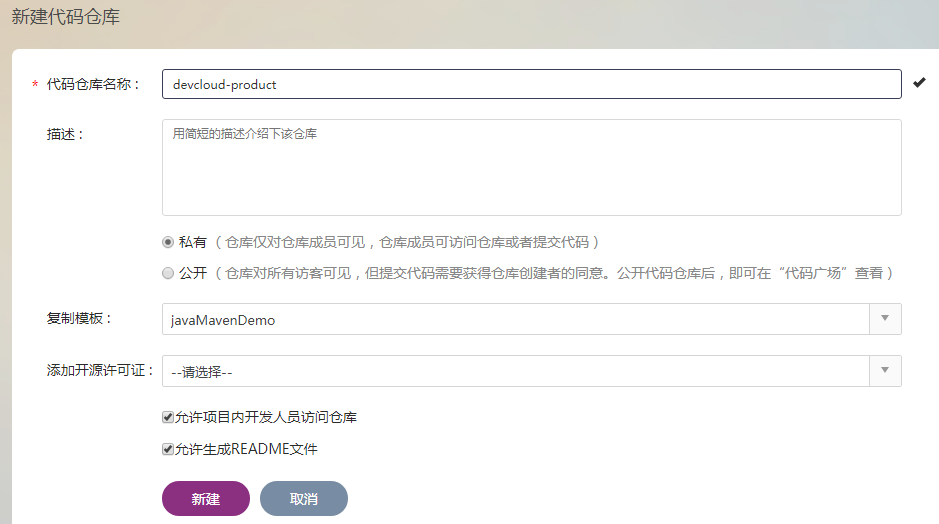


----结束

### 创建代码仓库

创建后台服务代码仓库。

1. 在项目左侧导航菜单栏中选择“代码”。
2. 单击“新建仓库”按钮。
3. 输入代码仓库名为“devcloud-product”，如下图所示：



创建前台服务代码仓库。

1. 进入项目页在左侧菜单栏中选择“代码”。
2. 单击“新建仓库”按钮。
3. 输入代码仓库名为“devcloud-product01”，如下图所示：

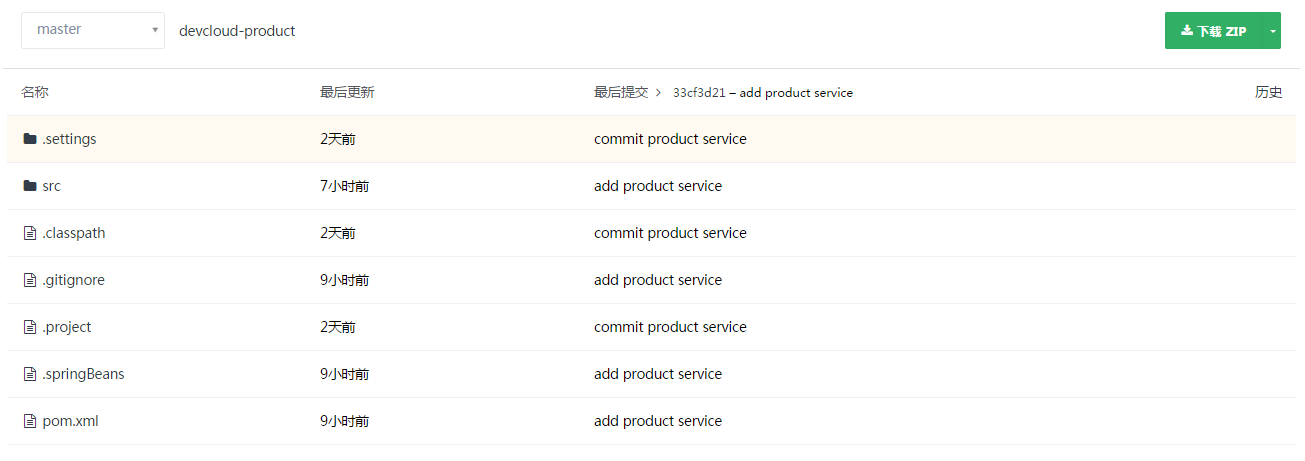


----结束

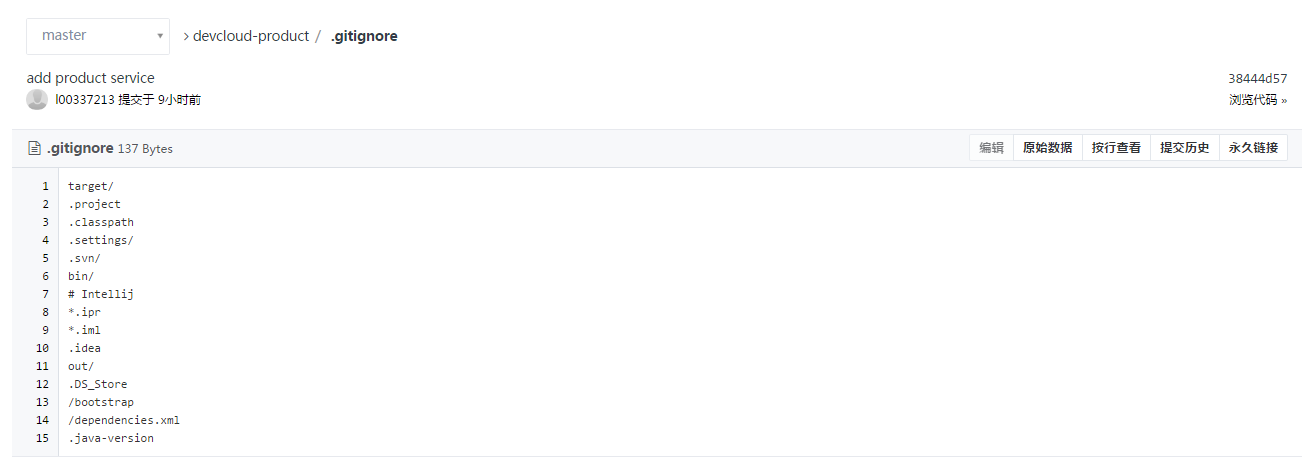
### Web编写代码

Web查看代码文件。

1. 在代码仓库卡片页面单击代码仓库“devcloud-product”。
2. 在代码仓库详情页面单击左侧的菜单项“文件”，打开如下的页面：

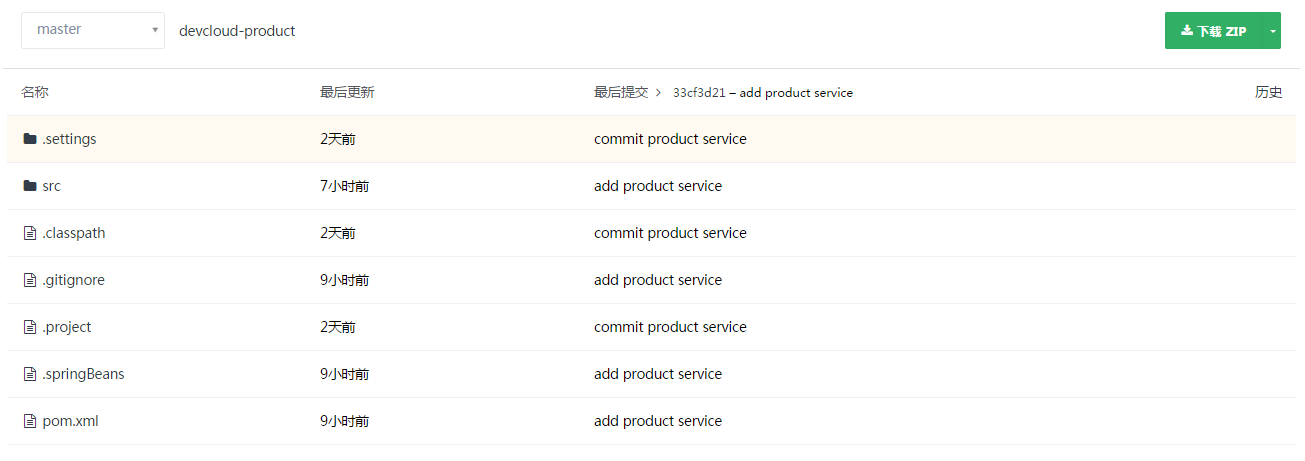


1. 打开文件“.gitignore”，如下图所示：



Web修改并提交代码文件。

1. 在代码仓库卡片页面单击代码仓库“devcloud-product”。
2. 在代码仓库详情页面单击左侧菜单项“文件”，打开如下的页面：



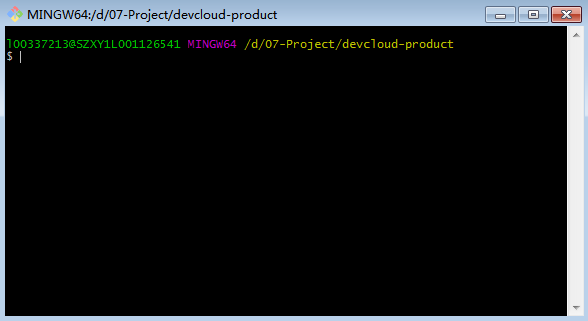
1. 打开目录“/src/test/java/com/huawei/devcloud/pbi”下的文件“ProductStarterApplicationTests.java”。
2. 单击“编辑”按钮，修改代码，然后提交。

----结束

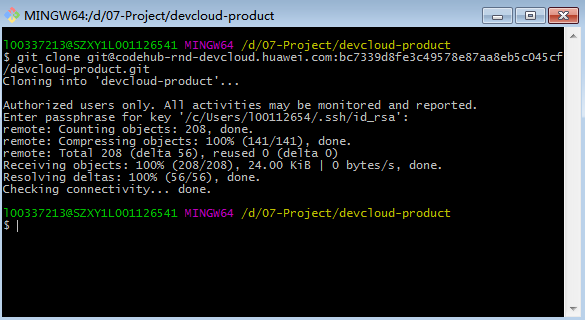
### 客户端IDE编写代码

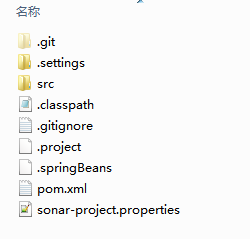
使用Git客户端克隆代码仓库。

1. 切换到本地目录，例如：“D:\07-Project\devcloud-product”。
2. 通过右键菜单运行本地“Git Bash”。



1. 执行命令：**git clone git@codehub-rnd-devcloud.huawei.com:bc7339d8fe3c49578e87aa8eb5c045cf/devcloud-product.git**。详见如下图所示：





master分支上编写商品上架的代码并Commit。

1. 使用IDE或者文本编辑器，打开源码文件：“D:\07-Project\devcloud-product\src\main\java\com\huawei\devcloud\pbi\service\ProductController.java”
2. 增加并保存如下代码：

@RequestMapping(value = "/{id}", method = RequestMethod.PUT)   
public HttpResult goOnSale(@PathVariable String id, @RequestParam Boolean action) {   
result.init();   
if (action) {   
result.setResult(projectService.goOnSale(id));   
} else {   
result.setResult(projectService.soldout(id));   
}   
return result;   
}

1. 添加修改至本地缓存区：**git add .**
2. 提交修改和描述至本地：**git commit –m "product go onsale"**

将master分支push到远端仓库

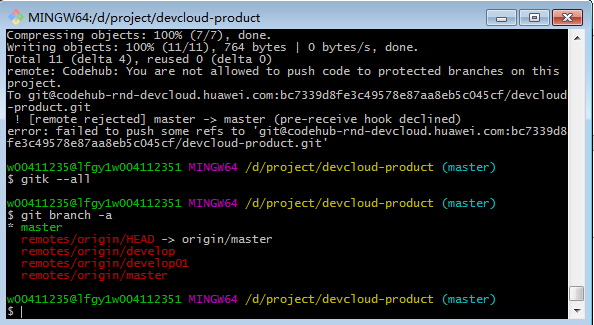
1. 同步远端服务器代码到本地：**git pull**
2. 将本地master分支push到远端仓库：**git push origin master**

----结束

### Git客户端代码分支管理

创建并切换“develop”的分支。

1. 打开本地“Git Bash”。
2. 进入指定代码仓库路径。
3. 查看所有分支：**git branch -a**，详见如下图所示：



1. 创建并切换至develop分支：**git checkout –b develop**

“develop”分支上编写删除商品的代码并commit。

1. 使用IDE或者文本编辑器，打开源码文件：“D:\07-Project\devcloud-product\src\main\java\com\huawei\devcloud\pbi\service\ProductController.java”
2. 修改并保存如下代码：

@RequestMapping(value = "/{id}", method = RequestMethod.DELETE)   
public HttpResult removeProduct(@PathVariable String id) {   
result.init();   
result.setResult(projectService.removeProduct(id));   
return result;   
}

1. 添加修改至本地缓存区：**git add .**
2. 提交修改和描述至本地：**git commit –m "delete a product"**

将“develop”分支push到远端仓库

1. 同步远端最新代码：**git pull**
2. 将“develop”分支推送到远端仓库：**git push origin develop**

“develop”分支合入master分支

1. 查看当前所处分支：**git branch**
2. 切换至master分支：**git checkout master**
3. 将develop分支合入master分支：**git merge develop**

----结束

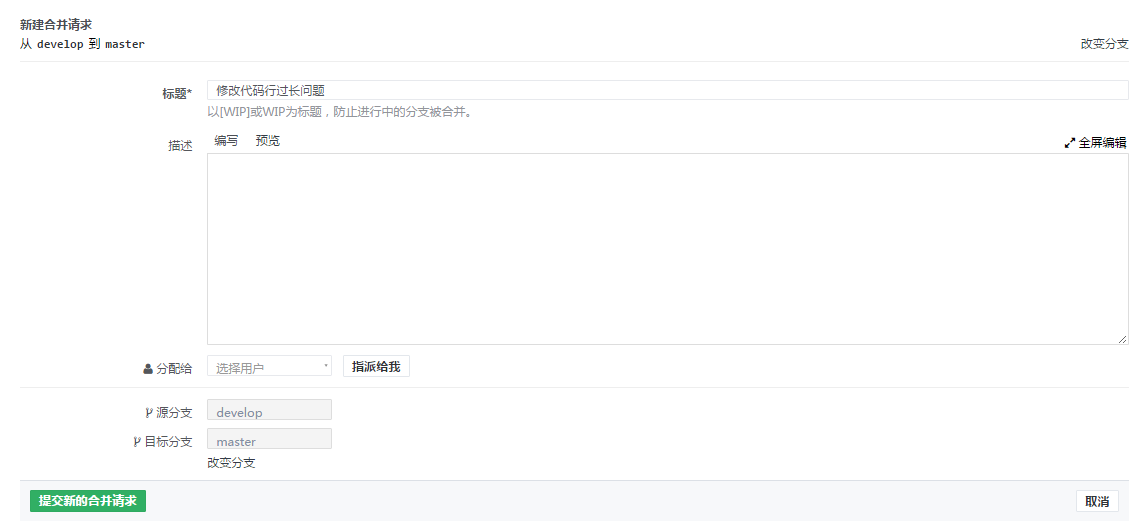
### Web代码分支管理

新建合并分支请求。

1. 进入代码仓库“devcloud-product”详情页，单击左侧菜单项“合并请求”项。
2. 单击页面右上方的“新建合并请求”按钮。
3. 在页面上显示的源分支处选择develop，然后单击“比较分支”按钮，详见如下图所示：

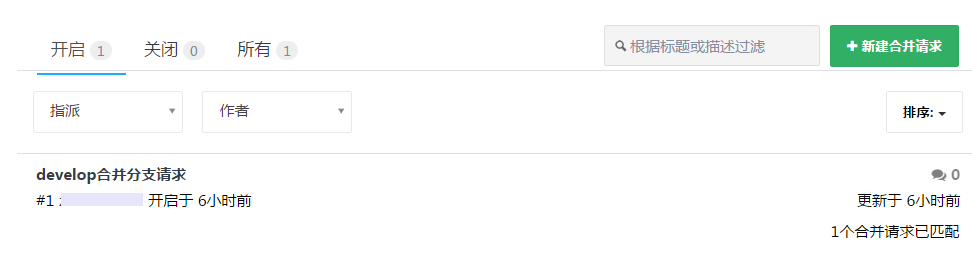


1. 进入如下图所示的合并请求新建页面，填写相应的内容。



审批分支合并请求。

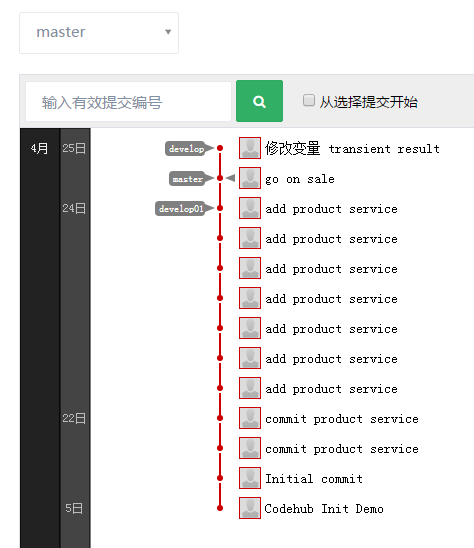
1. 如**步骤**[1](#li38971143)中所示，将合并请求分配给自己。
2. 单击进入“合并请求”列表页。
3. 查看**步骤**[1](#li38971143)中新建的合并请求，如下图所示处于开启状态的合并请求。



1. 单击“接受合并请求”按钮。

查看仓库分支网络。

1. 进入代码仓库“devcloud-product”详情页，单击左侧菜单项“仓库网络”。
2. 查看到如下图所示的仓库分支网络图。



----结束

# 实验三：代码检查

[5.1 实验介绍](#_ZH-CN_TOPIC_0069562472)

[5.2 实验任务](#_ZH-CN_TOPIC_0069562477)

## 实验介绍

### 关于本实验

本实验介绍如何使用代码检查服务发现和辅助修复代码质量问题。

### 实验目的

* 掌握如何创建和配置代码检查任务。
* 掌握执行代码检查任务和质量评估。
* 掌握如何修复代码问题。

### 前提条件

拥有华为云账号，已开通代码检查服务，并创建有项目和代码仓库。

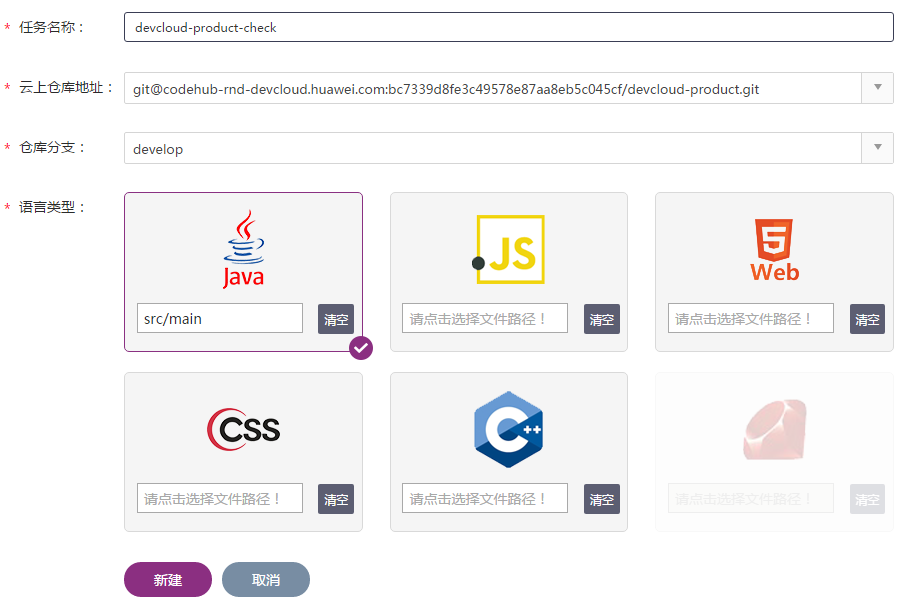
### 实验任务列表

| **任务** | **子任务** | **使用场景** |
| --- | --- | --- |
| 5.2.1 创建代码检查任务 | 选择商品管理的代码仓库 | 在代码检查服务中创建代码检查任务时，选择商品管理的代码仓库。 |
| 选择develop分支 | 在代码检查服务中创建代码检查任务时，仓库分支选择develop分支。 |
| 选择java语言 | 在代码检查服务中创建代码检查任务时，语言类型选择java语言和待检查的代码目录。 |
| 5.2.2 评估代码质量 | 执行代码检查任务 | 在代码检查服务中执行代码检查任务。 |
| 查看代码质量看板 | 在代码检查服务中查看代码质量看板，查看代码质量指数、代码复杂度、代码重复率等指标。 |
| 查看一个代码问题及其修改建议 | 在代码检查服务中查看检查出的一个代码问题及其修改建议。 |
| 查看急需处理的代码问题 | 在代码检查服务中查看检查出的急需处理的问题。 |
| 5.2.3 修复代码问题 | 在线打开代码文件 | 在代码检查服务中打开存在问题的代码文件。 |
| 在线修改代码问题并commit | 在代码检查服务中根据问题修改建议修改代码问题并提交。 |
| 再次执行代码检查任务 | 在代码检查服务中根据问题修改建议修改并提交代码后，再次执行代码检查任务。 |
| 查看代码质量看板，对比风险指数。 | 在代码检查服务中查看代码质量看板，并对比风险指数，检查上次发现的代码问题是否已修复。 |

## 实验任务

### 创建代码检查任务

创建名为“devcloud-product-check”的代码检查任务，选择名为“devcloud-product”的代码仓库。



选择“develop”分支。

在“新建任务”页面，选择名为“develop”的仓库分支。

选择java语言和文件路径。

“语言类型”选择“java语言”，检查目录选择java代码所在的目录，例如：“src/main”，单击“新建”。

----结束

### 评估代码质量

执行“devcloud-product-check”代码检查任务。

单击任务卡片上的“开始检查”。



查看代码质量看板。

执行检查完成后，单击“devcloud-product-check”名称链接。

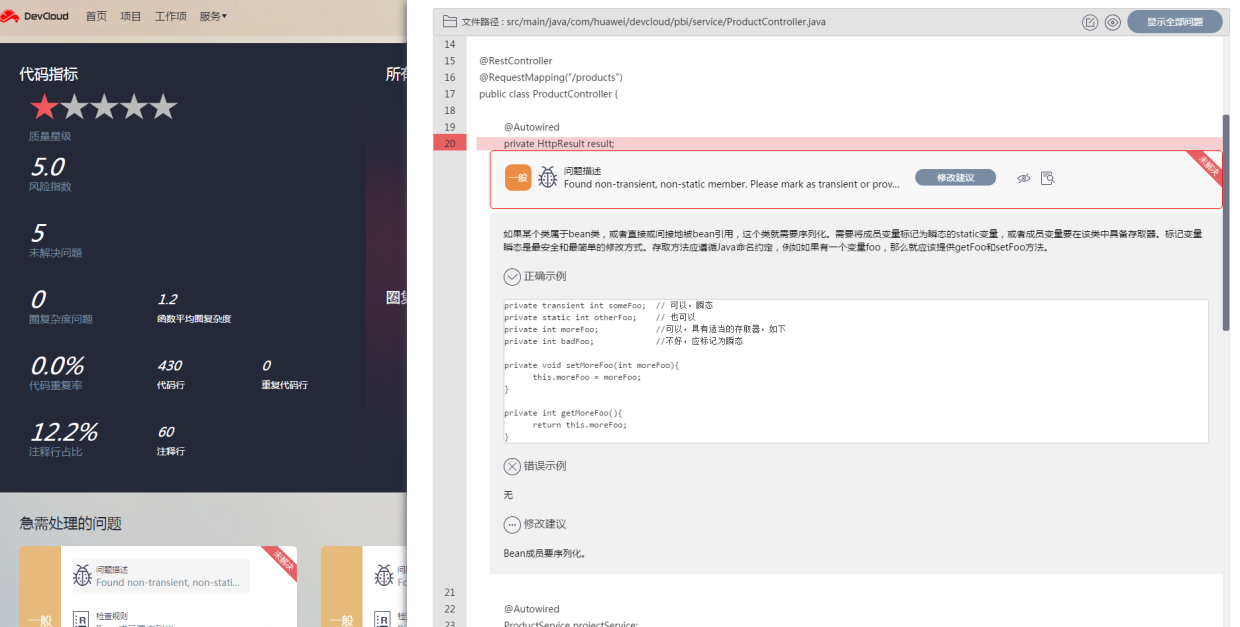
进入代码质量看板，查看最近检查任务的质量星级、问题数、复杂度问题、重复代码行以及急需处理的问题等。





查看一个代码问题及其修改建议。

在“急需处理的问题”中，单击问题描述，进入问题详情页面，显示文件路径、问题修改建议等，可以根据需要编辑源码或查看源码。

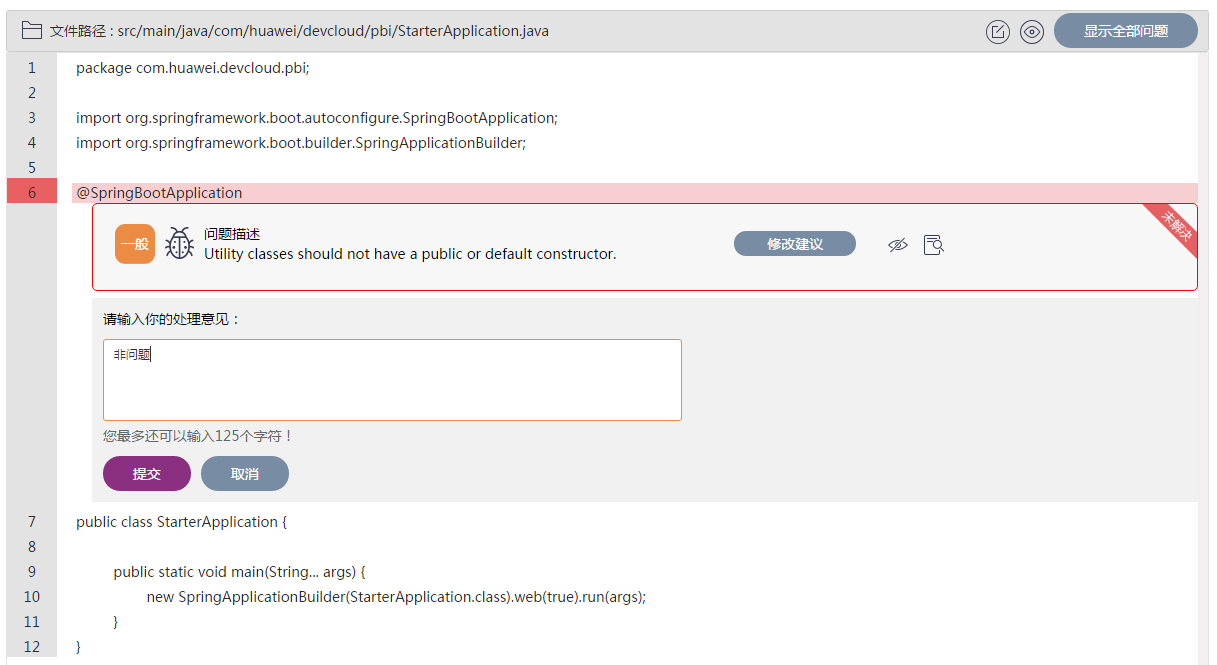


----结束

### 修复代码问题

非问题，可以选择忽略。



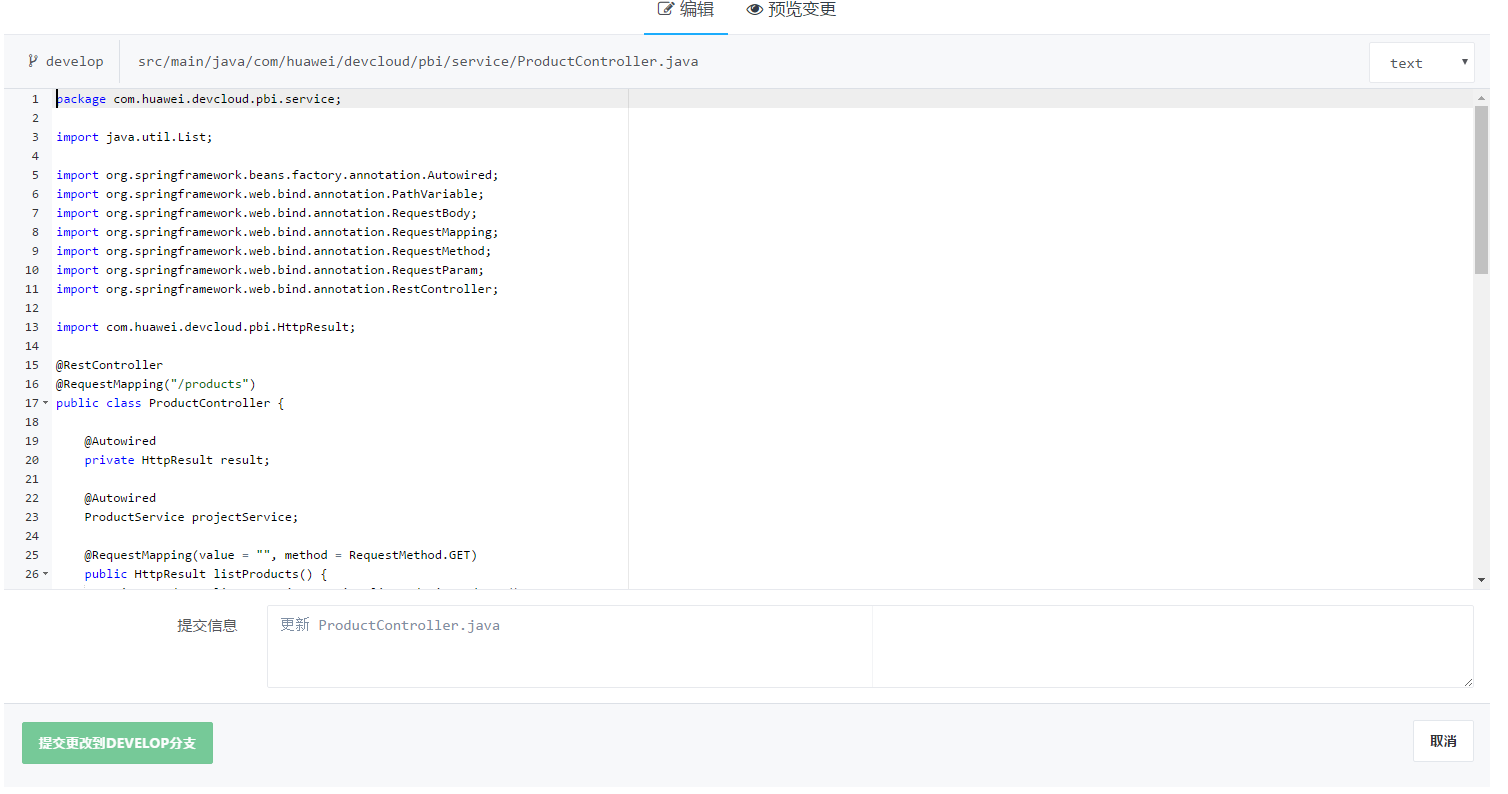


使用Web浏览器打开代码文件。

进入代码问题详情页面，单击编辑源码按钮。

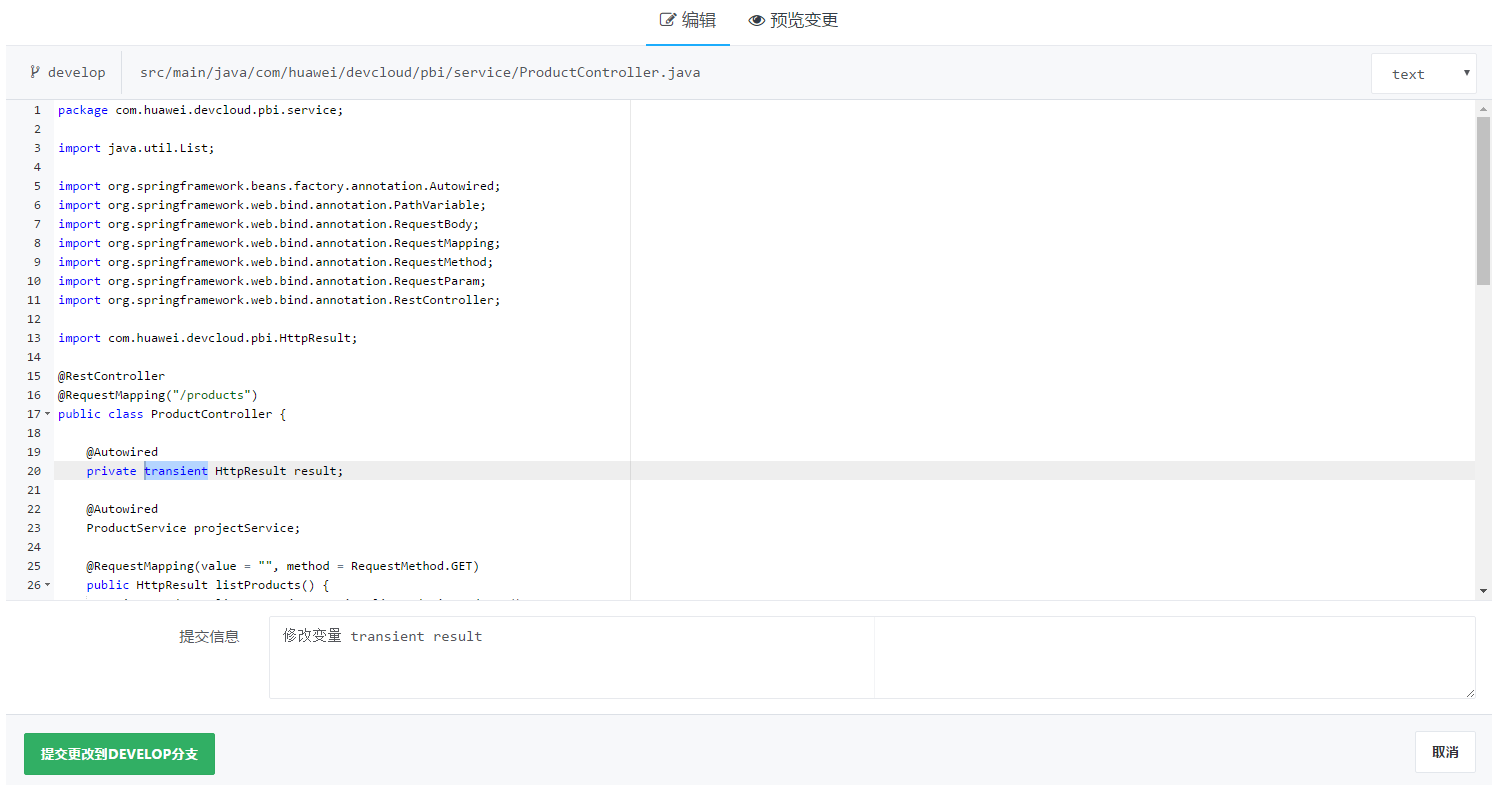


进入代码文件编辑页面。



Web浏览器线上修改代码问题并Commit。

在代码文件编辑页面，修改代码问题后，输入“提交信息”，单击“提交更改到DEVELOP分支”。



再次执行代码检查任务。

单击“devcloud-product-check”代码检查任务卡片上的“开始检查”。

查看代码质量看板，对比风险指数。

执行检查完成后，单击“devcloud-product-check”名称链接。该问题已经解决，风险指数已降低。



----结束

# 实验四：编译构建

[6.1 实验介绍](#_ZH-CN_TOPIC_0069562483)

[6.2 实验任务](#_ZH-CN_TOPIC_0069562488)

## 实验介绍

### 关于本实验

本实验介绍编译构建服务的基本操作，如何使用编译构建服务编译和构建软件产品。

### 实验目的

* 掌握如何创建和配置构建任务。
* 掌握构件包归档至软件仓库的方法。
* 通过构建日志定位构建问题。

### 前提条件

拥有华为云账号，已开通编译构建服务，并创建有项目和代码仓库。

### 实验任务列表

| **任务** | **子任务** | **使用场景** |
| --- | --- | --- |
| 6.2.1 创建并配置构建任务 | 创建并配置构建任务 | 在编译构建服务中创建和配置构建任务。 |
| 6.2.2 修改构建任务的配置 | 修改构建任务的配置 | 编译构建完成后，添加一个步骤，将构建结果归档到软件仓库中。 |
| 6.2.3 执行编译构建任务并查看构建日志 | 执行编译构建任务 | 执行编译构建任务并查看构建日志。 |
| 6.2.4 查看构建历史并下载构建包 | 查看构建历史 | 在构建历史列表查看所有的构建记录，并下载构建包。 |

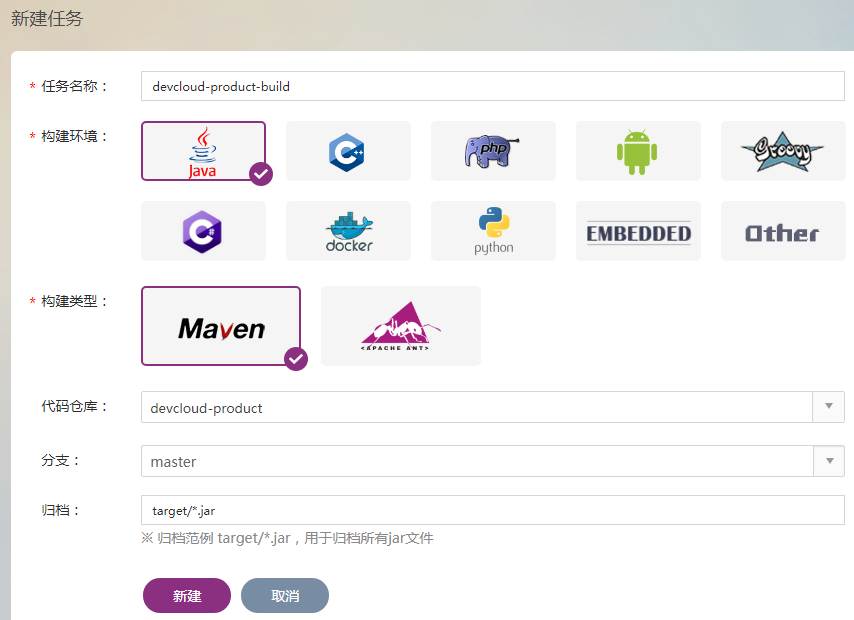
## 实验任务

### 创建并配置构建任务

输入构建任务名为“devcloud-product-build”，选择构建环境为“Java”，构建类型为“Maven”。

选择“devcloud-product”的代码仓库，选择master分支，然后创建名为“devcloud-product-build”的任务。

输入待归档软件包文件路径为“target/\*.jar”。



----结束

### 修改构建任务的配置

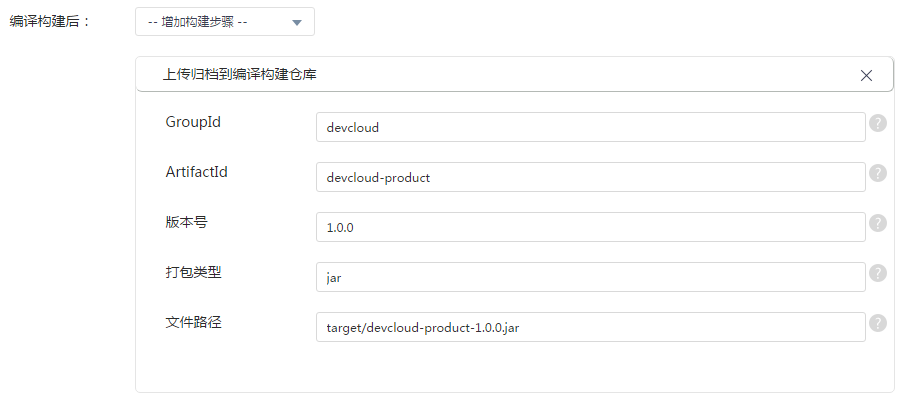
单击任务所在卡片的按钮修改任务的配置，如下图所示。



在配置任务页面修改任务的配置，例如：添加执行参数。



编译构建后，添加一个步骤，将构建结果归档到软件仓库中，如下图所示：



----结束

### 执行编译构建任务并查看构建日志

单击任务所在卡片的名称链接，进入编译构建任务详情页面，如下图所示。

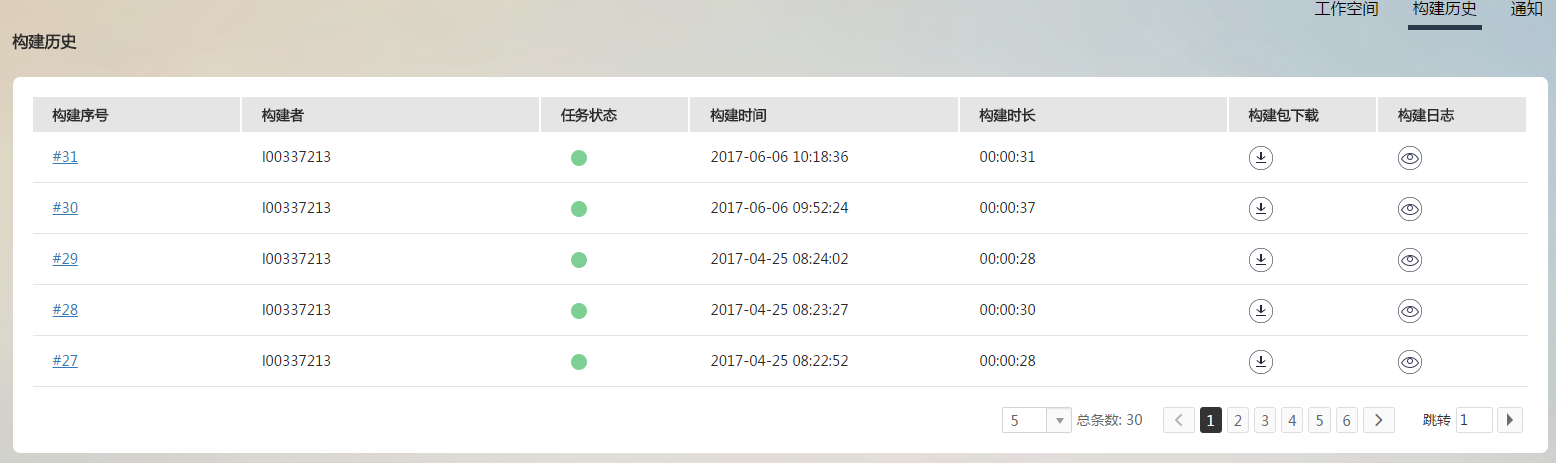


单击“开始构建”按钮执行构建任务并查看构建日志。

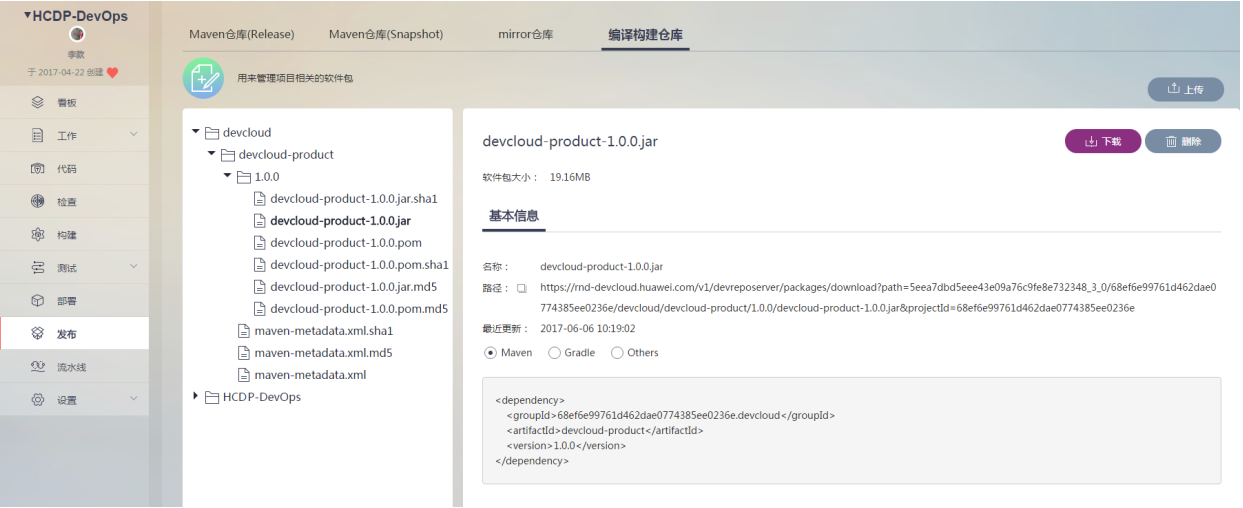
----结束

### 查看构建历史并下载构建包

进入编译构建任务详情页面，选择“构建历史”页签，在构建历史列表查看所有的构建记录，并下载构建包。



左侧导航选择“发布”菜单，单击“编译构建仓库”页签，在发布服务的编译构建仓库查看并下载构建包。



----结束

# 实验五：测试管理

[7.1 实验介绍](#_ZH-CN_TOPIC_0069562495)

[7.2 实验任务](#_ZH-CN_TOPIC_0069562500)

## 实验介绍

### 关于本实验

本实验介绍了测试管理服务的基本操作，如何通过测试管理服务高效管理软件交付过程的测试活动。

### 实验目的

* 掌握创建和分配测试用例。
* 掌握用例的设计与验收。
* 掌握如何通过测试验收报告评估产品质量。

### 前提条件

拥有华为云账号，且已经开通测试管理服务，并创建有项目。

### 实验任务列表

| **任务** | **子任务** | **使用场景** |
| --- | --- | --- |
| 7.2.1 用例管理 | 用例管理 | 创建用例，为用例分配责任人、设置等级、迭代计划、关联需求等。 |
| 7.2.2 测试用例设计 | 测试用例设计 | 为需求设计测试用例。 |
| 7.2.3 执行测试用例 | 执行测试用例 | 执行测试用例并设置用例执行结果。 |
| 7.2.4 查看验收报告 | 测试验收报告 | 查看测试验收概况、设计完成率、用例通过率等。 |

## 实验任务

### 用例管理

从项目下进入到测试管理页面，进入“用例管理”节点，可以查看当前项目下所有用例。

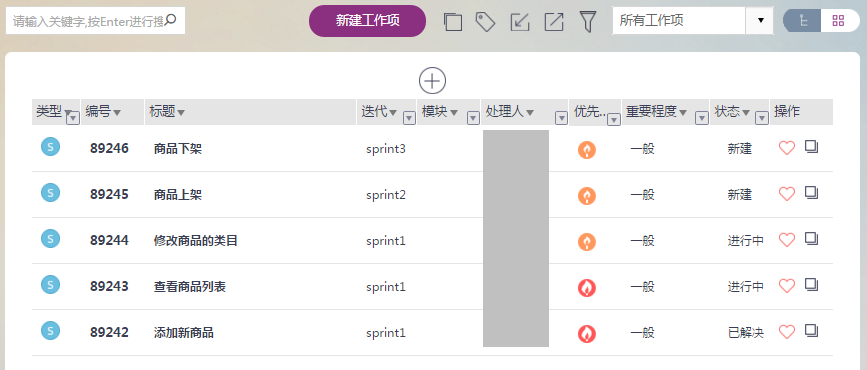
在该页面新建名为“修改商品的目录”的用例，并为用例分配责任人、设置等级、迭代计划、关联需求等。



----结束

### 测试用例设计

切换至“设计与验收”节点，进入“测试设计”，如果当前迭代下没有需求，切换至“工作”节点创建如下工作项。



工作项创建后，切回至“测试设计”节点为各工作项添加或复制测试用例。

----结束

### 执行测试用例

进入下一步“用例执行”，单击用例名称，进入到“用例详情”页面，可以修改用例基本信息。

切换至“设置结果”页签，可以设置用例执行结果、创建缺陷、查看历史记录等。



----结束

### 查看验收报告

进入下一步“设计与验收->验收报告”页面。

查看测试验收概况、未测试需求、已测试需求、未通过用例、已通过用例、缺陷概况及质量评价。



----结束

# 实验七：部署

[8.1 实验介绍](#_ZH-CN_TOPIC_0069562517)

[8.2 实验任务](#_ZH-CN_TOPIC_0069562522)

## 实验介绍

### 关于本实验

本实验介绍了部署服务的基本操作，如何通过部署服务实现软件产品的自动化部署。

### 实验目的

* 掌握如何管理部署主机。
* 掌握如何创建、执行、查看部署任务。

### 前提条件

拥有华为云账号，且已经申请部署服务的公测权限。

### 实验任务列表

| **任务** | **子任务** | **使用场景** |
| --- | --- | --- |
| 8.2.1 主机管理 | 添加授信主机 | 添加目标机器 |
| 8.2.2 服务部署 | alpha部署 | alpha环境下部署 |
| beta部署 | beta环境下部署 |
| gamma部署 | gamma环境下部署 |
| product部署 | product环境下部署 |

## 实验任务

### 主机管理

从项目下进入部署管理页面，单击右上角“主机管理”按钮进入部署主机管理页面。

单击“添加主机”按钮添加主机，如下图：



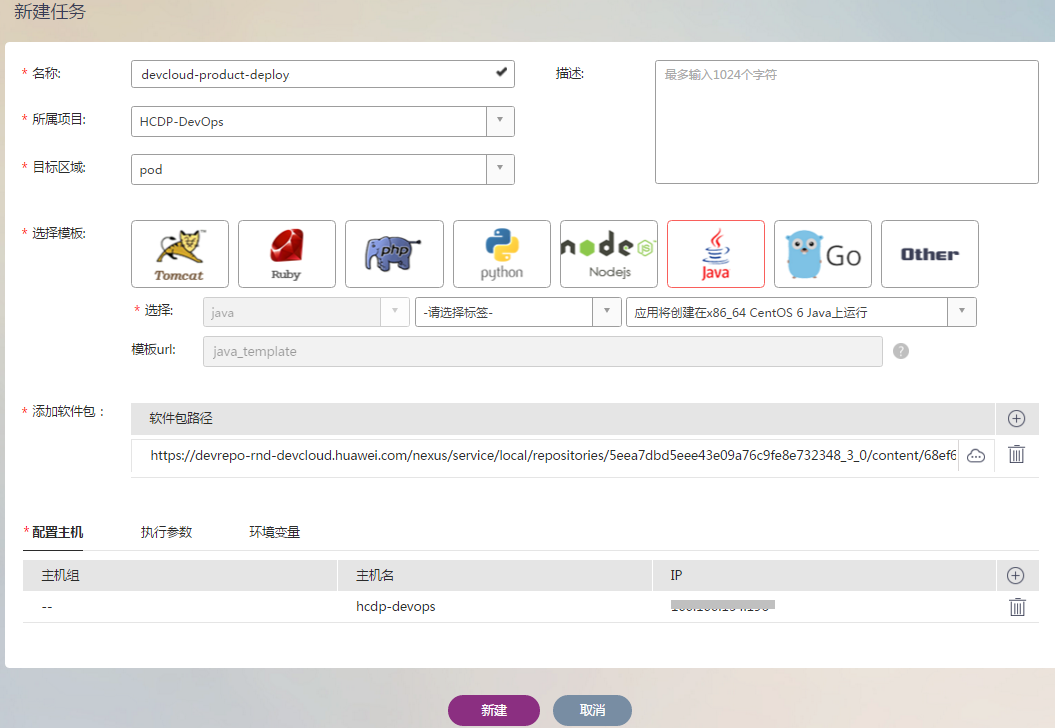
切换到主机组管理页签，可以新建主机组，用来管理主机。

----结束

### 服务部署

以beta环境部署为例讲解。

从项目下进入部署管理页面，单击右上角“新建任务”创建名为“devcloud-product-deploy”的部署任务。



为项目选择模板，如：Java预定义模板。

为项目添加软件包，单击“添加”，选择从发布仓库中添加应用程序，选择对应的编译构建包。



为项目配置主机，单击“添加”，选择目标主机。

（可选）为项目设置执行参数、环境变量、部署步骤等。

任务创建完成后单击任务名进入任务详情页面，可以根据需要修改任务信息。单击“开始部署”部署任务。



部署完成后可以查看部署进程、执行参数、部署信息、主机关系、部署日志、部署历史等，还可以通过应用验证路径访问该应用，确认该应用是否已经成功部署到目标机器。



----结束

# 实验八：发布服务

[9.1 实验介绍](#_ZH-CN_TOPIC_0069562528)

[9.2 实验任务](#_ZH-CN_TOPIC_0069562533)

## 实验介绍

### 关于本实验

本实验介绍发布仓库，让用户了解发布仓库上传、下载等操作以及如何使用发布仓库配合本地Maven工具进行构建。

### 实验目的

* 掌握发布仓库的基本使用方法。

### 前提条件

拥有华为云账号，已开通发布管理服务，并初始化创建了仓库。

### 实验任务列表

| **任务** | **子任务** | **使用场景** |
| --- | --- | --- |
| 9.2.2 包文件上传 | 页面包文件上传 | 用户可以通过页面上传功能将包文件上传到自己的私有仓中。 |
| Maven CLI上传包文件 | 用户可以通过本地Maven工具的命令行方式将包文件上传到自己的私有仓中。 |
| 9.2.3 依赖包文件下载 | 页面下载 | 用户可以通过界面下载功能下载自己私有仓中的包文件 |
| Mirror仓和私有仓（Release）依赖下载配置 | 用户使用本地Maven构建或云上构建时自动通过Mirror仓或私有仓下载依赖包文件的配置操作。 |

## 实验任务

### 环境准备

进入发布服务页面，系统自动初始化发布服务仓库。

安装Apache Maven 3.5.0工具。

下载地址：<https://maven.apache.org/download.cgi>

----结束

### 包文件上传

#### 页面包文件上传

通过服务下页面私有仓上传按钮将私有包文件上传到私有release仓。



“Groupid”、“artifactid”、“版本号”、“打包类型”是Maven类型仓库中包文件的坐标，用来唯一标识仓库中的包文件，为必填项。

Groupid：通常是包文件所属组织或企业的标识，可自定义，常用域名倒排方式表示，例如：“com.huawei.devcloud”

Artifactid：通常是指包文件所属模块的标识， 可自定义，如“model”

打包类型：包文件的后缀名，如“artifact.zip”则打包类型为“zip”

----结束

#### Maven CLI上传包文件

下载并配置发布仓库账号及密码。

在页面上单击“获取密码”按钮下载账号及密码xml文件。



配置Maven setting.xml文件。

a. 将下载的xml文件中“servers”标签下的“release”、“snapshots”、“group”仓库的用户ID、账号和密码填写到maven的“settings.xml”文件中“servers”标签下。



b. 同上，将下载的xml文件中“mirror”标签和“profile”标签分别拷贝到“settings.xml”文件中“mirrors”标签和“profiles”标签下。

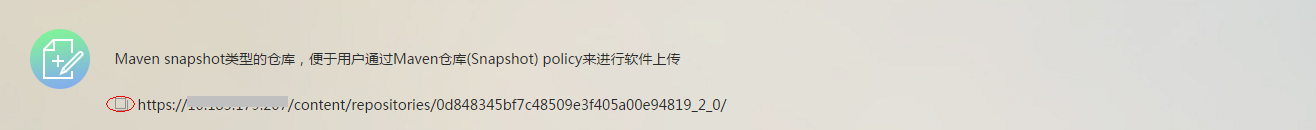
配置发布仓库地址。

单击“Maven仓库(Release)”和“Maven仓库(Snapshot)”的复制按钮，可以拷贝仓库url。

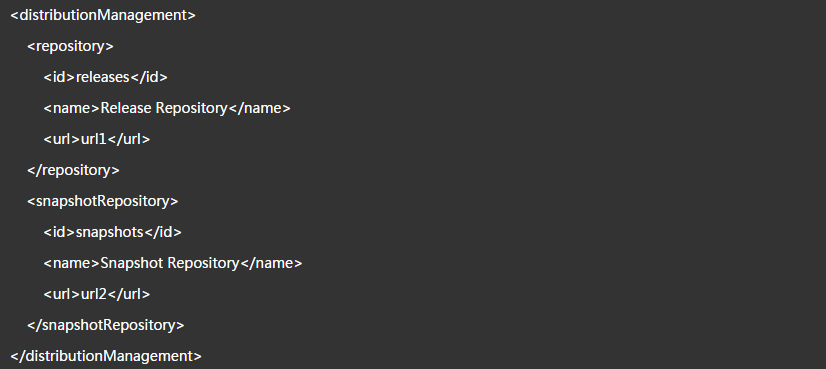
Maven仓库(Release)：



Maven仓库(Snapshot)：



分别将拷贝的url粘贴到Maven项目的“pom.xml”里“distributionManagement”节点对应的“url”标签下。



执行“mvn deploy”命令上传Maven项目。

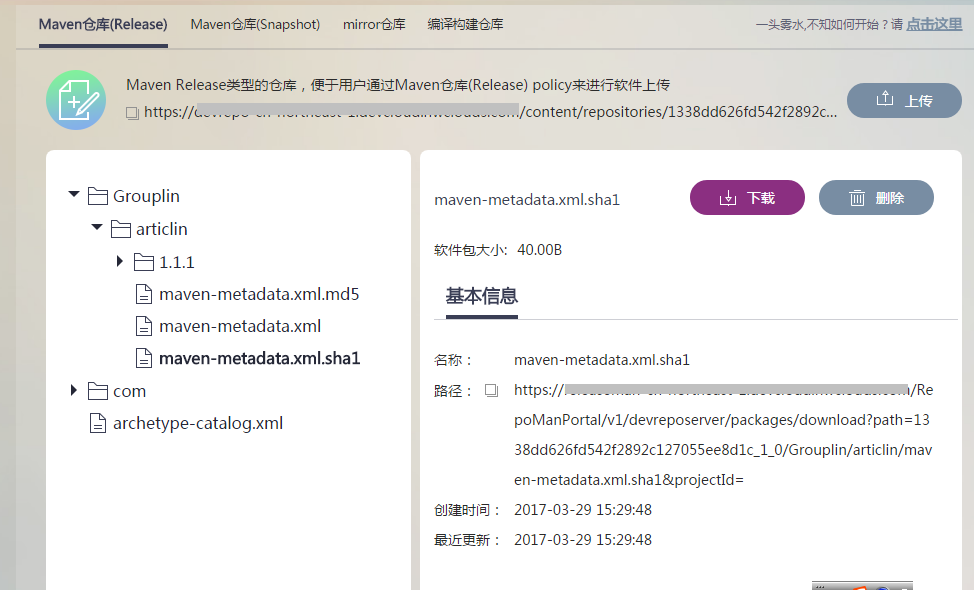
在命令窗口进入Maven项目“pom.xml”所在目录，执行“mvn deploy”即可将包上传到仓库里。

----结束

### 依赖包文件下载

#### 页面下载

在私有仓库页面选择要下载的包文件，通过“下载”按钮下载。



**----结束**

#### Mirror仓和私有仓（Release）依赖下载配置

设置mirror仓地址，直接拷贝“第三方依赖仓库”里的地址到maven的“settings.xml”中的“mirrors”节点的“url”标签下。

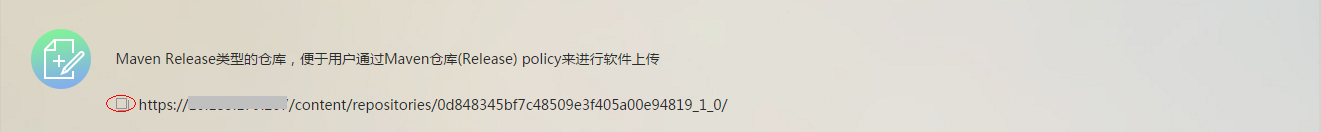




在maven的“settings.xml”中的“servers”标签中添加开源仓的账户及密码，添加几个开源仓就对应增加几个“server”节点。“id”统一填写为“mirror”，开源仓账号及密码默认相同，如jcenter的用户名密码都为jcenter，maven2的用户名密码都是maven2。

jiejie

私有仓（release）依赖下载配置，由于release仓的“servers”标签中的仓库账号密码已配置过，仅需在Maven的“setting.xml”文件中“mirrors”标签下新增“mirror”标签，并将release仓库的地址url拷贝到标签的url中即可，“id”填写为“release”。





配置完成后，在本地Maven工具构建时，即可自动通过发布仓库下载依赖包文件。

----结束

# 实验九：流水线

[10.1 实验介绍](#_ZH-CN_TOPIC_0069562543)

[10.2 实验任务](#_ZH-CN_TOPIC_0069562324)

## 实验介绍

### 关于本实验

本实验介绍了流水线是如何将代码仓库、编译构建、代码检查、部署以及子流水线等项目活动进行串行或并行执行的。

### 实验目的

* 掌握如何新建和配置流水线任务。
* 掌握如何设置流水线执行参数。
* 掌握如何执行流水线。

### 前提条件

拥有华为云账号，并创建有项目。

### 实验任务列表

| **任务** | **子任务** | **使用场景** |
| --- | --- | --- |
| 10.2.1 创建流水线 | 添加build阶段 | 添加代码检查任务，beta阶段进行代码检查。 |
| 添加编译构建任务，beta阶段进行代码检查。 |
| 添加beta阶段 | 添加构建任务，beta阶段进行产品包部署。 |
| 10.2.2 执行流水线 | 执行流水线 | 执行流水线，任务可以串行或者并行执行。 |
| 10.2.3 修改流水线参数 | 修改流水线参数 | 修改流水线相关信息，执行不同状态下的流水线。 |

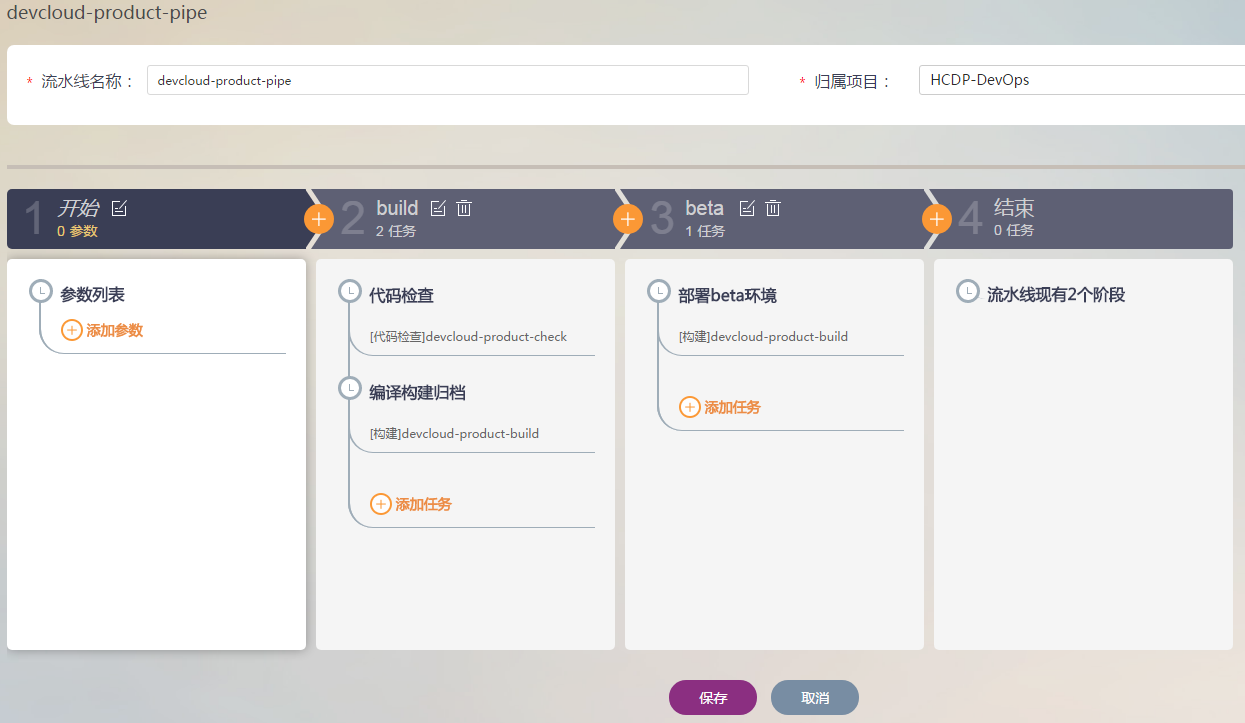
## 实验任务

### 创建流水线

新建名为“devcloud-product-pipe的流”水线任务，并分别添加build阶段和beta阶段。

在build阶段中添加代码检查任务和编辑构建任务。

在beta阶段中添加部署任务。



----结束

### 执行流水线

单击“开始执行”按钮，执行流水线。

流水线执行完成后，可查看流水线执行历史。

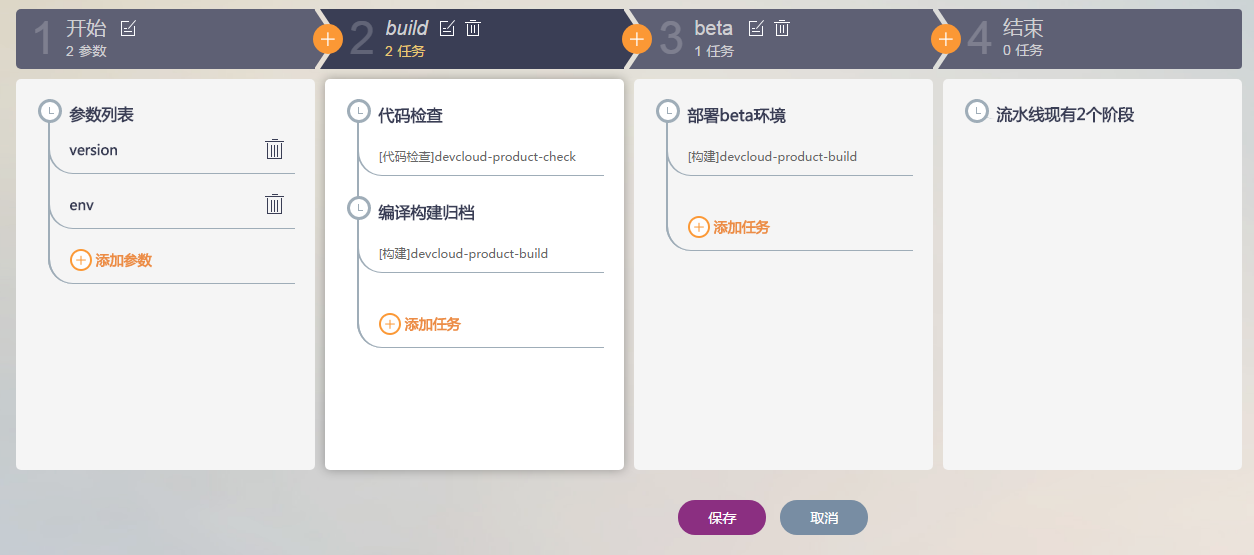


----结束

### 修改流水线参数

单击流水线名称旁边的修改按钮，进入流水线编辑页面。

单击流水线“开始”阶段的的修改按钮。修改完成后，单击“保存”按钮保存。



修改流水线参数，如下图：



----结束