

手把手教你在Eclipse中使用JavaSDK

原创 王章 FISCO BCOS开源社区 2019-10-23



王章

FISCO BCOS核心开发者

IDE是第一生产力

— AUTHOR — 作者 —

Eclipse是当前主流的Java IDE之一，本文将进行上手实操，指导如何在Eclipse中创建FISCO BCOS的JavaSDK应用。

本文首先介绍如何在Eclipse创建新的工程，在工程中引入JavaSDK依赖，通过配置与区块链进行交互，最后通过获取区块链的块高验证创建的项目。同时，本文也会介绍实际开发中更加便捷的使用方式，将已经提供的示例项目导入Eclipse。

注意：

文章项目以Gradle方式进行构建，请确认以下环境正确安装：

1. Java 1.8或者以上版本。
2. Gradle 5.0或者以上版本。

前置条件：

参考安装文档：

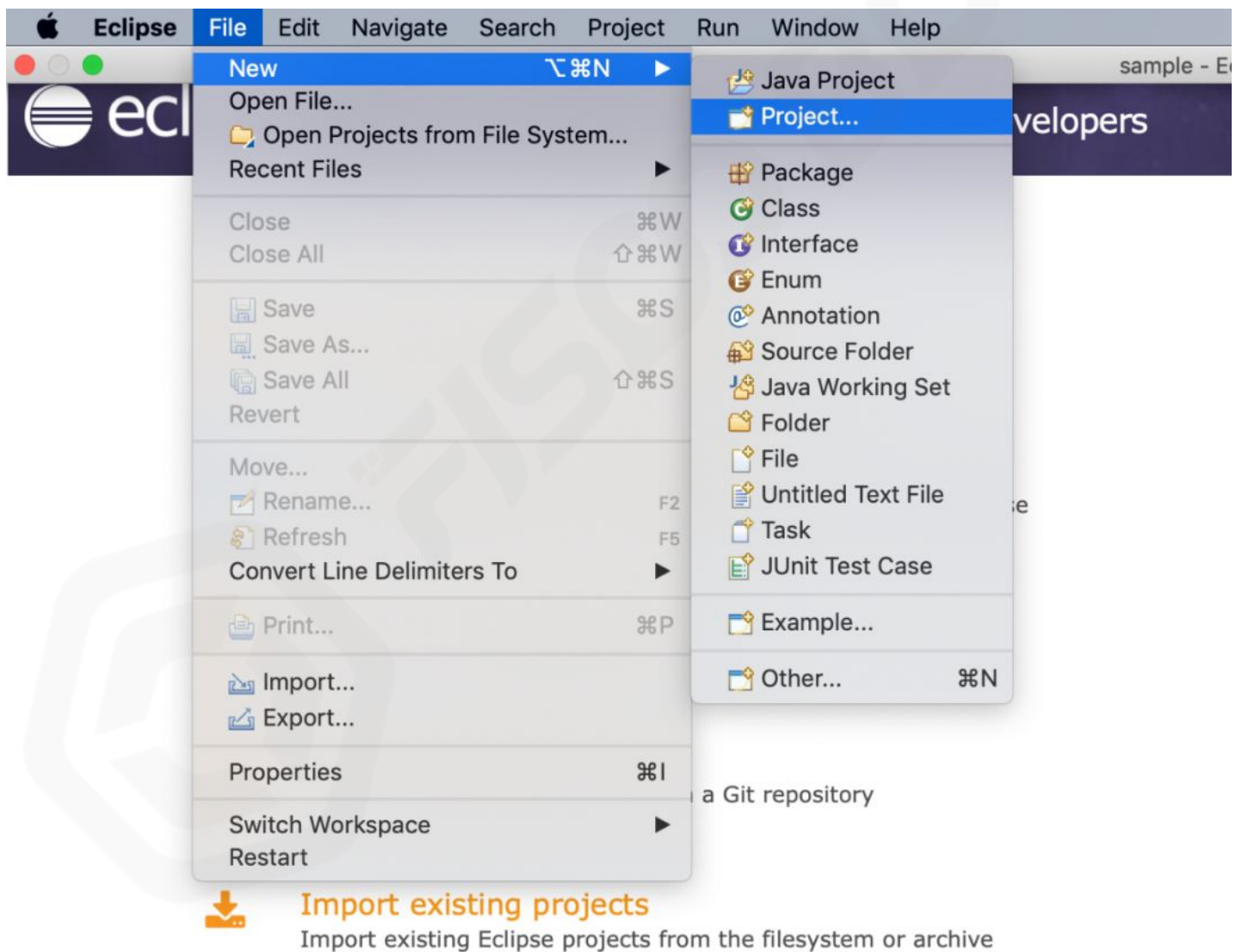
https://fisco-bcos-documentation.readthedocs.io/zh_CN/latest/docs/installation.html

完成FISCO BCOS区块链的搭建，功能测试需要与区块链节点进行交互。

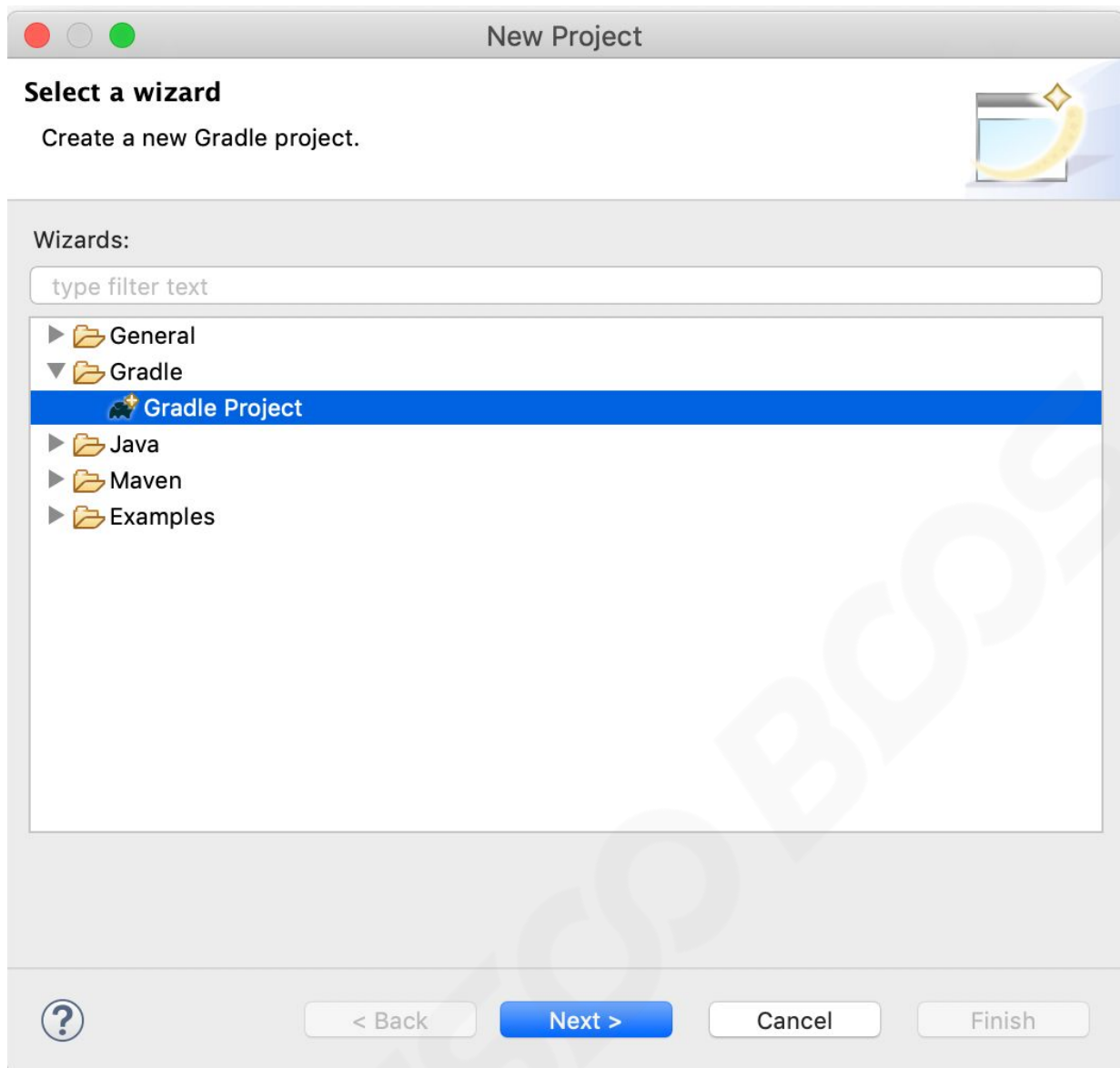
Eclipse 创建新工程

创建Gradle工程

打开Eclipse，右键选择New下Project选项，如下图所示：



在弹出的新建New Project界面，选择Gradle => Gradle Project，点击Next：



再次点击Next，在接下来的界面输入项目名称，这里项目名为sample。

New Gradle Project



Specify the name of the Gradle project to create.

Project name

Project location

☒ Use default location

Location

[Browse...](#)

Working sets

☐ Add project to working sets

[New...](#)

Working sets

[Select...](#)

Click the Finish button to create the project and import it into the workspace. Click the Next button to select optional options.



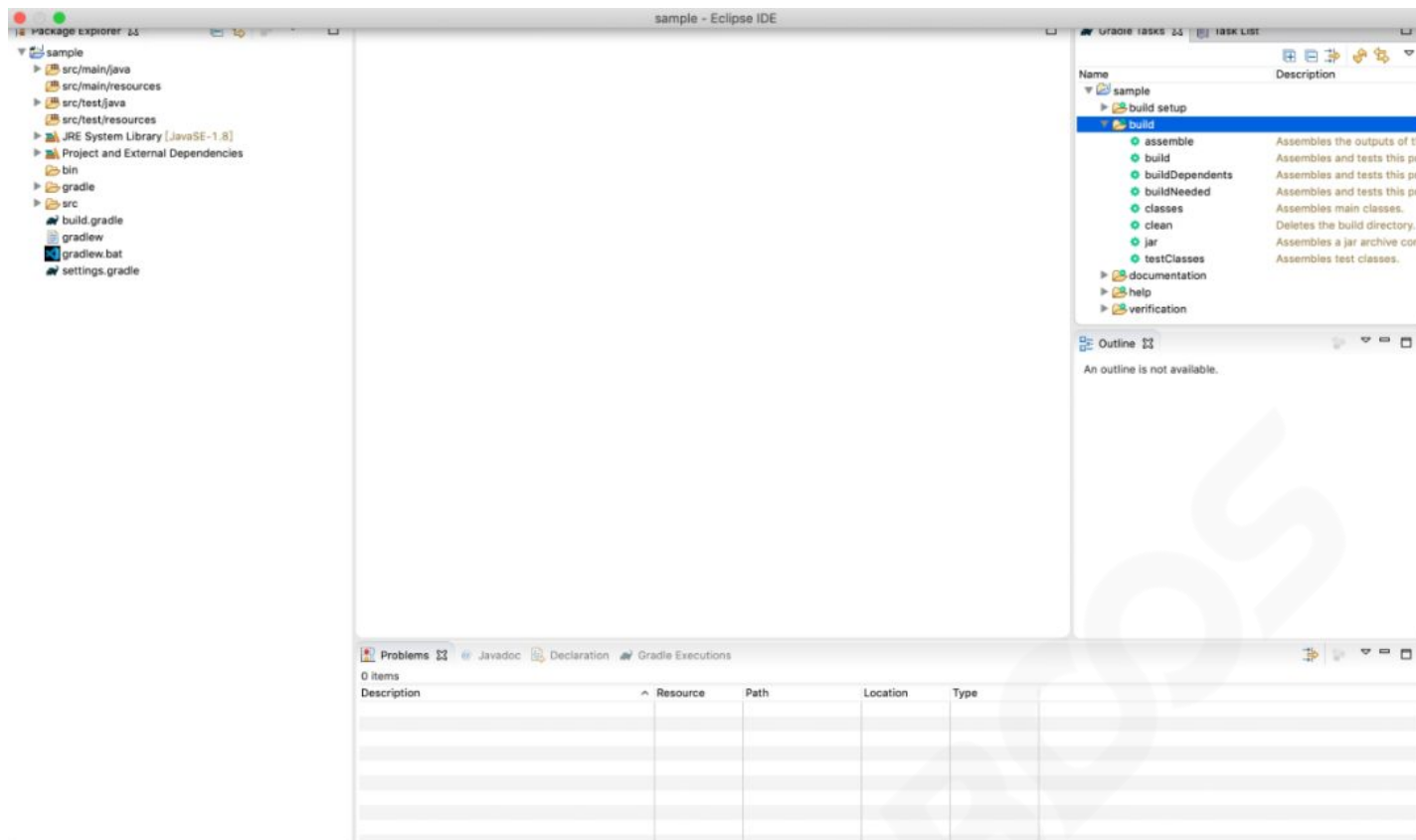
< Back

Next >

Cancel

Finish

点击Finish，然后等待项目初始化。



项目初始化之后，一个新的Gradle工程已经创建完成。

接下来，我们需要修改项目的build.gradle文件，引入JavaSDK的maven库以及依赖版本。

项目引入JavaSDK依赖

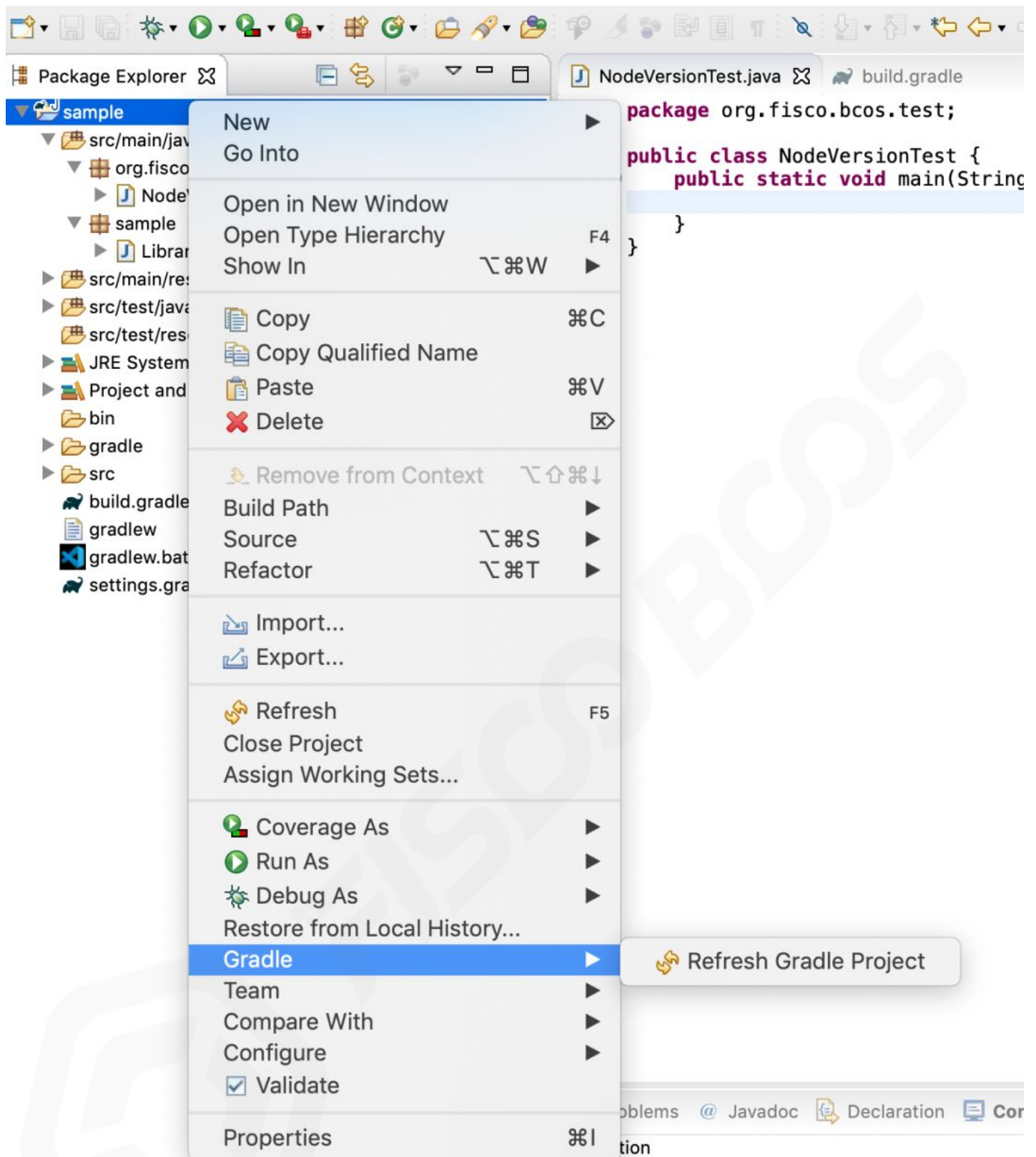
打开build.gradle文件，在repositories添加需要的maven库：

```
1 repositories {
2     // 其他maven库
3     // solidity编译器maven库
4     maven { url "https://dl.bintray.com/ethereum/maven/" }
5     // JavaSDK snapshot版本 maven库
6     maven { url "https://oss.sonatype.org/content/repositories/snapshots/" }
7     // JavaSDK maven库
8     maven { url "http://maven.aliyun.com/nexus/content/groups/public/" }
9     // 通用maven库
10    mavenCentral()
11 }
```

在dependencies添加JavaSDK的依赖版本：

```
1 dependencies {
2     // 其他依赖
3
4     // 日志依赖
5     compile ('org.slf4j:slf4j-log4j12:1.7.25')
6     // JavaSDK依赖, 2.1.0版本
7     compile ("org.fisco-bcos:web3sdk:2.1.0")
8 }
```

然后右键点击项目名：Gradle=> Refresh Gradle Project刷新项目。



注意：

刷新项目可能需要从远程maven库下载项目依赖jar的包，请确保网络畅通，下载过程会需要一些时间。

证书与配置文件

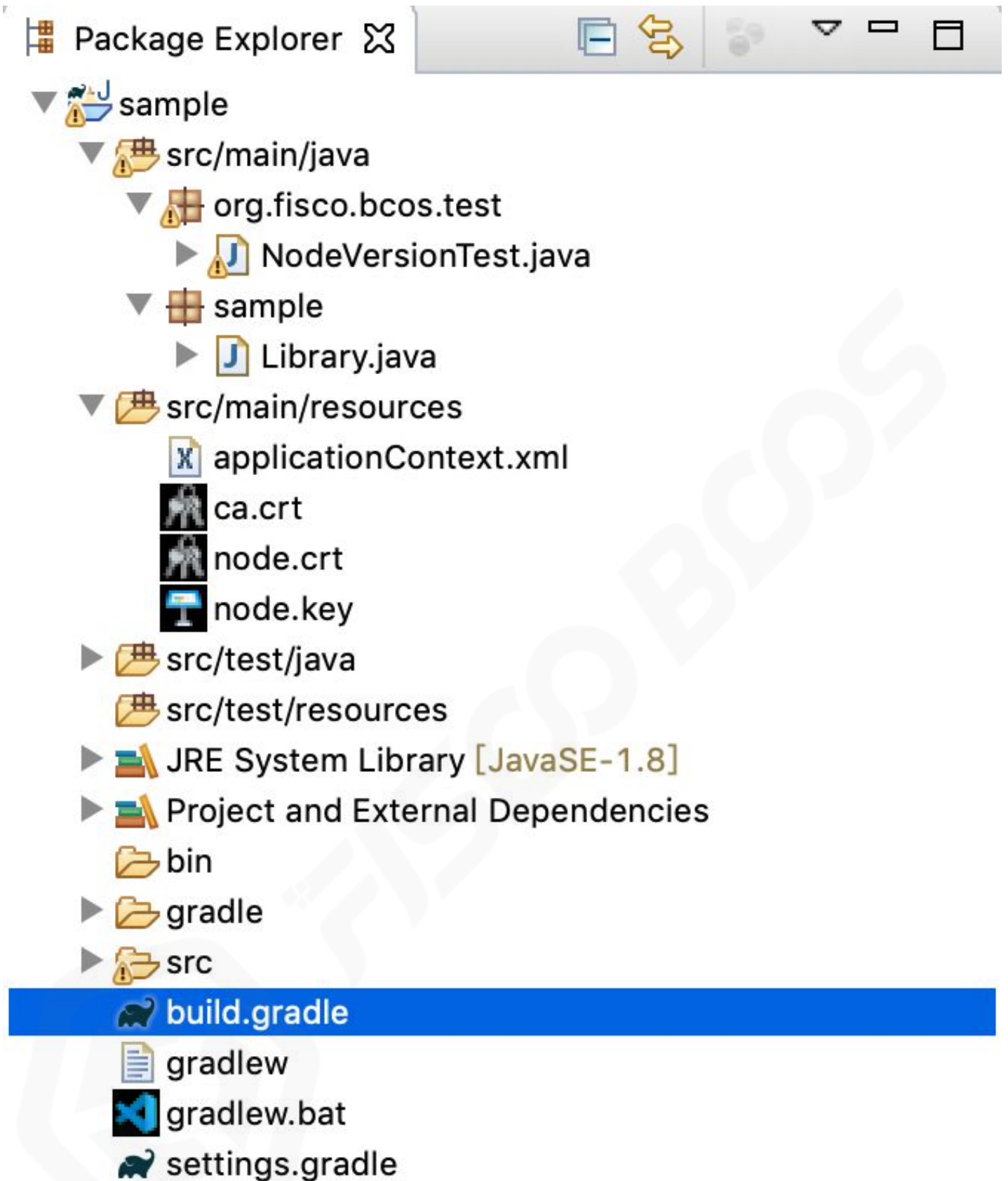
JavaSDK服务运行需要依赖applicationContext.xml配置文件以及证书。配置参考：

<https://fisco-bcos->

[documentation.readthedocs.io/zh_CN/latest/docs/sdk/java_sdk.html#fisco-bcos](https://fisco-bcos-documentation.readthedocs.io/zh_CN/latest/docs/sdk/java_sdk.html#fisco-bcos)

将配置文件 applicationContext.xml，证书文件 ca.crt sdk.crt sdk.key 拷贝到项目 src/main/resources目录下。



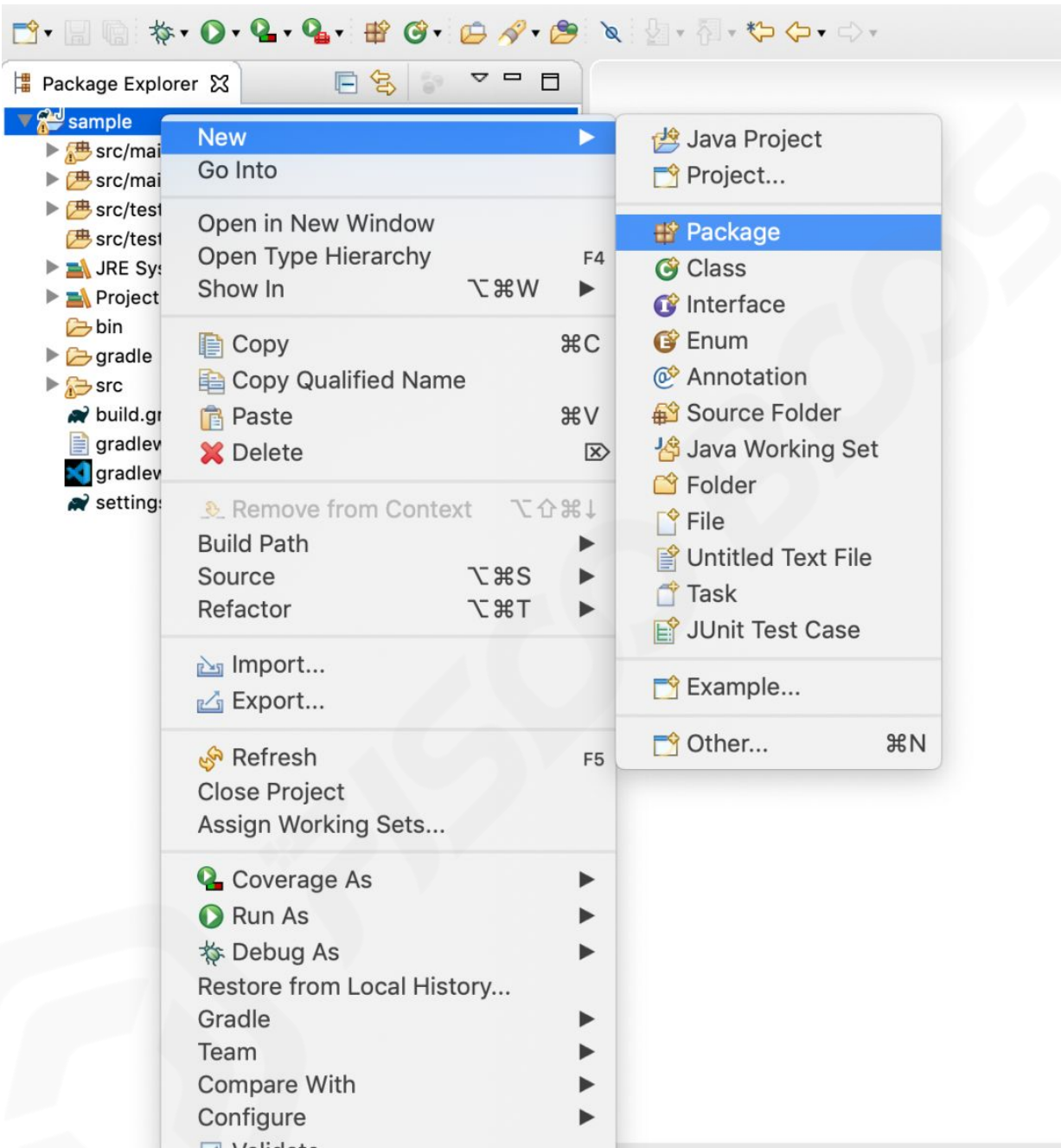


创建运行测试类

到此，我们已经完成了新项目的创建，引入并完成了JavaSDK的配置，现在我们来测试项目的功能。

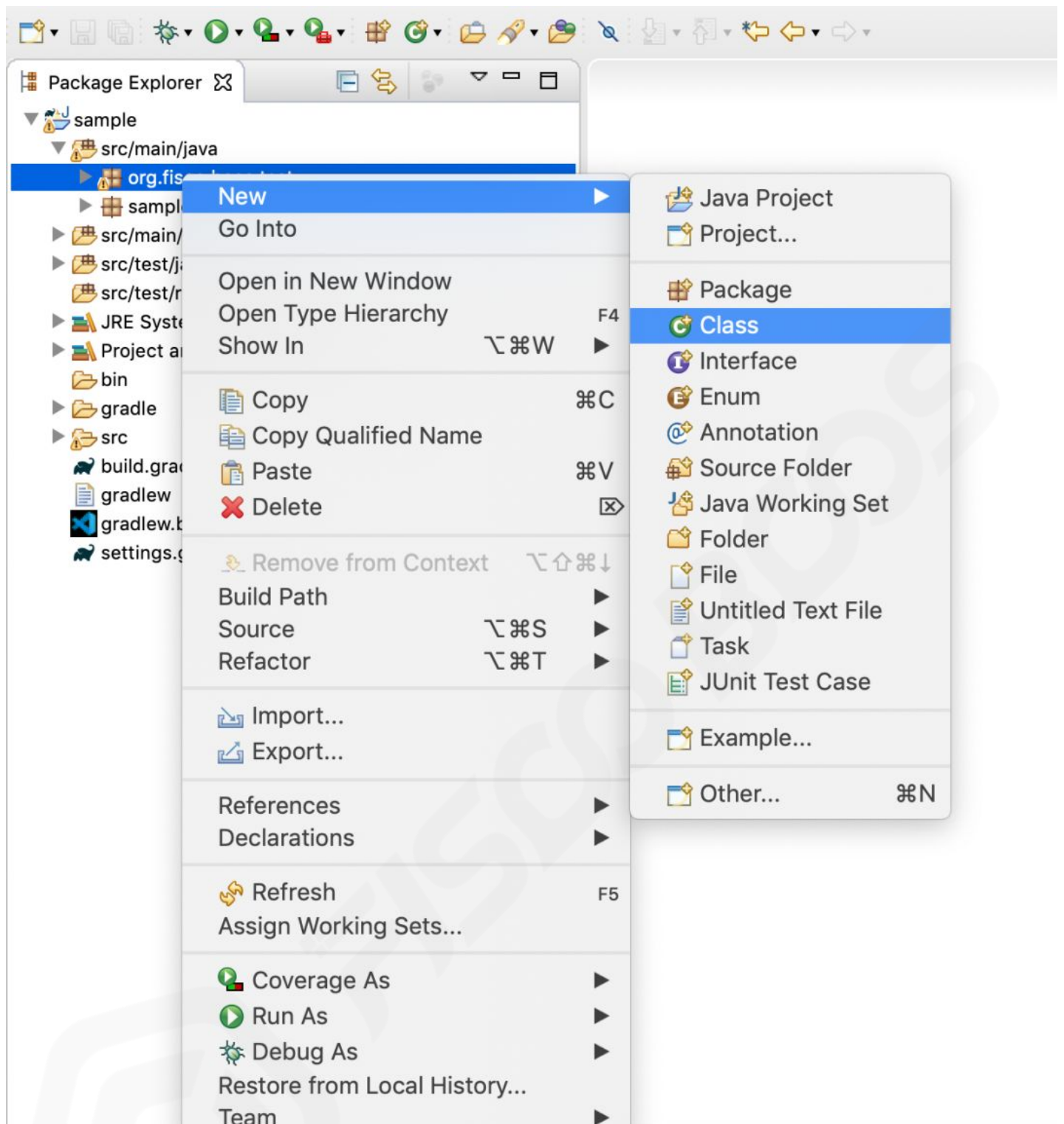
创建类包package

右键点击项目名称选择New => Package然后输入包名，这里使用org.fisco.bcos.test。



创建测试类

右键点击类包名选择 ‘New => Class’。



输入类名，点击Finish，这里类名使用NodeVersionTest。

New Java Class

Java Class

Create a new Java class.



Source folder:

Package:

☐ Enclosing type:

Name:

Modifiers: ☒ public ☐ package ☐ private ☐ protected
☐ abstract ☐ final ☐ static

Superclass:

Interfaces:

Which method stubs would you like to create?

- ☐ public static void main(String[] args)
- ☐ Constructors from superclass
- ☒ Inherited abstract methods

Do you want to add comments? (Configure templates and default value [here](#))

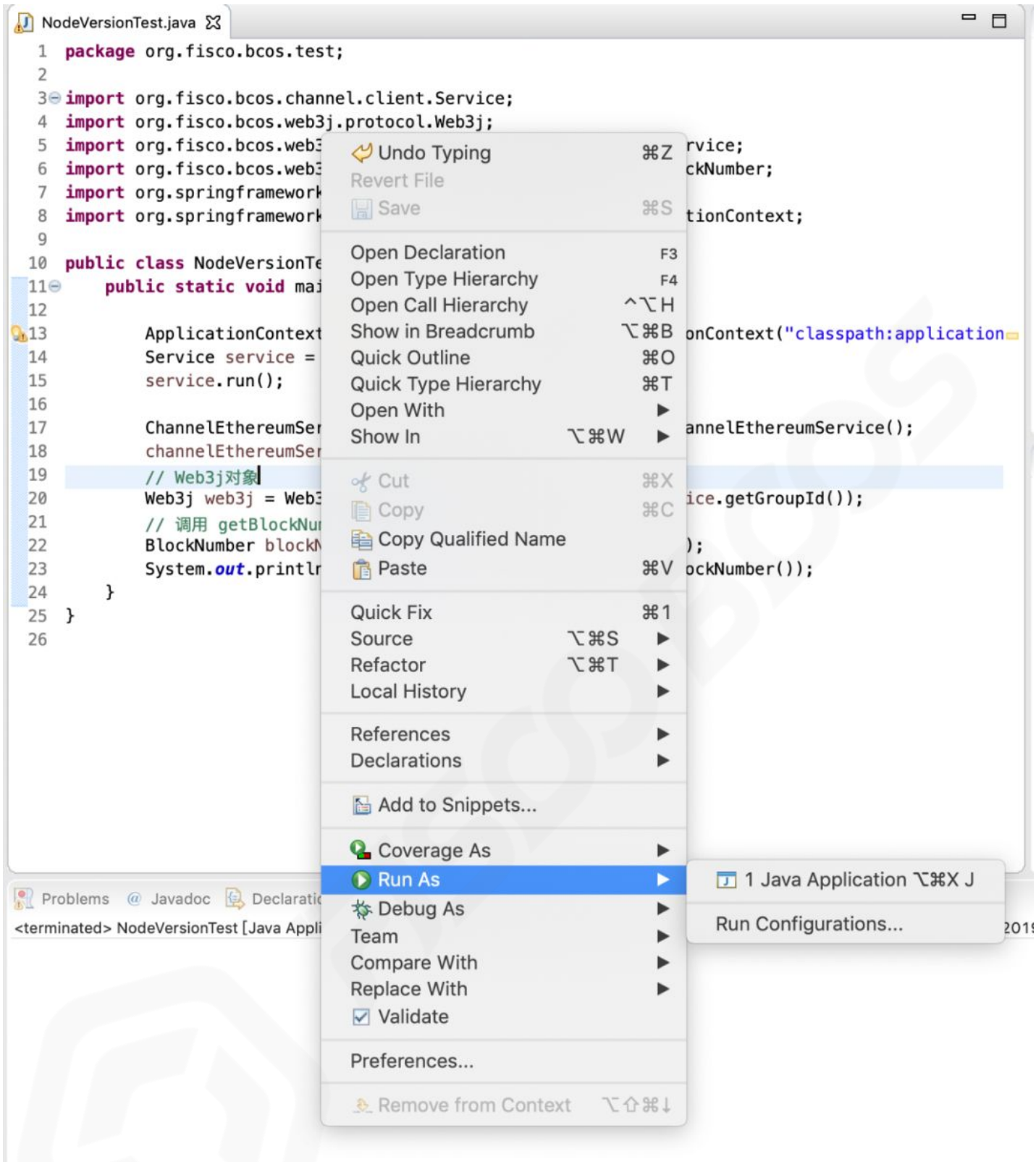
- ☐ Generate comments



测试类的功能为调用getBlockNumber接口获取当前的块高：

```
1 package org.fisco.bcos.test;
2
3 import org.fisco.bcos.channel.client.Service;
4 import org.fisco.bcos.web3j.protocol.Web3j;
5 import org.fisco.bcos.web3j.protocol.channel.ChannelEthereumService;
6 import org.fisco.bcos.web3j.protocol.core.methods.response.BlockNumbe
7 import org.springframework.context.ApplicationContext;
8 import org.springframework.context.support.ClassPathXmlApplicationCon
9
10 public class NodeVersionTest {
11     public static void main(String[] args) throws Exception {
12
13         ApplicationContext context = new ClassPathXmlApplicationContext("cl
14         Service service = context.getBean(Service.class);
15         service.run();
16
17         ChannelEthereumService channelEthereumService = new ChannelEthereum
18         channelEthereumService.setChannelService(service);
19         // Web3j对象
20         Web3j web3j = Web3j.build(channelEthereumService, service.getGroupI
21         // 调用 getBlockNumber 接口获取
22         BlockNumber blockNumber = web3j.getBlockNumber().send();
23         System.out.println("BlockNumber: " + blockNumber.getBlockNumber());
24         System.exit(-1);
25     }
26 }
```

右键点击NodeVersionTest.java文件选择Run As => Java Application运行测试类。



运行结果：

```
log4j:WARN No appenders could be found for logger (org.springframework.core.env.StandardEnvironment).
log4j:WARN Please initialize the log4j system properly.
log4j:WARN See http://logging.apache.org/log4j/1.2/faq.html#noconfig for more info.
BlockNumber: 1157
```

Eclipse 导入项目

//////////

上面流程可以看出创建新项目的过程需要比较多的配置流程，为了便于用户使用，我们提供已有的示例项目asset-app，用户可以将其快速导入Eclipse，在示例基础上快速修改开发自己的应用。

asset-app项目详情请参考：

<https://fisco-bcos->

[documentation.readthedocs.io/zh_CN/latest/docs/tutorial/sdk_application.html](https://fisco-bcos-documentation.readthedocs.io/zh_CN/latest/docs/tutorial/sdk_application.html)

下载asset-app项目工程

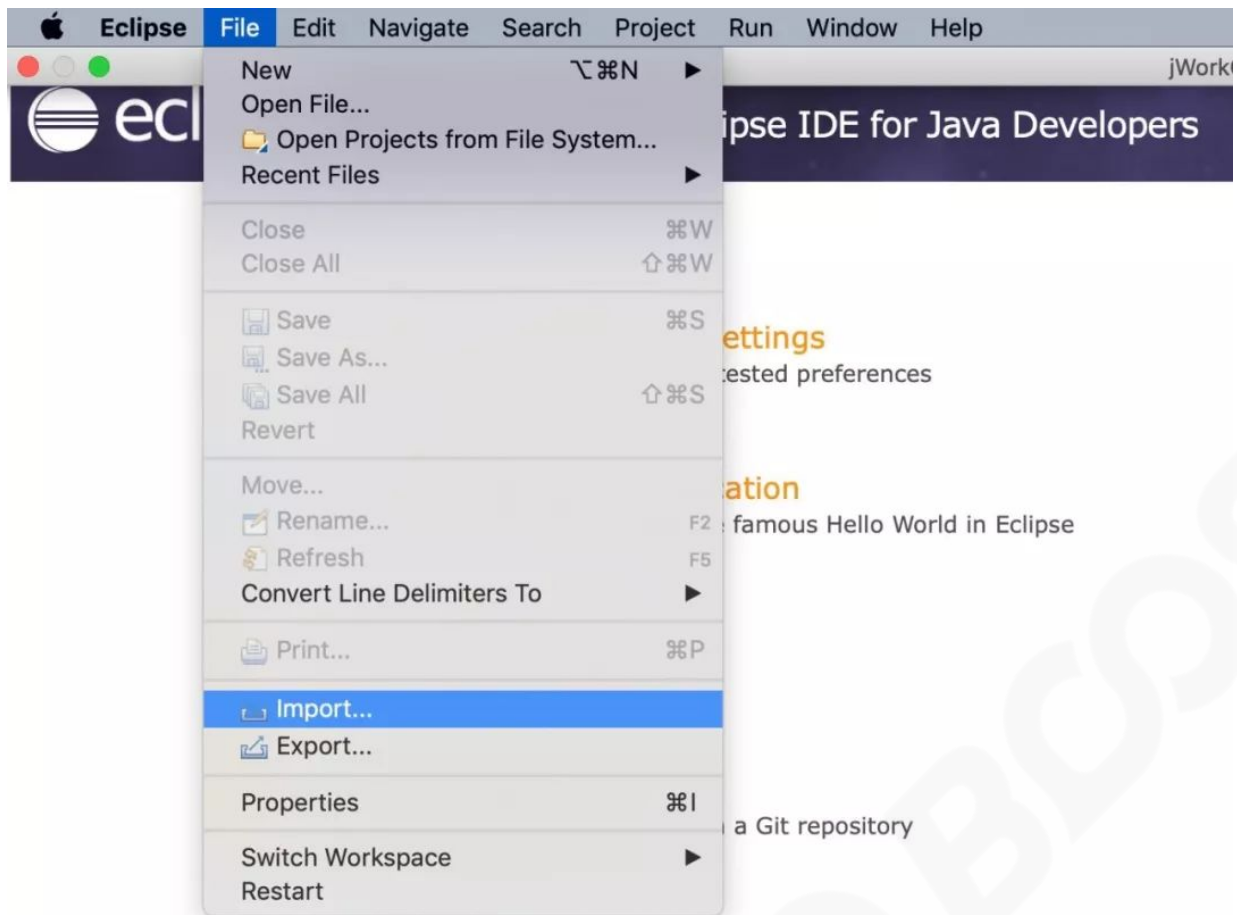
asset-app下载地址：

<https://github.com/FISCO-BCOS/LargeFiles/raw/master/tools/asset-app.tar.gz>

下载之后本地解压至指定路径。

导入工程

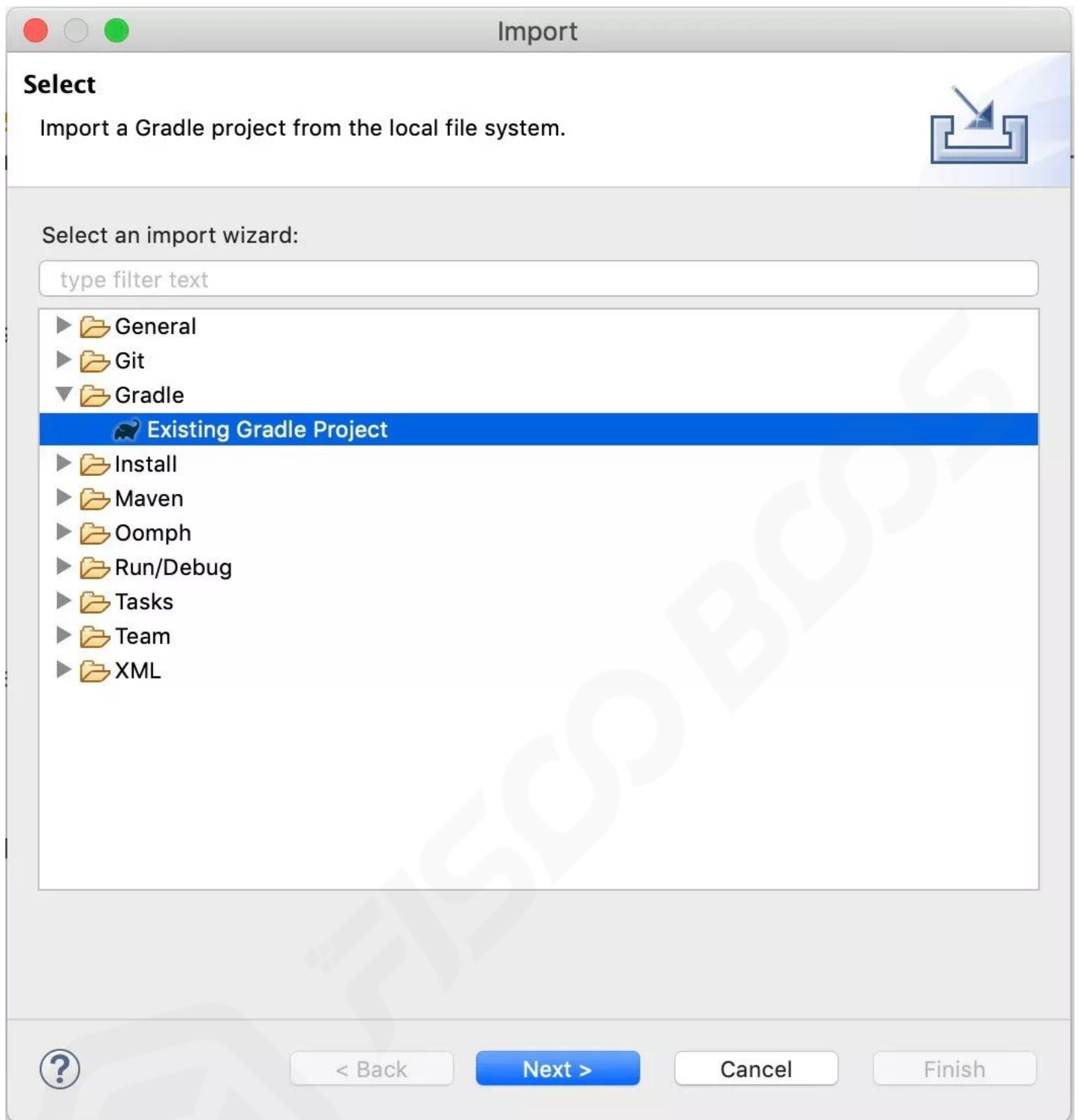
打开Eclipse选择: File => Import => Gradle => Existing Gradle Project。



Import existing projects

Import existing Eclipse projects from the filesystem or archive





点击Next，选择asset-app路径，点击Finish等待项目加载。

Import Gradle Project

Specify the root directory of the Gradle project to import.



Project root directory

[Browse...](#)

Working sets

☐ Add project to working sets

[New...](#)

Working sets

[Select...](#)

Click the Finish button to import the project into the workspace. Click the Next button to select optional import options.



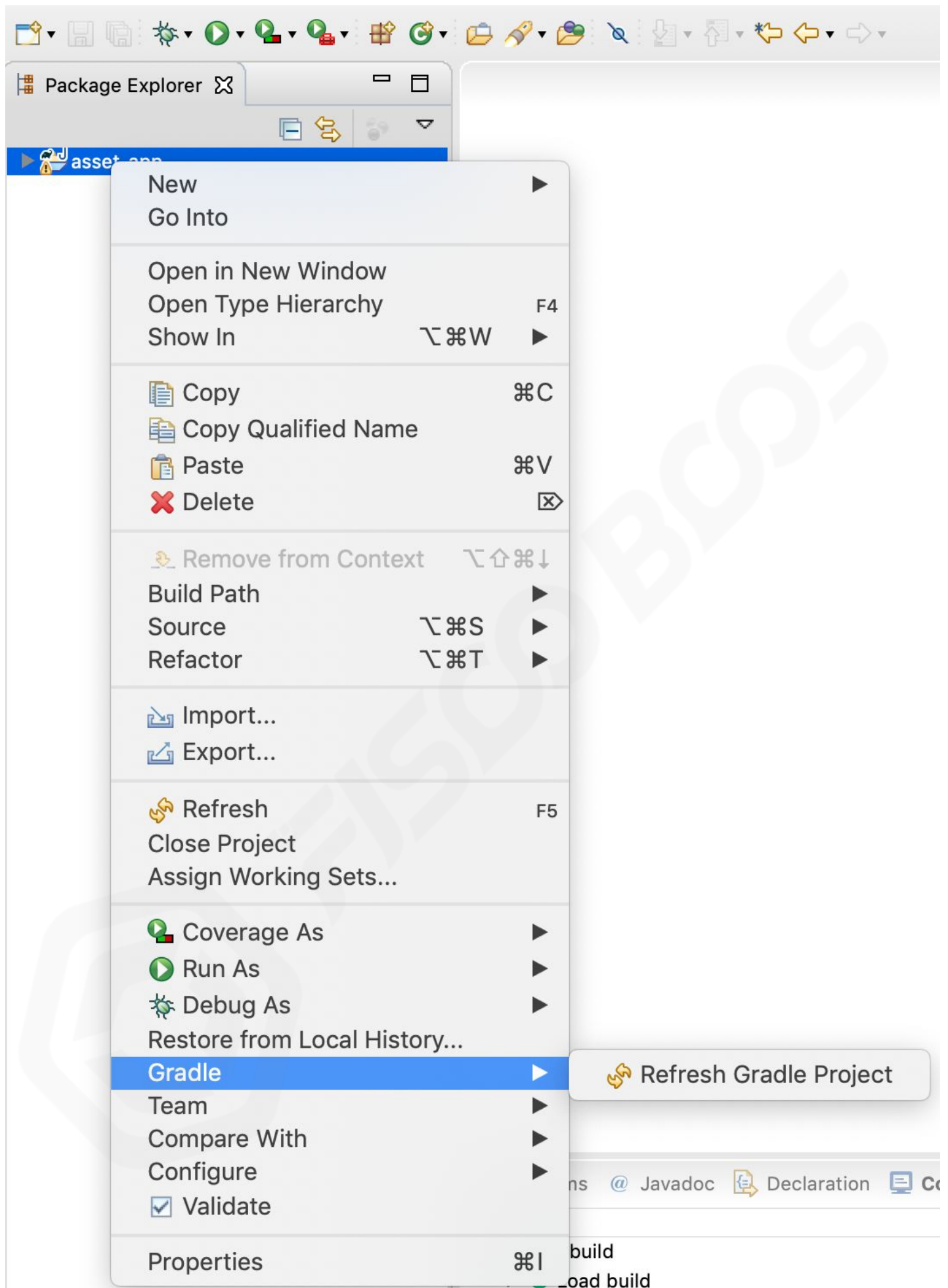
< Back

Next >

Cancel

Finish

项目加载完成之后，右键点击项目名：Gradle=> Refresh Gradle Project刷新项目。



ok! asset-app项目已经被正常加载。

FISCO BCOS的代码完全开源且免费

下载地址↓↓↓

<https://github.com/FISCO-BCOS/FISCO-BCOS>



FISCO BCOS

////////

长按二维码关注

下载最新区块链应用案例

