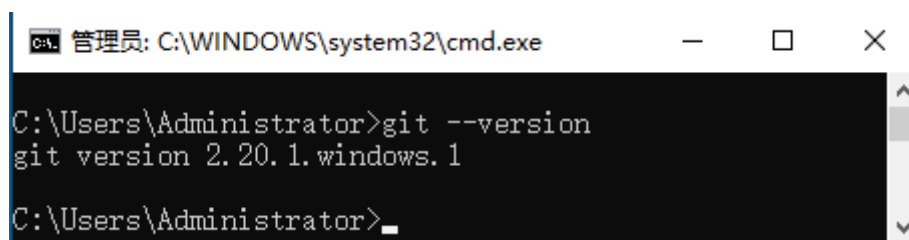


持续集成环境部署手册

1. Git

1.1 安装Git

1. 下载安装包，下载地址：<https://git-scm.com/downloads>
2. 双击运行，然后按默认选项安装即可
3. 校验是否安装成功。打开命令行窗口，输入命令：`git --version`，看到git版本信息表示已安装成功



```
管理员: C:\WINDOWS\system32\cmd.exe
C:\Users\Administrator>git --version
git version 2.20.1.windows.1
C:\Users\Administrator>
```

1.2 配置Git

安装完成后，还需要配置邮箱和用户名，在命令行输入：

提示：执行命令之前替换成自己的邮箱地址和用户名

```
git config --global user.email "email@example.com"
git config --global user.name "Your Name"
```

2. Gitee

2.1 在Gitee中创建项目

1. 注册码云账号，如果已注册直接登录即可。网址：<https://gitee.com/>
2. 点击按钮创建一个仓库



3. 输入仓库名称，点击‘Create repository’按钮

新建仓库

仓库名称  输入项目名称

apiTest001

归属  / 路径

apiTest001

仓库地址: <https://gitee.com/liujing/apiTest001>

仓库介绍 非必填

用简短的语言来描述一下吧

是否开源

☐ 私有 ☒ 公开

任何人都可以访问该仓库的代码和其他任何形式的资源

选择语言 Python

添加 .gitignore 请选择 .gitignore 模板

添加开源许可证  点此快速选择许可证

☒ 使用Readme文件初始化这个仓库

☐ 使用Issue模板文件初始化这个仓库 

☐ 使用Pull Request模板文件初始化这个仓库 

选择分支模型 (仓库初始化后将根据所选分支模型创建分支)

单分支模型 (只创建 master 分支)

导入已有仓库

创建

4. 进入仓库详情页面，点击‘克隆/下载’，再选择‘复制’即可得到项目地址

 开源软件 企业版  高校版 博客 我的码云

搜索项目

liujing / apiTest001 Python

Unwatch 1 Star 0 Fork 0

代码 Issues 0 Pull Requests 0 附件 0 Wiki 0 统计 DevOps 服务 管理

你当前开源项目尚未选择许可证 (LICENSE)，点此选择并创建开源许可

暂无描述 编辑

1 次提交 1 个分支 0 个标签 0 个发行版 1 位贡献者

master + Pull Request + Issue 文件 Web IDE 挂件

最后提交于 2分钟前 Initial commit

README.en.md Initial commit

README.md Initial commit

克隆/下载

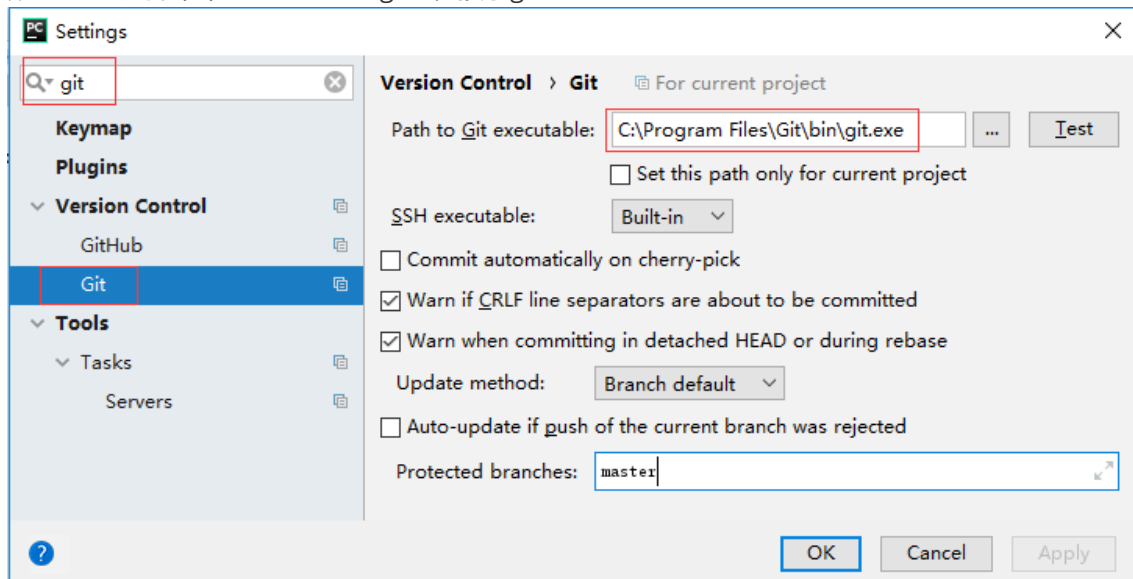
HTTPS SSH SVN SVN+SSH

<https://gitee.com/liujing/apiTest001.git> 复制

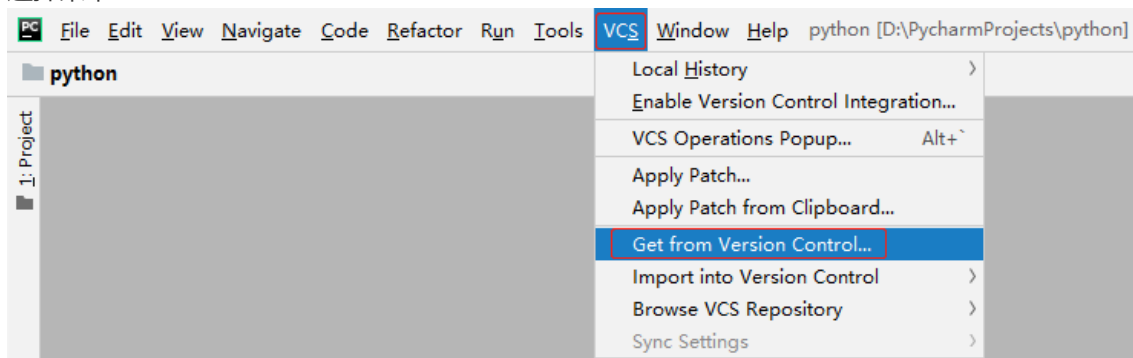
下载ZIP

2.2 在PyCharm中检出Gitee中的项目

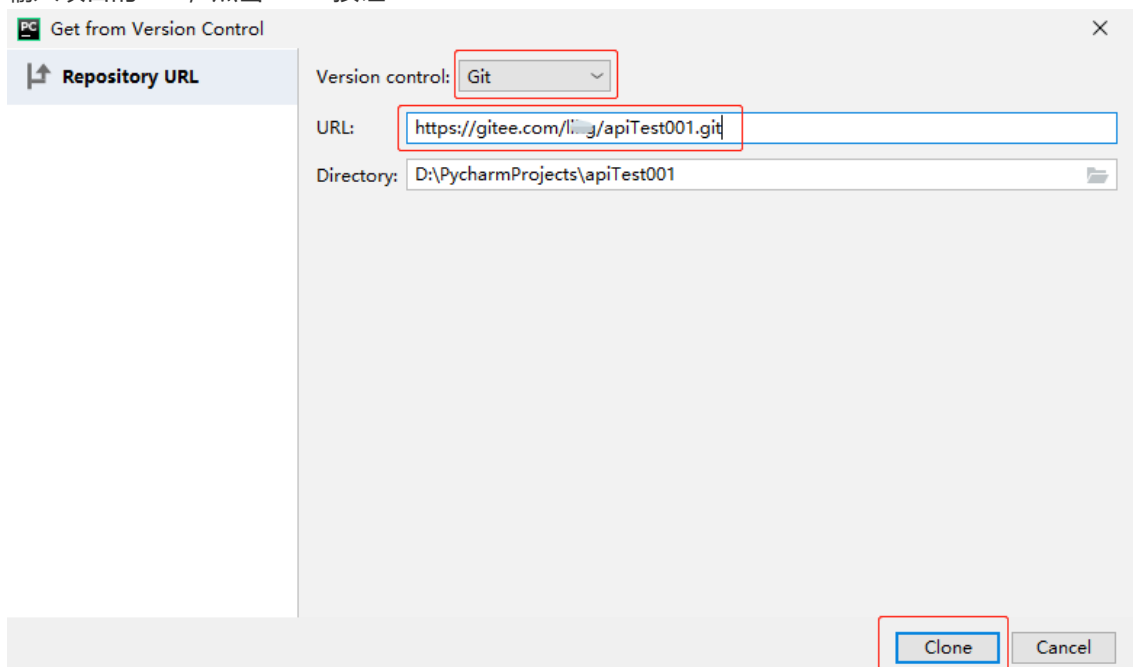
1. 打开PyCharm
2. 配置Git。选择菜单'File'-->'Settings...', 搜索git



3. 选择菜单'VCS'-->'Get from Version Control...'



4. 输入项目的URL, 点击Clone按钮

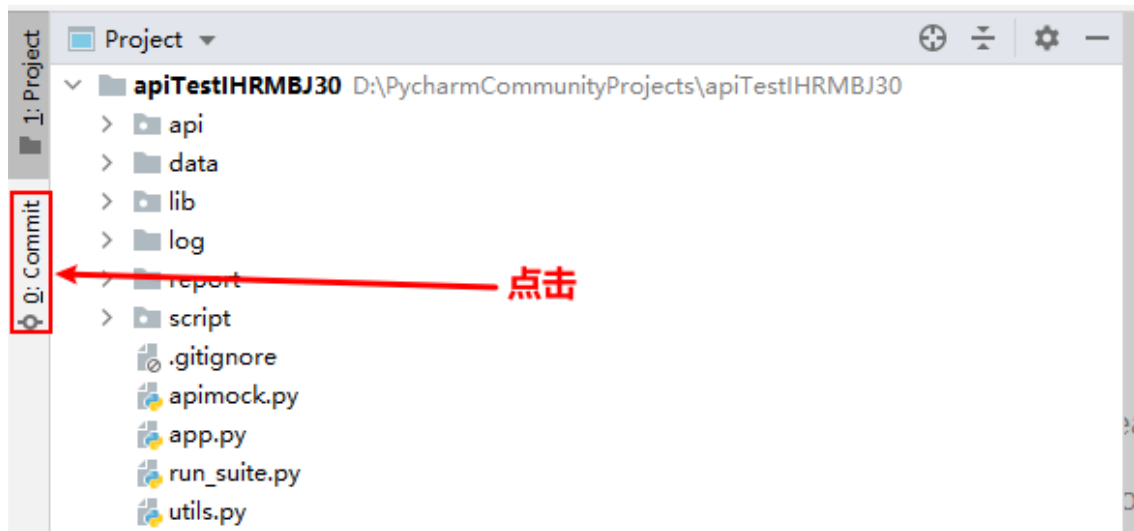


2.3 在PyCharm中编辑项目文件

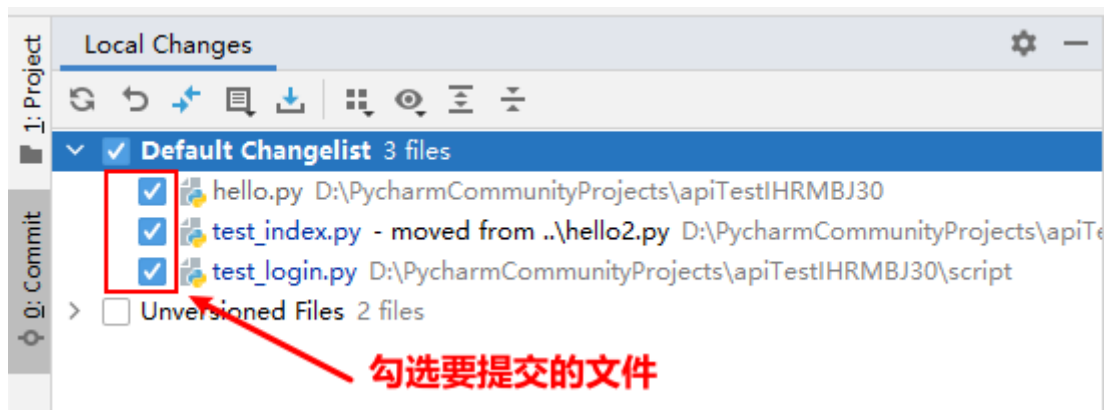
2.3.1 把本地文件提交到服务器

本地新建或修改的文件都可以提交到代码服务器

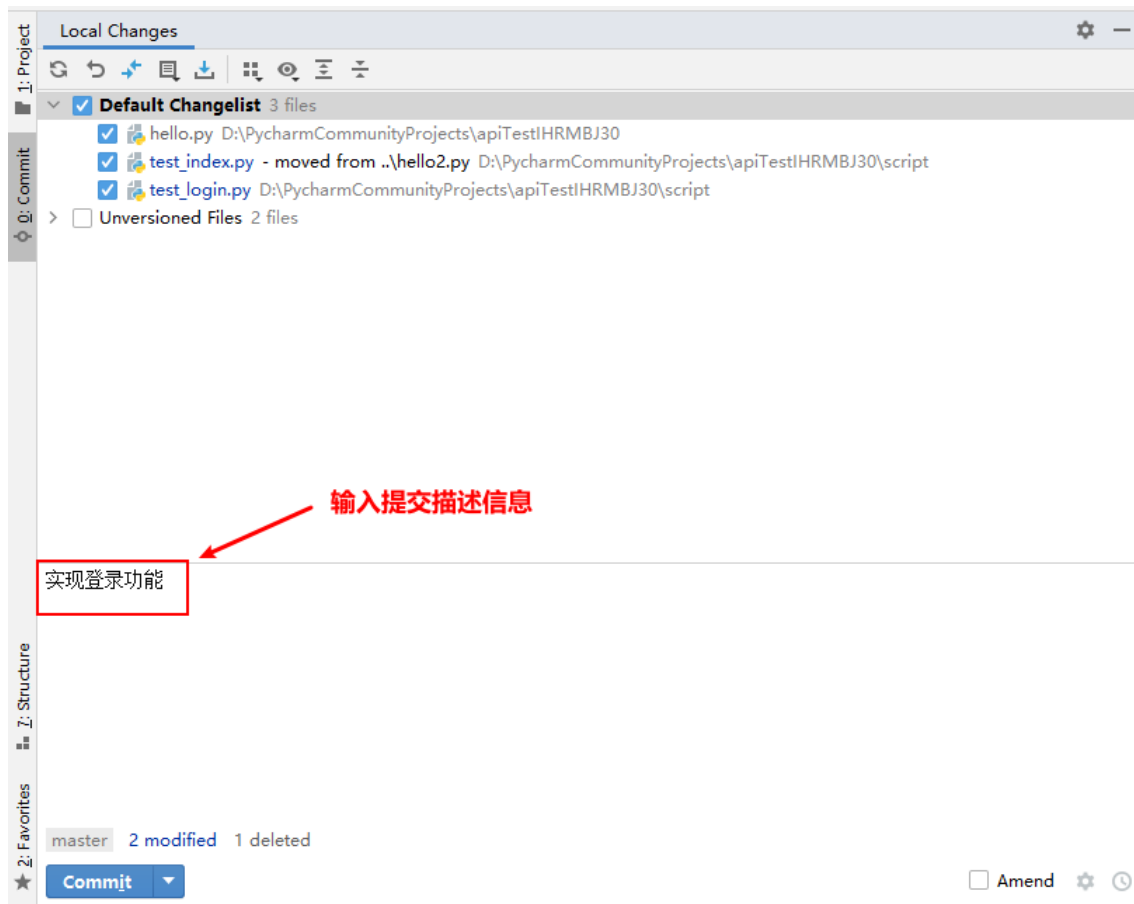
1. 切换到“Commit”窗口



2. 勾选要提交的文件



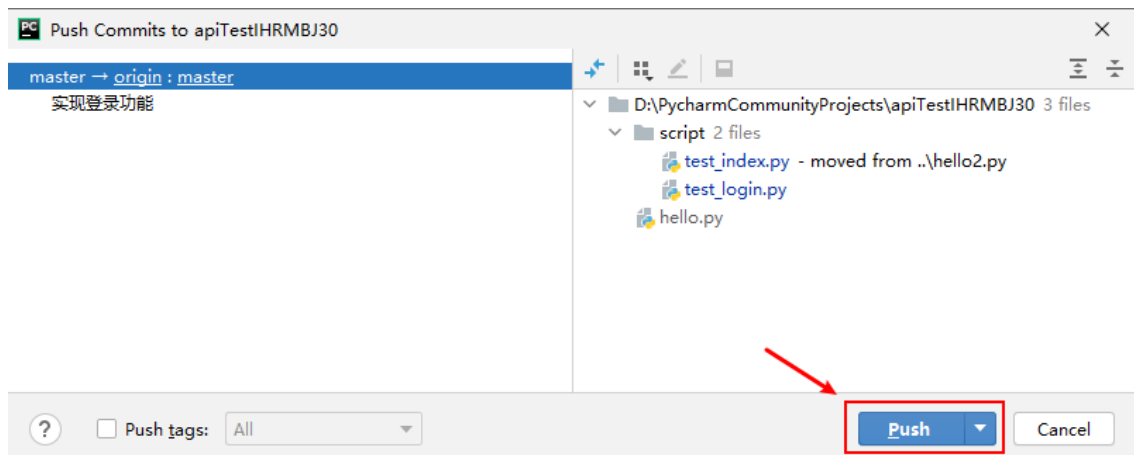
3. 输入提交信息



4. 点击“Commit”按钮后面的选择下拉选项选择“Commit and Push...”提交并推送代码



5. 在弹出的推送窗口中点击“Push”按钮推送

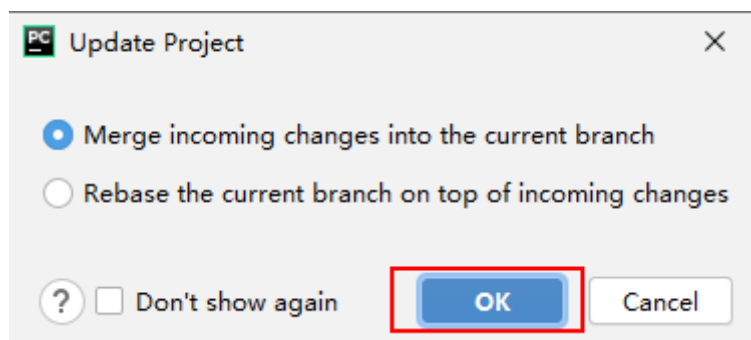


2.3.2 从服务器中拉取文件

1. 点击“Update Project...”图标



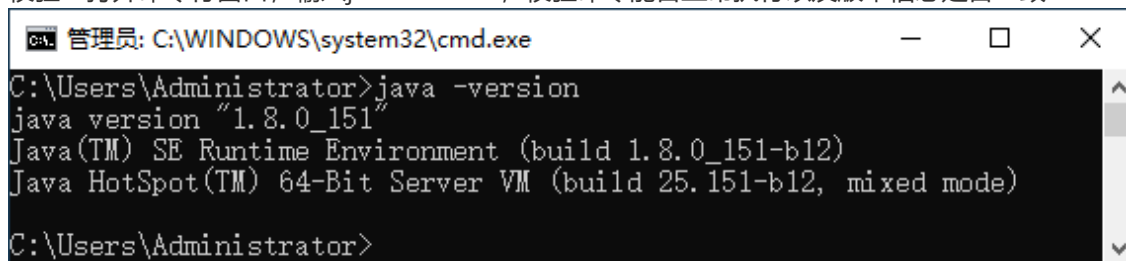
2. 点击Update Project窗口中的“OK”开始拉取代码



3. Jenkins环境搭建

3.1 安装JDK

- 版本：JDK1.8
- 下载地址：<http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>
- 配置环境变量
 - JAVA_HOME=C:\Program Files\Java\jdk1.8.0_151
 - 在Path中添加:%JAVA_HOME%\bin;
- 校验：打开命令行窗口，输入java -version，校验命令能否正常执行以及版本信息是否一致



3.2 安装Jenkins

3.2.1 在线安装

1. 下载最新的版本war包，官网：<https://jenkins.io/>
2. 进入war包所在目录，命运行运行

```
java -jar jenkins.war
java -jar jenkins.war --httpPort=8888
```

默认端口是8080
可以用httpPort来指定端口

3. 在浏览器中访问<http://localhost:8080/>

Getting Started

Unlock Jenkins

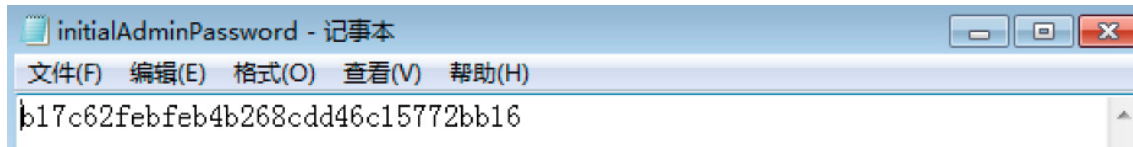
To ensure Jenkins is securely set up by the administrator, a password has been written to the log (not sure where to find it?) and this file on the server:

C:\Program Files (x86)\Jenkins\secrets\initialAdminPassword

Please copy the password from either location and paste it below.

Administrator password

4. 进入启动页后，这里需要一个密码进行验证，密码所在的位置会在页面上标有，直接打开这个文件就可以看到密码



5. 密码输入后，点击继续。此时会提示是否安装插件，我们选择第一个，安装建议的插件

Getting Started

Customize Jenkins

Plugins extend Jenkins with additional features to support many different needs.

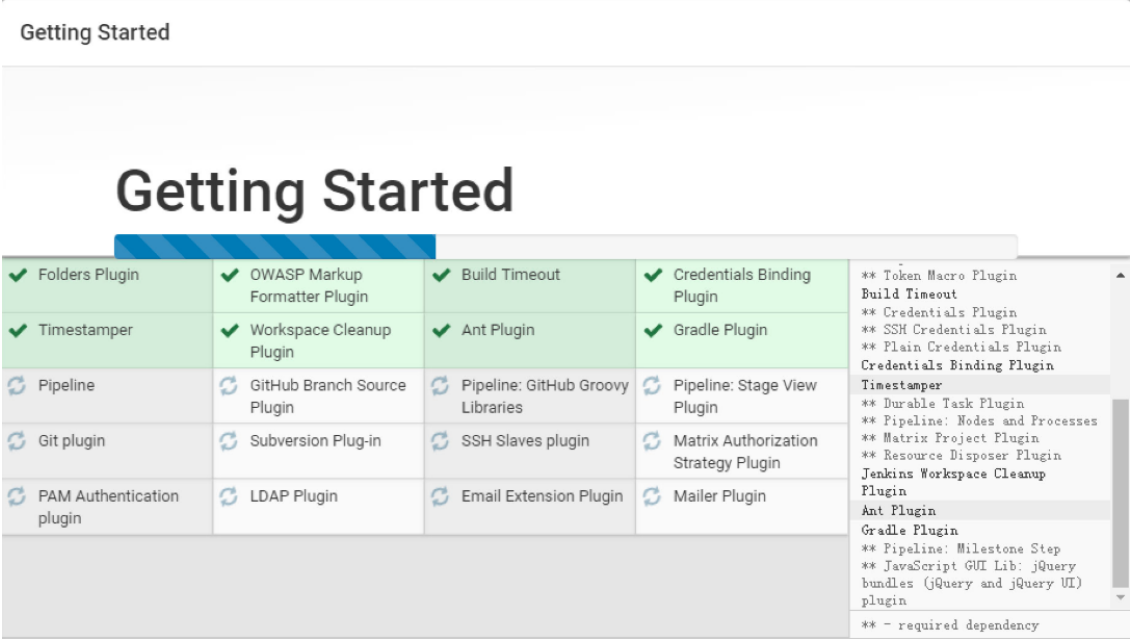
Install suggested plugins

Install plugins the Jenkins community finds most useful.

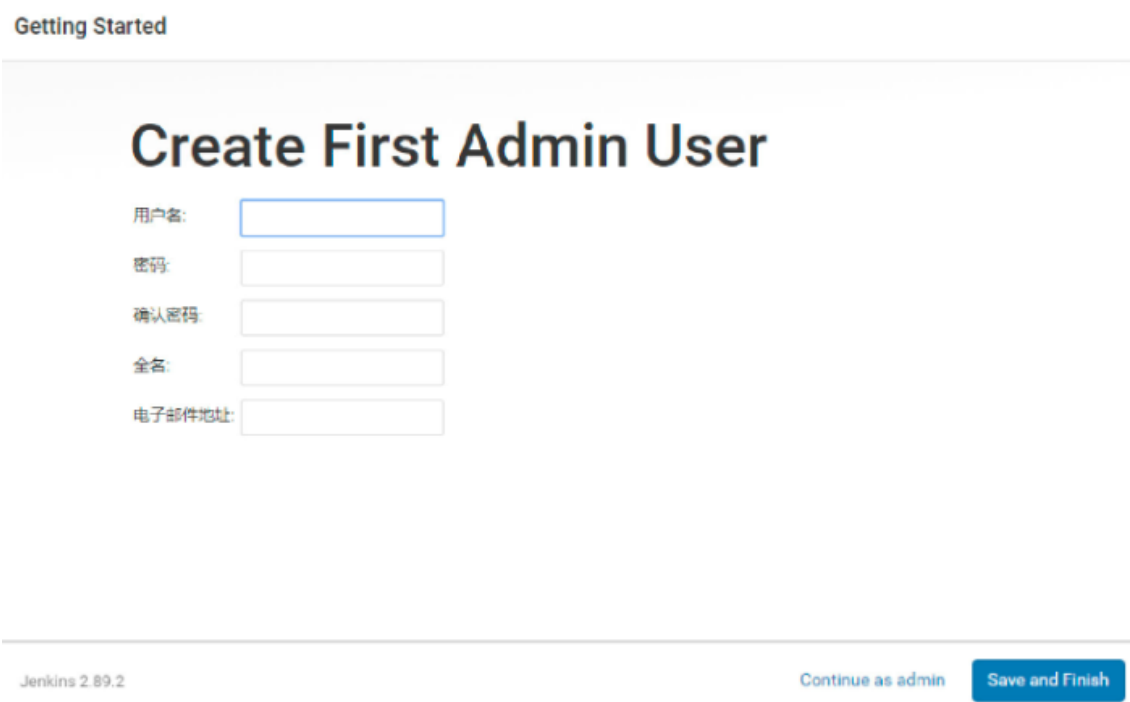
Select plugins to install

Select and install plugins most suitable for your needs.

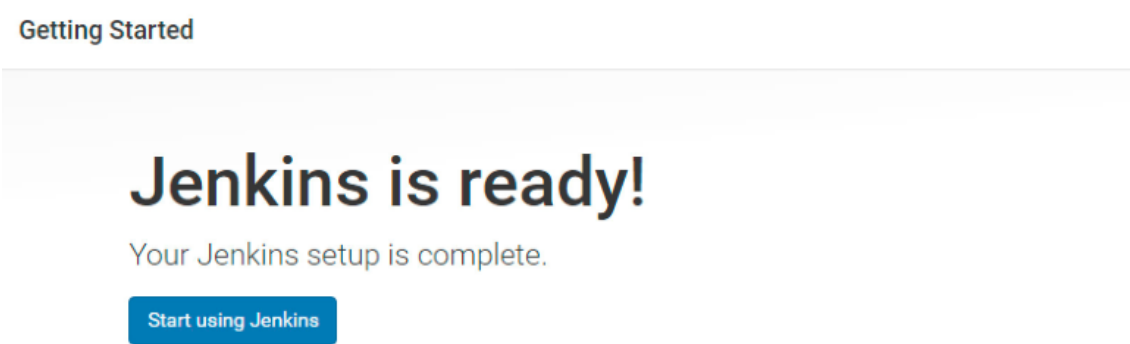
6. 等待插件安装完成



7. 安装完成后，会提示设置用户名和密码，进行后期登录使用



8. 配置完成后，提示准备就绪



9. 点击按钮，即可进入首页



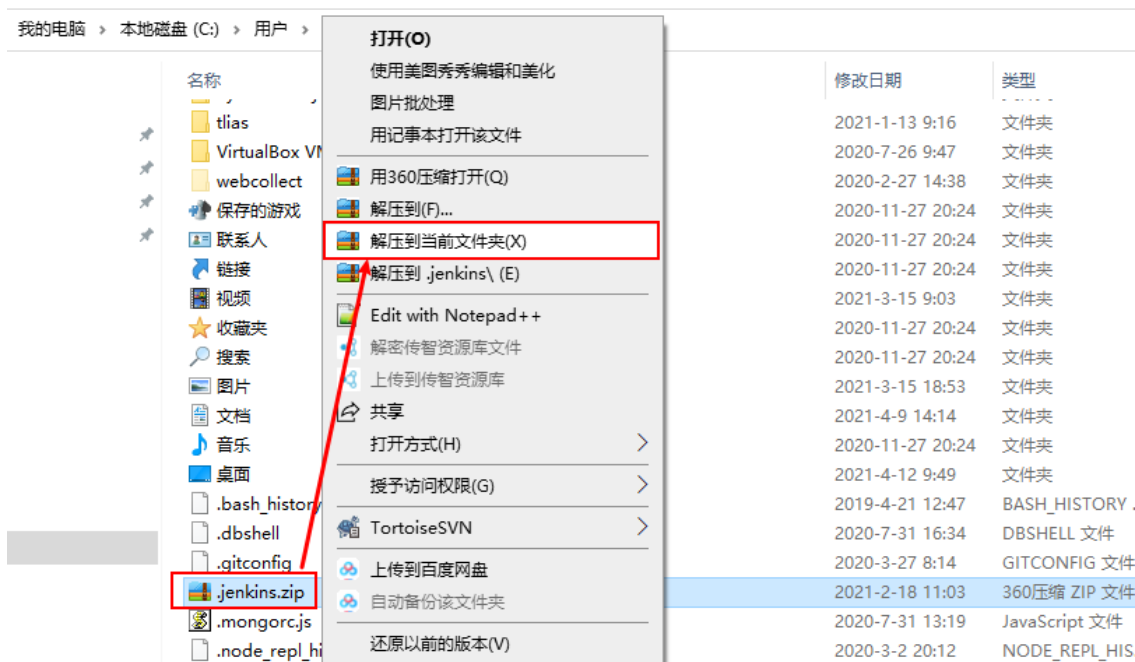
3.2.2 离线安装

1. 从提供的软件安装包中找到“.jenkins.zip”压缩包，把该压缩包放到当前用户的目录下。

不同用户的目录名称会有差异，类似路径：C:\Users\Administrator\

如果解压之前，用户目录下已经存在“.jenkins”目录，先删除即可

2. 把压缩包解压到当前目录中



3. 进入war包所在目录，命运行运行：`java -jar jenkins.war`

4. 在浏览器中访问<http://localhost:8080/>，进入登录页面后输入用户名和密码进行登录

用户名：admin 密码：123456



欢迎来到 Jenkins!

admin

登录

☐ 保持登录状态

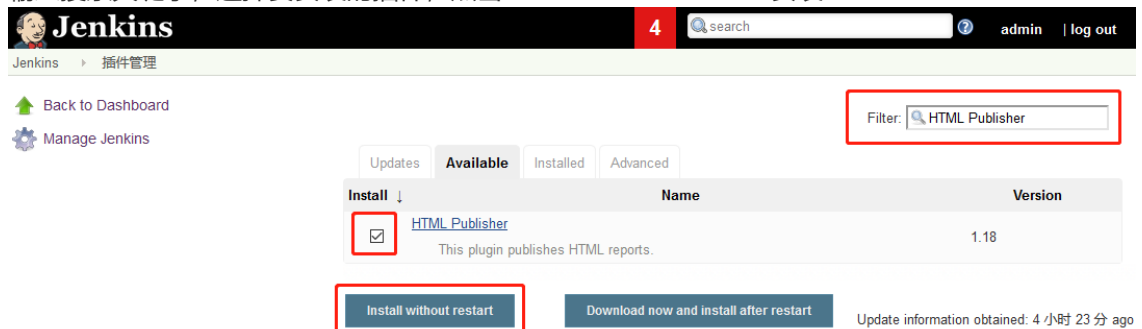
3.3 Jenkins安装插件

以安装HTML Publisher插件为例进行演示，其他插件安装方式都是一样的

1. 进入首页，选择'Jenkins'后面的小图标->'系统管理'->'管理插件'



2. 输入搜索关键字，选择要安装的插件，点击'Install without restart'安装



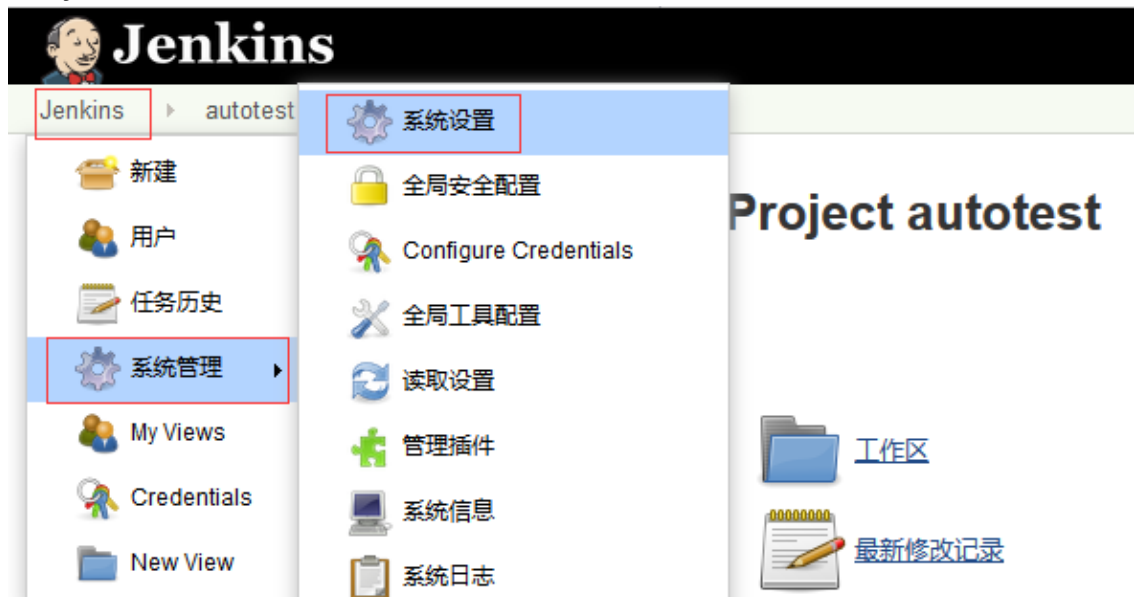
3. 查看安装进度



The screenshot shows the Jenkins installation progress page. The top navigation bar includes the Jenkins logo, a search bar, and user links for 'admin' and 'log out'. The main content area is titled 'Installing Plugins/Upgrades'. A red box highlights the 'Preparation' section, which lists the following steps: '检查网络连接' (Check network connection), '检查更新中心连接' (Check update center connection), and '成功' (Success). Below this, the 'HTML Publisher' plugin is shown with a 'Success' status. At the bottom, there are two green arrows with instructions: 'Go back to the top page (you can start using the installed plugins right away)' and 'Restart Jenkins when installation is complete and no jobs are running'.

3.4 系统设置

1. 选择'Jenkins'-'>'系统管理'-'>'系统设置'



The screenshot shows the Jenkins interface with the 'System Settings' (系统设置) menu item highlighted. The left sidebar contains a list of navigation items: '新建' (New), '用户' (Users), '任务历史' (Task History), '系统管理' (System Management), 'My Views', 'Credentials', and 'New View'. The '系统管理' item is highlighted with a red box. The main content area shows the 'Project autotest' configuration page. The '系统设置' menu is open, showing a list of settings: '全局安全配置' (Global Security Configuration), 'Configure Credentials', '全局工具配置' (Global Tool Configuration), '读取设置' (Load Settings), '管理插件' (Manage Plugins), '系统信息' (System Information), and '系统日志' (System Log).

2. 配置'系统管理员邮件地址'

Jenkins Location

Jenkins URL

⚠ Please set a valid host name, instead of localhost

系统管理员邮件地址

3. 配置'Extended E-mail Notification'

Extended E-mail Notification

SMTP server	<input type="text" value="smtp.163.com"/>	?
Default user E-mail suffix	<input type="text" value="@163.com"/>	?
<input checked="" type="checkbox"/> Use SMTP Authentication		?
User Name	<input type="text" value="hello@163.com"/>	
Password	<input type="password" value="..."/>	
Use SSL	<input type="checkbox"/>	?
SMTP port	<input type="text"/>	?
Charset	<input type="text" value="UTF-8"/>	
Default Content Type	<input type="text" value="HTML (text/html)"/>	?
<input type="checkbox"/> Use List-ID Email Header		?
<input type="checkbox"/> Add 'Precedence: bulk' Email Header		?
Default Recipients	<input type="text"/>	?

4. 配置'邮件通知'

邮件通知

SMTP服务器	<input type="text" value="smtp.163.com"/>	?
用户默认邮件后缀	<input type="text" value="@163.com"/>	?
<input checked="" type="checkbox"/> 使用SMTP认证		?
用户名	<input type="text" value="hello@163.com"/>	
密码	<input type="password" value="..."/>	
使用SSL协议	<input type="checkbox"/>	?
SMTP端口	<input type="text"/>	?
Reply-To Address	<input type="text"/>	
字符集	<input type="text" value="UTF-8"/>	
<input type="checkbox"/> 通过发送测试邮件测试配置		

4. Jenkins持续集成实现

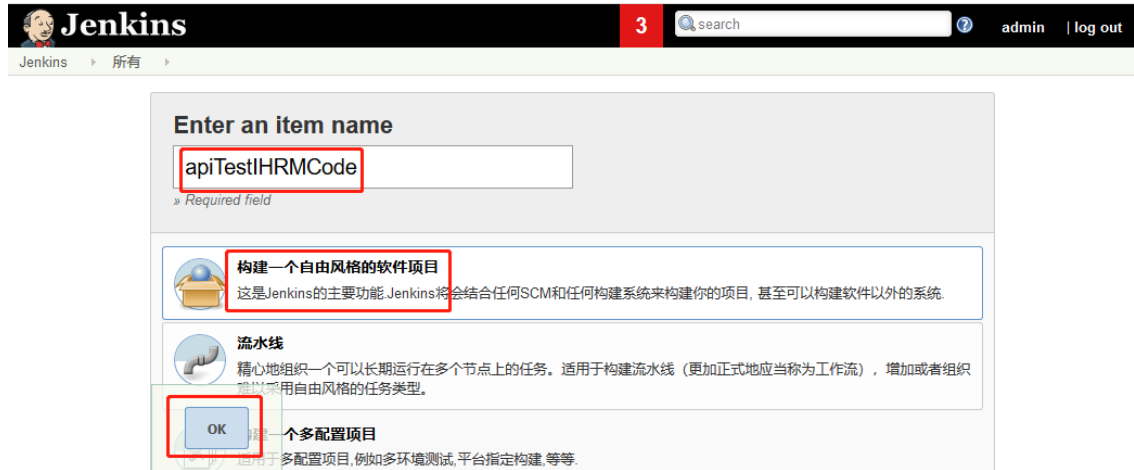
4.1 创建任务



1. 点击'New 任务'菜单

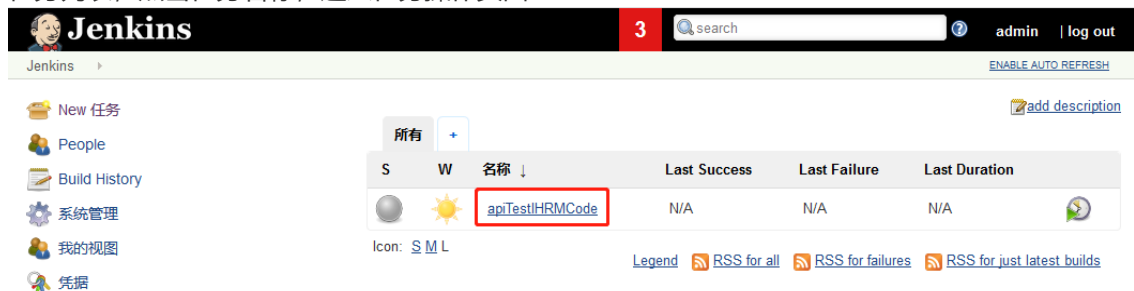


2. 输入任务名称，选择'构建一个自由风格的软件项目'，点击'确定'



The image shows the Jenkins 'Enter an item name' dialog box. The title is 'Enter an item name'. There is a text input field containing 'apiTestIHRMCode'. Below the input field, it says '» Required field'. There are three radio button options: '构建一个自由风格的软件项目' (Build a free-style software project), '流水线' (Pipeline), and '多配置项目' (Multi-configuration project). The first option is selected. Below the options, there is an 'OK' button. The descriptions for each option are: '构建一个自由风格的软件项目' (This is the main function of Jenkins. Jenkins will combine any SCM and any build system to build your project, even build software outside the system.), '流水线' (Carefully organize a task that can run on multiple nodes. Suitable for building pipelines (more formally should be called workflows), increase or organize tasks using free-style task types.), and '多配置项目' (Multiple configuration projects, such as multi-environment testing, platform-specific builds, etc.).

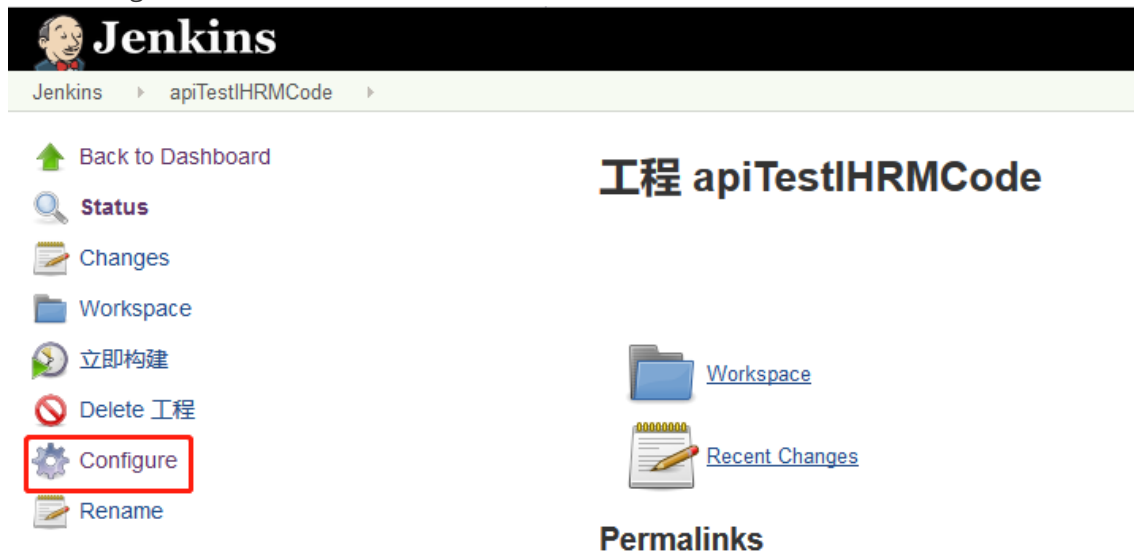
3. 任务列表，点击任务名称，进入任务操作页面



The image shows the Jenkins task list. The title is 'Jenkins'. There is a search bar and a '3' button. The table has columns: 'S', 'W', '名称 ↓', 'Last Success', 'Last Failure', and 'Last Duration'. The first row has a sun icon, a sun icon, 'apiTestIHRMCode', 'N/A', 'N/A', and 'N/A'. Below the table, there is a 'Legend' and 'RSS for all', 'RSS for failures', and 'RSS for just latest builds' links.

S	W	名称 ↓	Last Success	Last Failure	Last Duration
		apiTestIHRMCode	N/A	N/A	N/A

4. 点击'Configure'菜单，进入任务配置页面



The image shows the Jenkins task configuration page. The title is 'Jenkins'. The breadcrumb is 'Jenkins > apiTestIHRMCode >'. On the left, there is a sidebar with links: 'Back to Dashboard', 'Status', 'Changes', 'Workspace', '立即构建' (Build Now), 'Delete 工程' (Delete Project), 'Configure' (highlighted with a red box), and 'Rename'. On the right, there is a section titled '工程 apiTestIHRMCode' with links 'Workspace' and 'Recent Changes'. Below this, there is a section titled 'Permalinks'.

5. 源码管理

General **Source Code Management** Build Triggers Build Environment Build Post-build Actions

Source Code Management

☐ 无
☒ **Git**

Repositories

Repository URL

Credentials

Advanced...
Add Repository

Branches to build

Branch Specifier (blank for 'any')

Add Branch

Repository browser (Auto)

6. 构建触发器

构建触发器

☐ 触发远程构建 (例如,使用脚本)

☐ Build after other projects are built

☒ Build periodically **定时构建**

日程表

☐ GitHub hook trigger for GITScm polling

☒ Poll SCM **轮询构建**

日程表

☐ 忽略钩子 post-commit

7. 添加构建，选择'执行 Windows 批处理命令'，输入执行脚本

The image shows two parts of the Jenkins 'Build' configuration interface. The top part shows the 'Add build step' dropdown menu, which lists various build actions. The option '执行 Windows 批处理命令' (Execute Windows batch command) is highlighted with a red box. The bottom part shows the configuration for this step. The 'Command' field contains the text 'python run_suite.py', which is also highlighted with a red box. Below the command field, there is a link to 'See the list of available environment variables' and an 'Advanced...' button. The 'Add build step' button is visible at the bottom left of the configuration area.

Build

Add build step ▼

- Execute Groovy script
- Execute system Groovy script
- GitHub PR: set 'pending' status
- Invoke Ant
- Invoke Gradle script
- Run with timeout
- Set build status to "pending" on GitHub commit
- 执行 Windows 批处理命令**
- 执行 shell
- 调用顶层 Maven 目标

Build

执行 Windows 批处理命令

Command: `python run_suite.py`

See [the list of available environment variables](#)

Advanced...

Add build step ▼

8. 添加构建后操作，点击‘Add post-build action’按钮，选择‘Publish HTML reports’，点击‘Add’按钮，输入配置信息

The image shows the Jenkins configuration interface. At the top, a dropdown menu is open, listing various post-build actions. The 'Publish HTML reports' option is highlighted with a red box. Below the menu, the 'Add post-build action' button is also highlighted with a red box. The main configuration area is titled 'Post-build Actions' and contains a list of actions. The 'Publish HTML reports' action is selected, and its configuration form is displayed. The form has a sidebar with 'Reports' and a main area with input fields. The 'HTML directory to archive' field is set to '/report' and the 'Index page[s]' field is set to 'report.html', both highlighted with red boxes. The 'Index page title[s] (Optional)' field is empty, and the 'Report title' field is set to 'HTML Report'. A 'Publishing options...' button is located at the bottom right of the configuration form. At the bottom of the page, the 'Add post-build action' dropdown menu is visible again.

GitHub PR: set PR status

Publish HTML reports

Publish JUnit test result report

归档成品

构建其他工程

记录文件的指纹用于追踪

Git Publisher

E-mail Notification

Editable Email Notification

Add post-build action

Post-build Actions

Publish HTML reports

Reports

HTML directory to archive:

Index page[s]:

Index page title[s] (Optional):

Report title:

Publishing options...

Add

Add post-build action

9. 添加构建后操作步骤，选择'Editable Email Notification'，然后配置相关信息

Aggregate downstream test results

Archive the artifacts

Build other projects

Publish HTML reports

Publish JUnit test result report

Record fingerprints of files to track usage

Git Publisher

E-mail Notification

Editable Email Notification

Set GitHub commit status (universal)

Set build status on GitHub commit [deprecated]

Delete workspace when build is done

test-report/\${BUILD_ID}

index.html

HTML Report

Publishing options...

增加构建后操作步骤

Editable Email Notification

Disable Extended Email Publisher

Project Recipient List

Project Reply-To List

Content Type

Default Subject

Default Content

Attachments

Attach Build Log

Additional groovy classpath

Save to Workspace

Triggers

Allows the user to disable the publisher, while maintaining the settings

123456@qq.com

Comma-separated list of email address that should receive notifications for this project.

\$DEFAULT_REPLYTO

Comma-separated list of email address that should be in the Reply-To header for this project.

HTML (text/html)

\$DEFAULT_SUBJECT

测试报告内容...

Can use wildcards like 'module/dist/**/*.zip'. See the [@includes of Ant files](#) for the exact format. The base directory is [the workspace](#).

Do Not Attach Build Log

增加

Save to Workspace

Always

Send To

Developers

Recipient List

Add

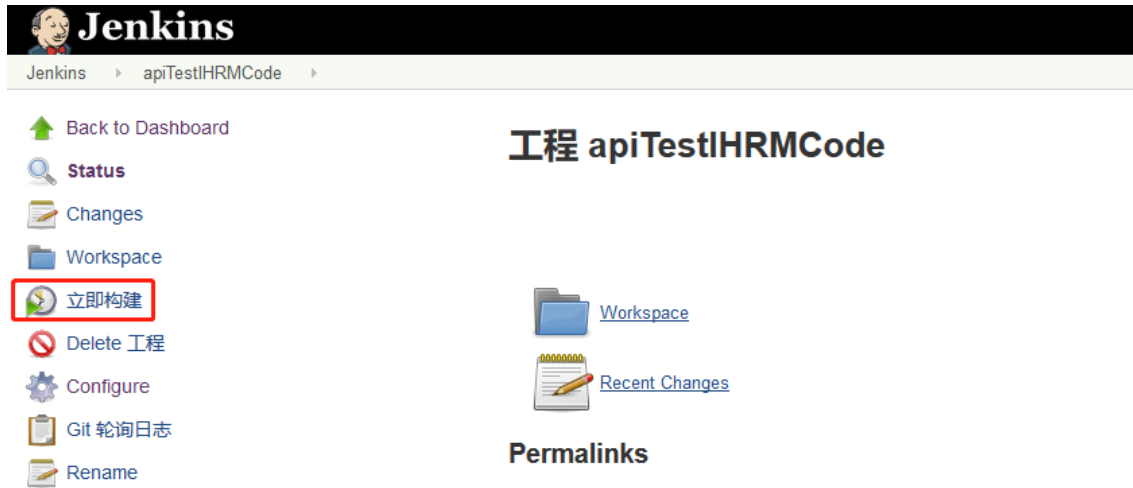
高级...

Add Trigger

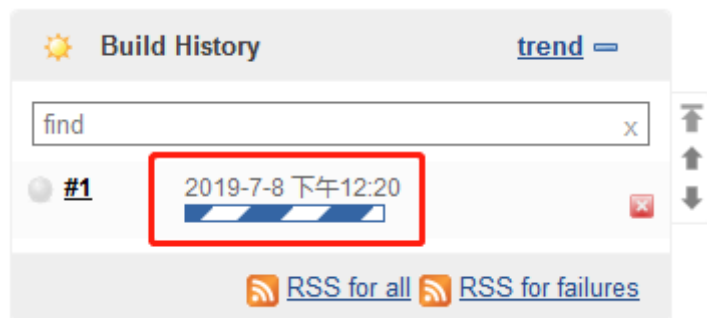
10. 点击'Apply'和'Save'按钮

4.2 手动构建

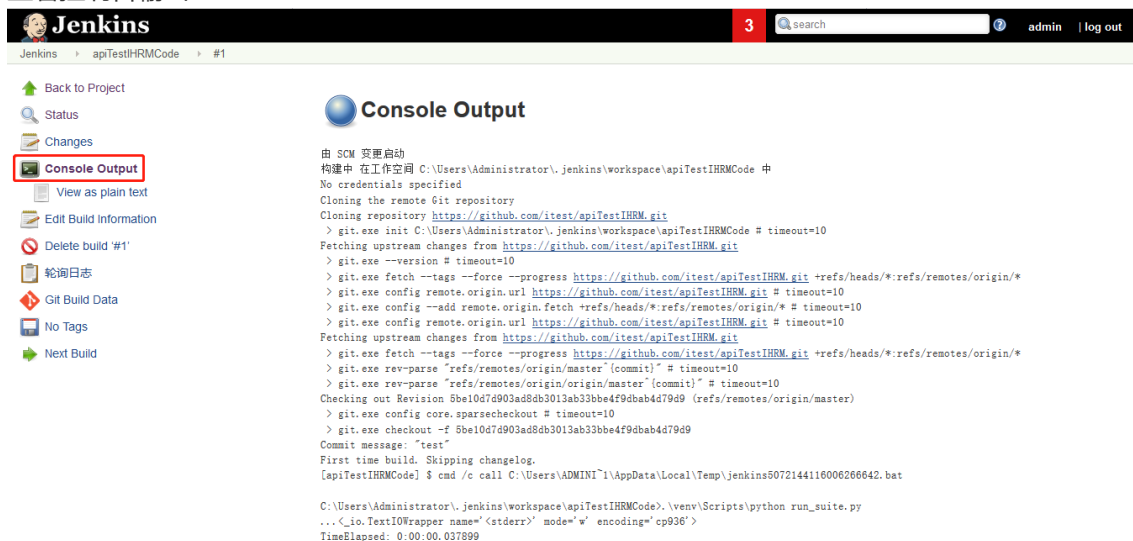
1. 进入项目操作页面，点击'立即构建'菜单



2. 构建进度条，点击查看构建进度



3. 查看控制台输出



4. 查看测试报告



Jenkins

apiTestIHRMCode

Back to Dashboard

Status

Changes

Workspace

立即构建

Delete 工程

Configure

HTML Report

Git 轮询日志

Rename

点击查看测试报告

工程 apiTestIHRMCode

HTML Report

Workspace

Recent Changes

Permalinks

5. HTML测试报告

[Back to apiTestIHRMCode](#) [report](#)

IHRM接口自动化测试报告

Start Time: 2019-07-08 12:21:21
Duration: 0:00:00.040890
Status: Pass 3

V1.0

Show [Summary](#) [Failed](#) [All](#)

Test Group/Test case	Count	Pass	Fail	Error	View
script.test_login.TestLogin	1	1	0	0	Detail
script.test_employee.TestEmployee	2	2	0	0	Detail
Total	3	3	0	0	