

**I = Github+Eclipse+SDLC**

## 《学习 Github 和 Eclipse 两系统集成在 SDLC 开发上之简单入门手册（1）》

作者：郭鹏@北京理工大学珠海学院-计算机学院, 用于《软件工程导论》授课

2019-04-06

**前言：**对于缺少项目经验和编程不多的在校的大学生来说，好的工具非常重要，好的系统集成更是重要！这里就是 IDE 和代码仓库工具的集成在软件生命周期（SDLC software development life cycle）上的一个应用，可以帮助学生快速入门！

代码仓库使用和 IDE 集成的重要性在于：

团队之间可以合作，个人历史可以跟踪，有利于大学生创业，进入工业界的状态。

手册的特点：

整个手册，是以创业者“从零开始”为出发点，新开 github 账户，在 elipse 建立 projects，上传 projects...

### 1. 目的：Github 的项目名称，也是仓库的代号：**github-eclipse-repo**

**仓库的描述：**2019 年 4 月课堂《软件工程导论》实训（这是笔者的项目名称，读者自选描述）

- 1) 最新 Eclipse IDE for Enterprise Java Development
- 2) Github
- 3) Software Development Life Cycle

可以开始软件创新创业，可以开始软件的毕业设计，可以自己平时学习软件编程

# 目录

[本地目录结构](#)

[Eclipse 的 JDK 相对目录配置](#)

[Github 开户](#)

[建立你的 Github 账户的一个仓库 repository](#)

[一个空的 github 代码仓库就是这个样子](#)

[SDLC 不同角色的工作空间和本地仓库的目录](#)

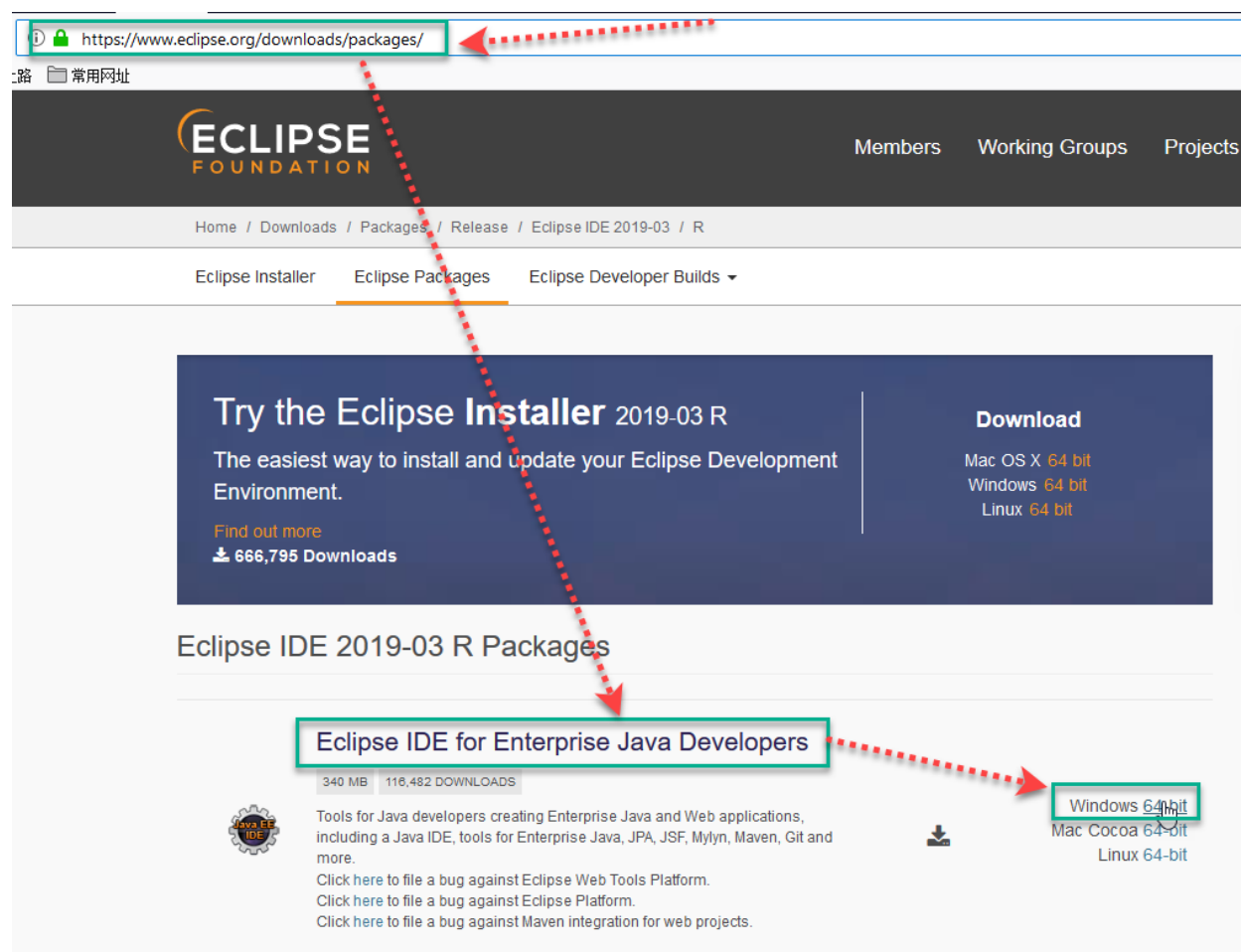
[团队开始建立和命名仓库中的 project](#)

1. [运维人员的工作](#) : 在 adm-A 的 Eclipse 的 Workspace 中建立一般 Eclipse project

## 2. 准备本地手提电脑（或台式电脑）的 IDE Eclipse 环境：

步骤 1: 下载最新的 Eclipse for J2EE：一个 zip file。

<https://www.eclipse.org/downloads/packages/>



## 步骤 2: JDK1.8 下载 :

<https://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>

🔒 <https://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>

新手上路 常用网址

Linux ARM 32 Hard Float ABI	9.05 MB	<a href="#">jdk-8u201-linux-arm32-vfp-hflt-demos.tar.gz</a>
Linux ARM 64 Hard Float ABI	9.06 MB	<a href="#">jdk-8u201-linux-arm64-vfp-hflt-demos.tar.gz</a>
Linux x86	56.13 MB	<a href="#">jdk-8u201-linux-i586-demos.rpm</a>
Linux x86	55.98 MB	<a href="#">jdk-8u201-linux-i586-demos.tar.gz</a>
Linux x64	56.23 MB	<a href="#">jdk-8u201-linux-x64-demos.rpm</a>
Linux x64	56.08 MB	<a href="#">jdk-8u201-linux-x64-demos.tar.gz</a>
Mac OS X	56.25 MB	<a href="#">jdk-8u201-macosx-x86_64-demos.zip</a>
Solaris SPARC 64-bit	12.2 MB	<a href="#">jdk-8u201-solaris-sparcv9-demos.tar.Z</a>
Solaris SPARC 64-bit	8.46 MB	<a href="#">jdk-8u201-solaris-sparcv9-demos.tar.gz</a>
Solaris x64	12.19 MB	<a href="#">jdk-8u201-solaris-x64-demos.tar.Z</a>
Solaris x64	8.42 MB	<a href="#">jdk-8u201-solaris-x64-demos.tar.gz</a>
Windows x86	56.96 MB	<a href="#">jdk-8u201-windows-i586-demos.zip</a>
Windows x64	56.98 MB	<a href="#">jdk-8u201-windows-x64-demos.zip</a>

**Java SE Development Kit 8u202**

You must accept the [Oracle Binary Code License Agreement for Java SE](#) to download this software.

☐ Accept License Agreement ☒ Decline License Agreement

Product / File Description	File Size	Download
Linux ARM 32 Hard Float ABI	72.86 MB	<a href="#">jdk-8u202-linux-arm32-vfp-hflt.tar.gz</a>
Linux ARM 64 Hard Float ABI	69.75 MB	<a href="#">jdk-8u202-linux-arm64-vfp-hflt.tar.gz</a>
Linux x86	173.08 MB	<a href="#">jdk-8u202-linux-i586.rpm</a>
Linux x86	187.9 MB	<a href="#">jdk-8u202-linux-i586.tar.gz</a>
Linux x64	170.15 MB	<a href="#">jdk-8u202-linux-x64.rpm</a>
Linux x64	185.05 MB	<a href="#">jdk-8u202-linux-x64.tar.gz</a>
Mac OS X x64	249.15 MB	<a href="#">jdk-8u202-macosx-x64.dmg</a>
Solaris SPARC 64-bit (SVR4 package)	125.09 MB	<a href="#">jdk-8u202-solaris-sparcv9.tar.Z</a>
Solaris SPARC 64-bit	88.1 MB	<a href="#">jdk-8u202-solaris-sparcv9.tar.gz</a>
Solaris x64 (SVR4 package)	124.37 MB	<a href="#">jdk-8u202-solaris-x64.tar.Z</a>
Solaris x64	85.38 MB	<a href="#">jdk-8u202-solaris-x64.tar.gz</a>
Windows x86	201.64 MB	<a href="#">jdk-8u202-windows-i586.exe</a>
Windows x64	211.58 MB	<a href="#">jdk-8u202-windows-x64.exe</a>

点击下载：

https://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html

Java SE Development Kit 8u202 Demos and Samples Downloads

You must accept the [Oracle BSD License](#), to download this software.

☐ Accept License Agreement ☒ Decline License Agreement

Product / File Description	File Size	Download
Linux ARM 32 Hard Float ABI	9.05 MB	<a href="#">jdk-8u201-linux-arm32-vfp-hflt-demos.tar.gz</a>
Linux ARM 64 Hard Float ABI	9.06 MB	<a href="#">jdk-8u201-linux-arm64-vfp-hflt-demos.tar.gz</a>
Linux x86	56.13 MB	<a href="#">jdk-8u201-linux-i586-demos.rpm</a>
Linux x86	55.98 MB	<a href="#">jdk-8u201-linux-i586-demos.tar.gz</a>
Linux x64	56.23 MB	<a href="#">jdk-8u201-linux-x64-demos.rpm</a>
Linux x64	56.08 MB	<a href="#">jdk-8u201-linux-x64-demos.tar.gz</a>
Mac OS X	56.25 MB	<a href="#">jdk-8u201-macosx-x86_64-demos.zip</a>
Solaris SPARC 64-bit	12.2 MB	<a href="#">jdk-8u201-solaris-sparcv9-demos.tar.Z</a>

正在打开 jdk-8u202-windows-x64.exe

您选择了打开：

☒ jdk-8u202-windows-x64.exe

文件类型： Binary File (212 MB)

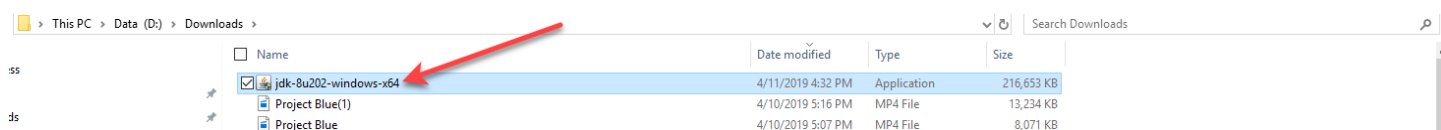
来源： https://download.oracle.com

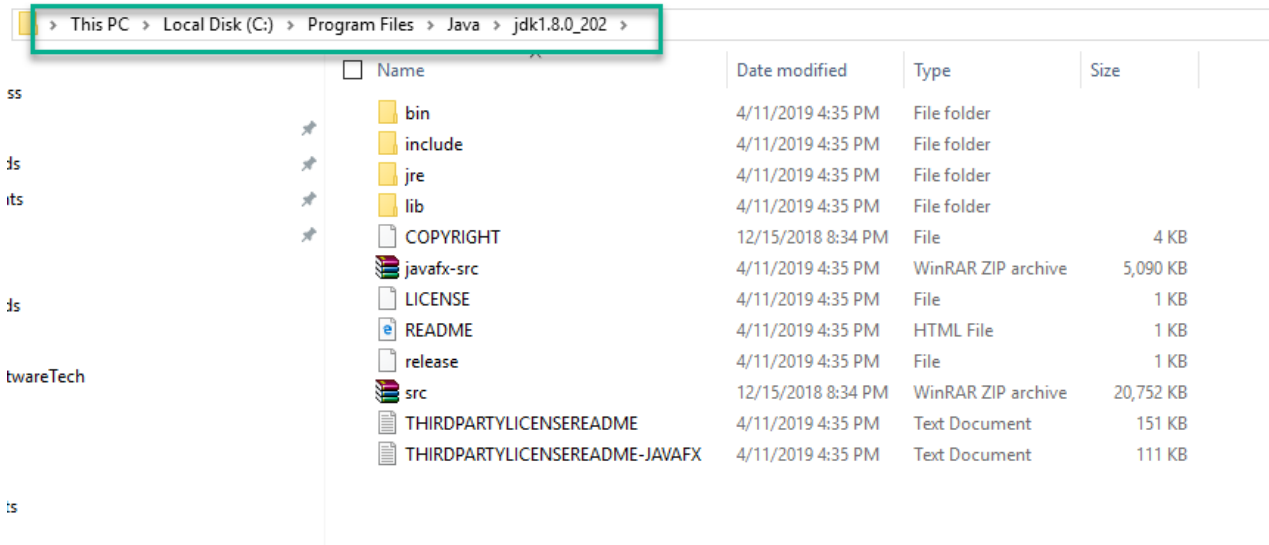
您想要保存此文件吗？

保存文件 取消

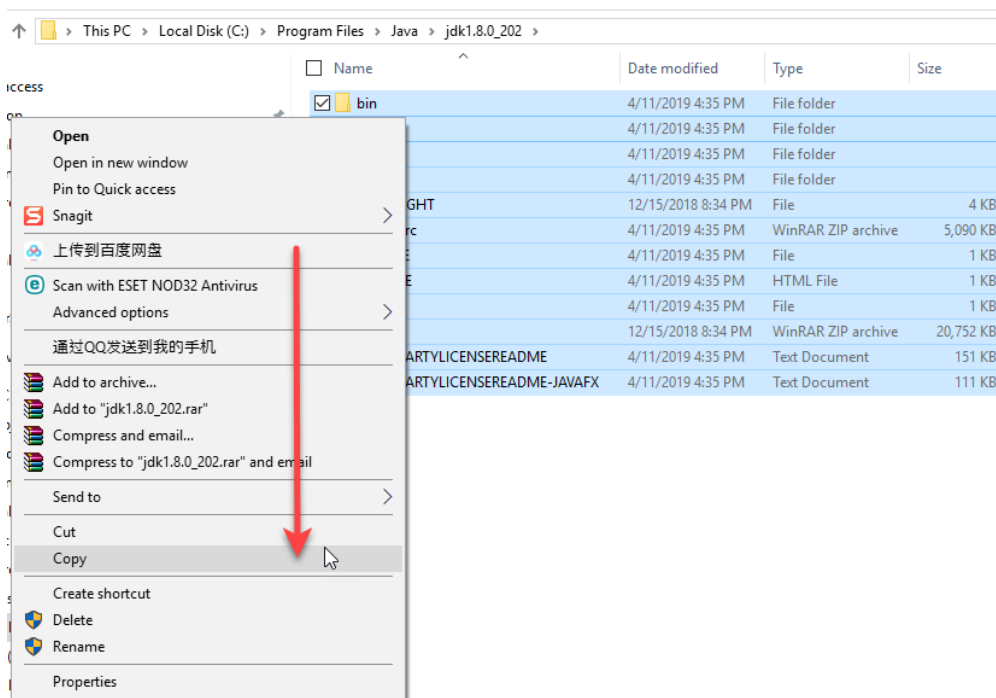
Product / File Description	File Size	Download
Linux ARM 32 Hard Float ABI	72.86 MB	<a href="#">jdk-8u202-linux-arm32-vfp-hflt.tar.gz</a>
Linux ARM 64 Hard Float ABI	69.75 MB	<a href="#">jdk-8u202-linux-arm64-vfp-hflt.tar.gz</a>
Linux x86	173.08 MB	<a href="#">jdk-8u202-linux-i586.rpm</a>
Linux x86	187.9 MB	<a href="#">jdk-8u202-linux-i586.tar.gz</a>
Linux x64	170.15 MB	<a href="#">jdk-8u202-linux-x64.rpm</a>
Linux x64	185.05 MB	<a href="#">jdk-8u202-linux-x64.tar.gz</a>
Mac OS X x64	249.15 MB	<a href="#">jdk-8u202-macosx-x64.dmg</a>
Solaris SPARC 64-bit (SVR4 package)	125.09 MB	<a href="#">jdk-8u202-solaris-sparcv9.tar.Z</a>
Solaris SPARC 64-bit	88.1 MB	<a href="#">jdk-8u202-solaris-sparcv9.tar.gz</a>
Solaris x64 (SVR4 package)	124.37 MB	<a href="#">jdk-8u202-solaris-x64.tar.Z</a>
Solaris x64	85.38 MB	<a href="#">jdk-8u202-solaris-x64.tar.gz</a>
Windows x86	201.64 MB	<a href="#">jdk-8u202-windows-i586.exe</a>
Windows x64	211.58 MB	<a href="#">jdk-8u202-windows-x64.exe</a>

下载后找到文件 D:\Downloads （这是笔者的目录，读者可能不同），点击解开在某一目录下 C:\Program Files\Java （这是笔者的目录，读者可能不同）

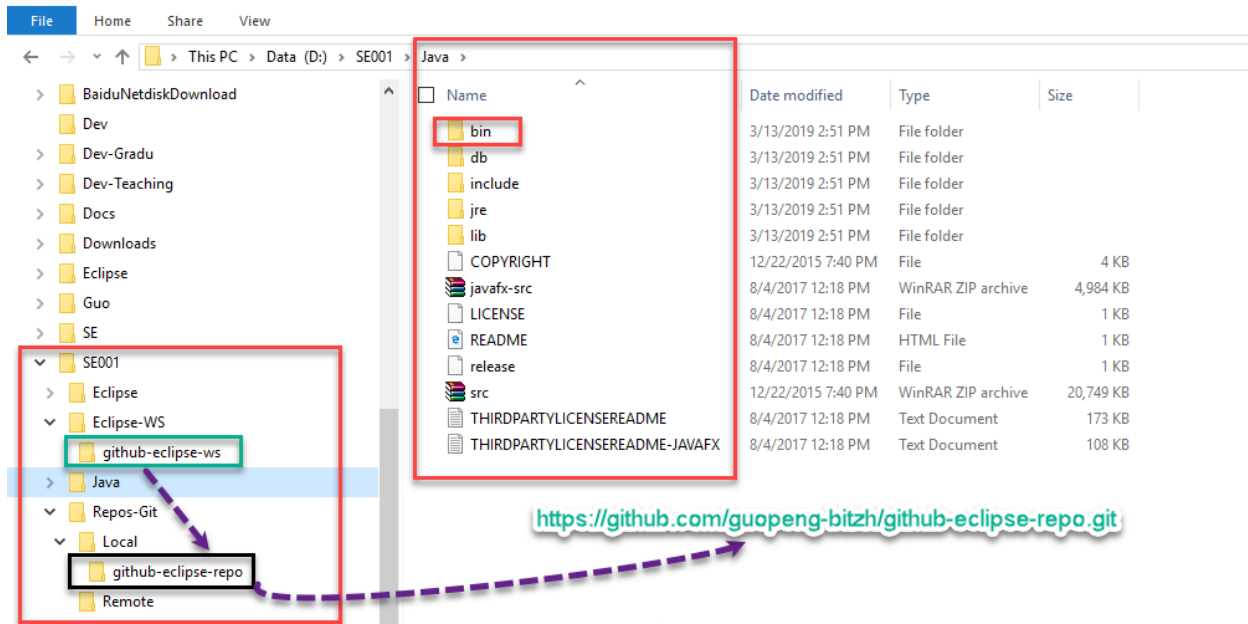




准备拷贝整个目录如下所示到 Eclipse 中去。 笔者的 Eclipse Java 目录是 D:\SE001\Java

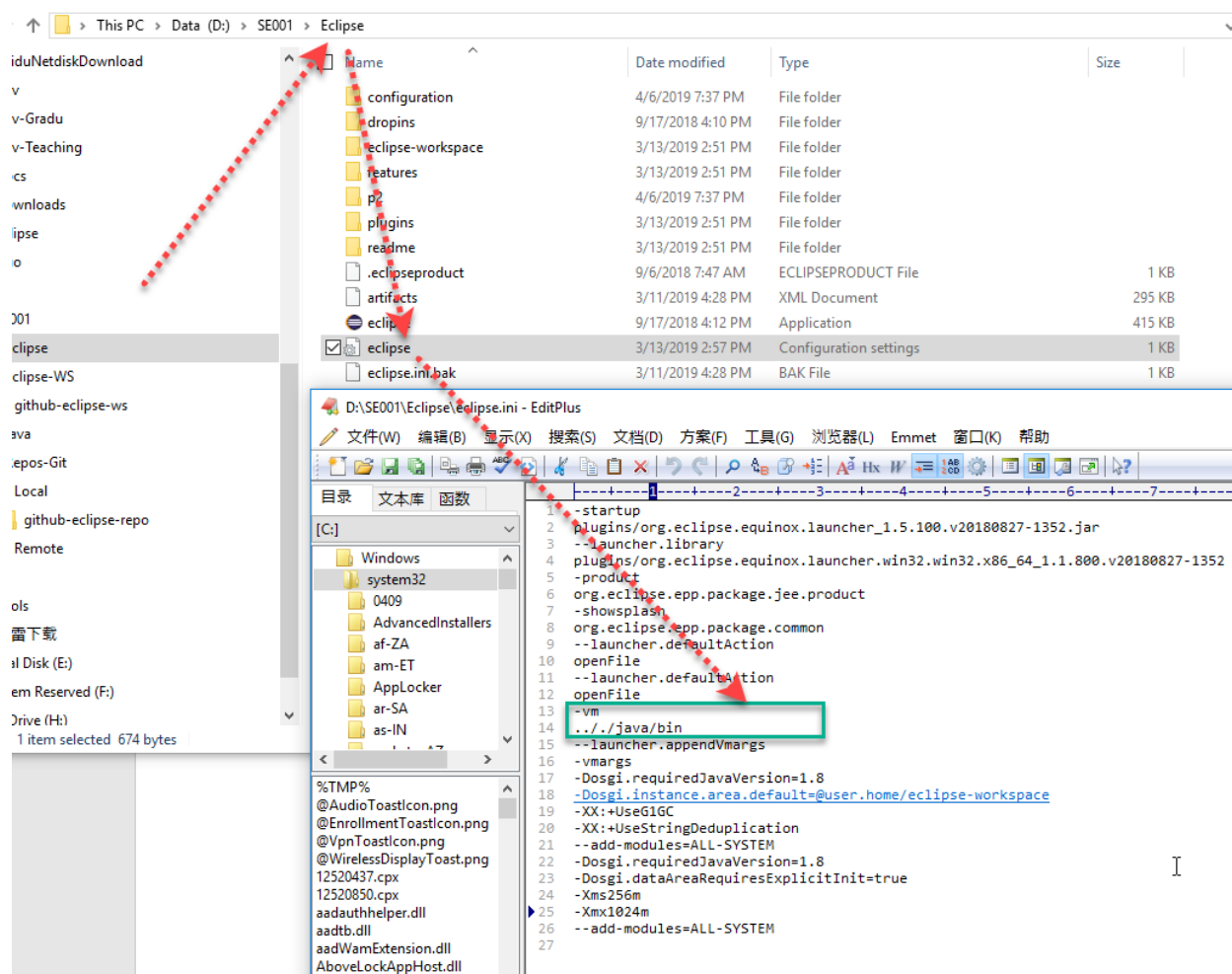


步骤 3: 在自己的本地硬盘（任何分区）建立如下目录结构，例如笔者建在 D 盘的 SE001 目录（注释：软件工程 SoftwareEngineering001=SE001），并将 eclipse 和 JDK 1.8 解压和命名到相应的目录 Eclipse 和 Java:



步骤 4: 在 eclipse 配置文件中, 稍微改动, 加入以下绿色边框部分, 目的是使自己的 eclipse 的运行依赖于相对目录下的 JDK, 最重要的是方便未来拷贝。

建议用 EditPlus.exe (建议放在 SE001\Tools 下面)



在打开 eclipse.ini 如上, 加入两行 :

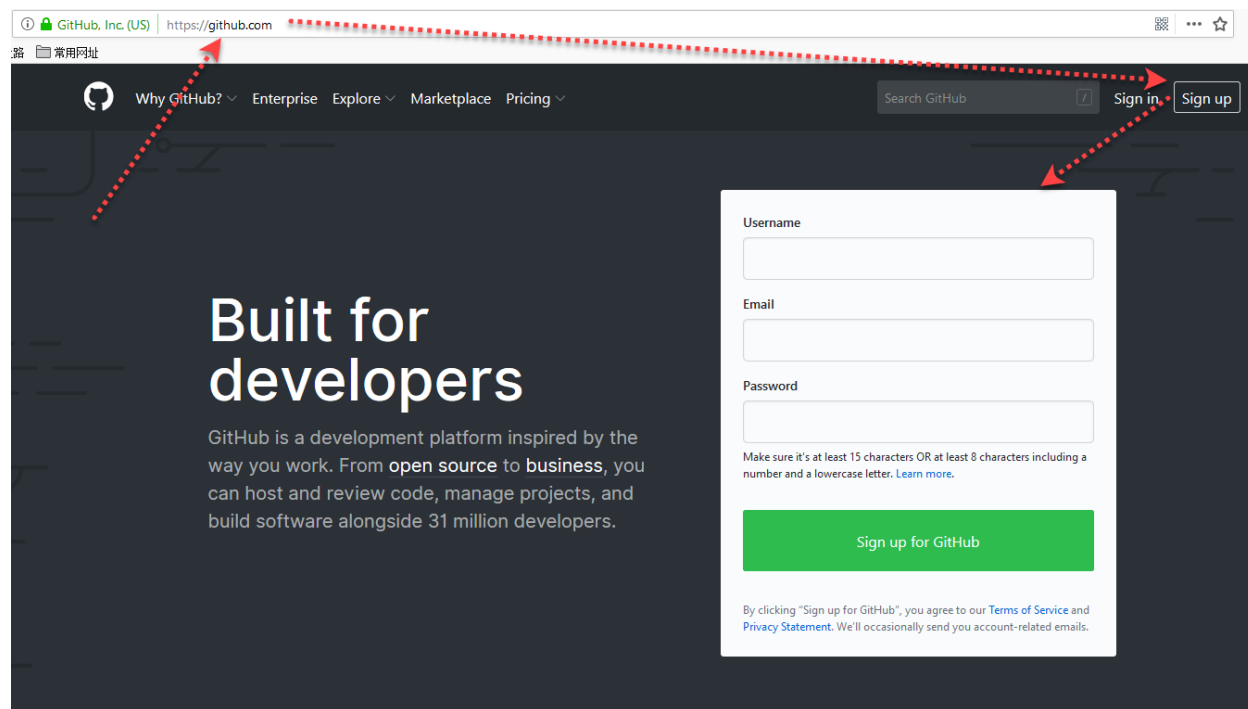
-vm (回车换行)

../../Java/bin (回车换行)



### 3. 到 Github 网站: <https://github.com/> 去开设账户, 创建第一个代码仓库: `github-eclipse-repo`

步骤 1: 开设新账户, 需要一个电子邮箱验证



## 步骤 2: 建立账户

GitHub, Inc. (US) <https://github.com/join?source=header-home>

Why GitHub? Enterprise Explore Marketplace Pricing

Search GitHub Sign in Sign up

### Join GitHub

The best way to design, build, and ship software.

Step 1:  
Set up your account

Step 2:  
Choose your subscription

Step 3:  
Tailor your experience

#### Create your personal account

**Username \***

guopeng-bitzh ✓

This will be your username. You can add the name of your organization later.

**Email address \***

guopeng5432@dingtalk.com ✓

We'll occasionally send updates about your account to this inbox. We'll never share your email address with anyone.

**Password \***

••••••••

Make sure it's at least 15 characters OR at least 8 characters including a number and a lowercase letter. [Learn more.](#)

**Verify account**

请解决下方的问题，  
以便让我们知道您是真人而非机器

#### You'll love GitHub

- Unlimited public repositories
- Unlimited private repositories
- ✓ Limitless collaboration
- ✓ Frictionless development
- ✓ Open source community

### 步骤 3: 选择免费账户类型 :

GitHub, Inc. (US)

https://github.com/join/plan

路 常用网址

With tools developers love and the world's largest open source community, there's no wrong choice.

✓

Free

The basics of GitHub for every developer

\$0

per month

**Includes:**

- ∞ Unlimited public and private repositories
- ✓ 3 collaborators for private repositories
- ✓ Issues and bug tracking
- ✓ Project management

Are you a [student](#)? Get access to the best developer tools for free with the [GitHub Student Developer Pack](#).

Pro

Pro tools for developers with advanced requirements

\$7

per month

[\(view in CNY\)](#)

**Includes:**

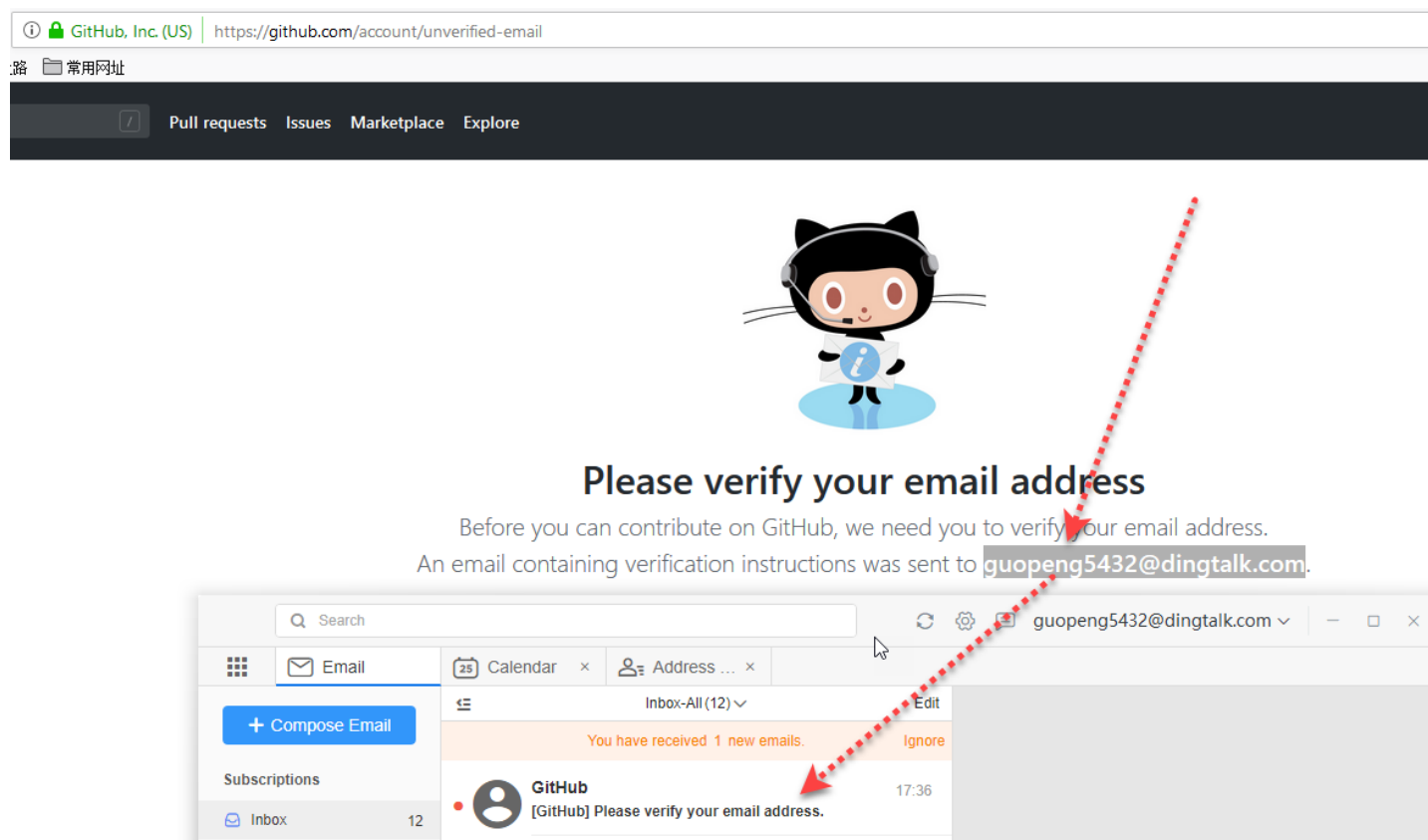
- ∞ Unlimited public and private repositories
- ∞ Unlimited collaborators
- ✓ Issues and bug tracking
- ✓ Project management
- ✓ [Advanced tools and insights](#)

☐ **Help me set up an organization next**  
Organizations are separate from personal accounts and are best suited for businesses who need to manage permissions for many employees. [Learn more about organizations](#)

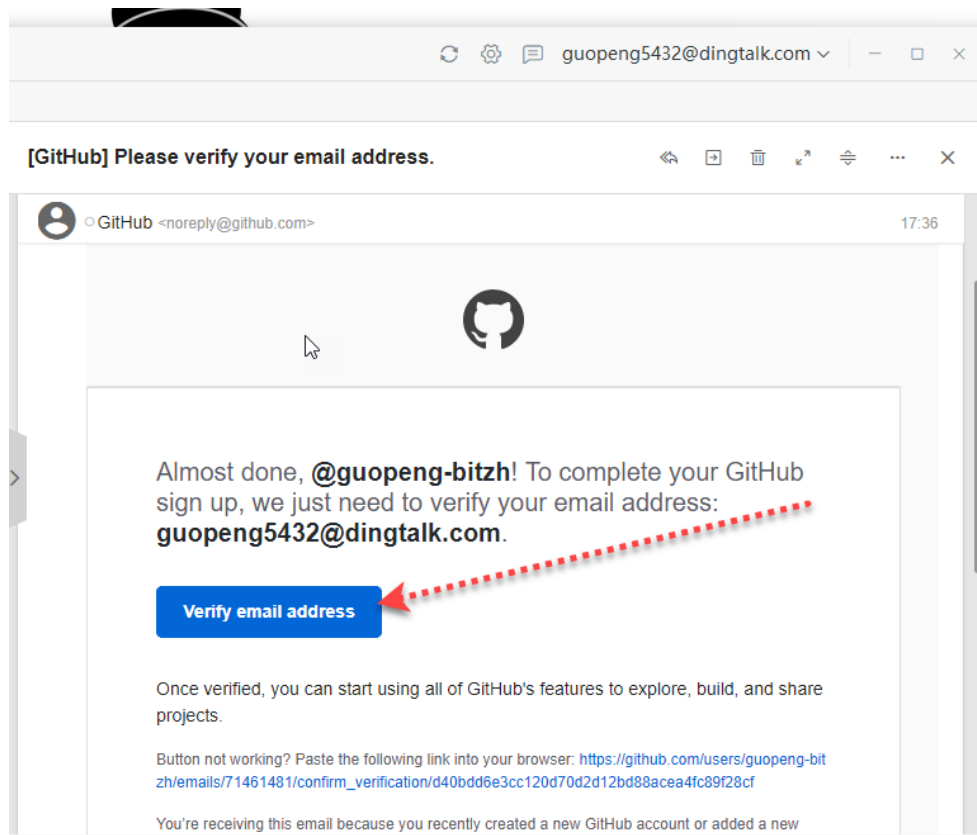
☐ **Send me updates on GitHub news, offers, and events**  
Unsubscribe anytime in your email preferences. [Learn more](#)

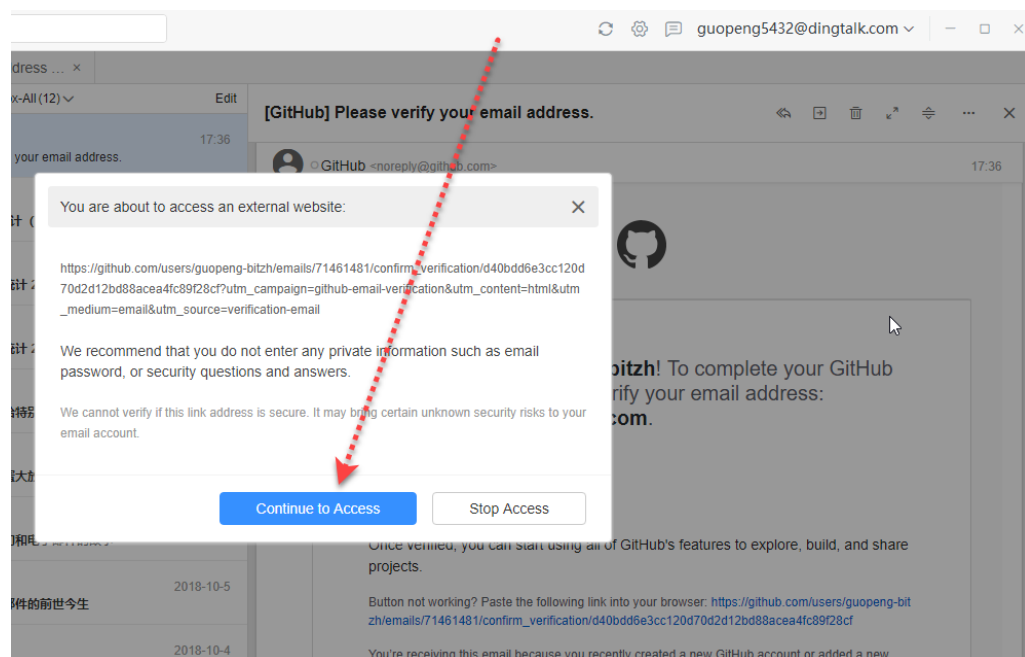
Continue

步骤 4: 去自己的电子邮箱寻找 Github 发来的验证邮件 :

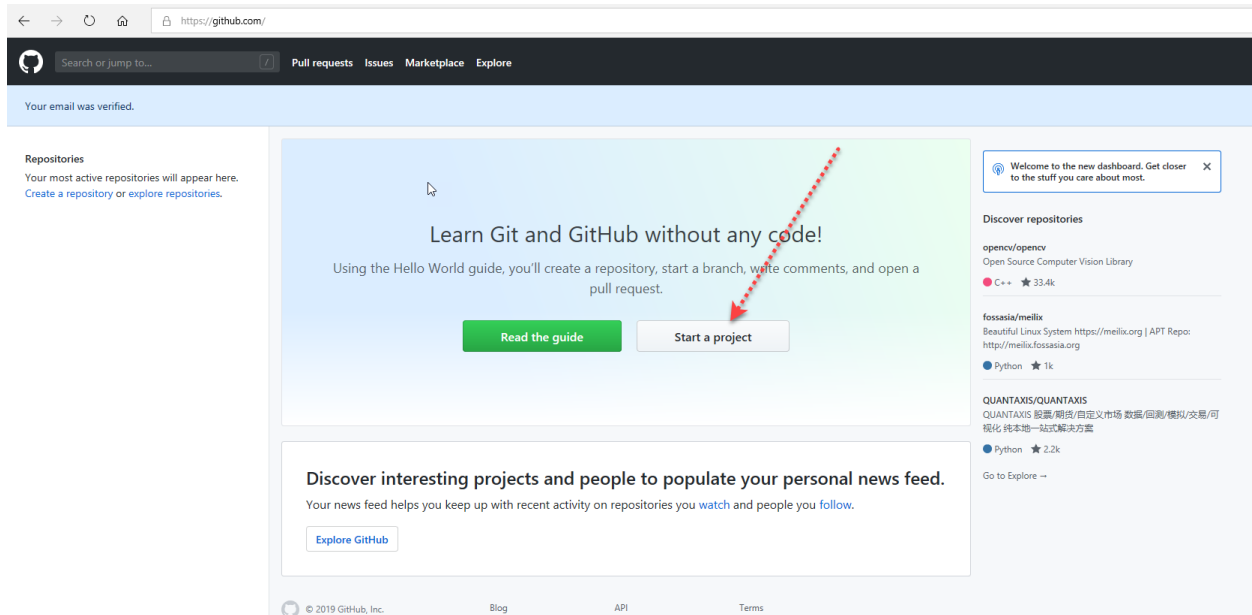


步骤 5: 点开验证连接, 成功后可以用账号和密码再次在 Github 登录:





步骤 6: 建立一个 Github 的 Project (整个大的工程, 其实就是一个仓库, 存放这个工程下的所有项目, 每个项目最后用一个或几个目录代表, 注意每个具体的项目在 eclipse 则是一个 project, 这些概念比较容易引起混淆)



步骤 7: 点击 “Start a project” 后进入一下，建立你的 Github 账户的一个仓库 repository

The screenshot shows the GitHub 'Create a new repository' page. Red dotted arrows indicate the following steps:

- Clicking the 'Start a project' button in the top navigation bar.
- Clicking the 'Repository name' input field.
- Clicking the 'Description (optional)' input field.
- Clicking the 'Public' radio button.
- Clicking the 'Initialize this repository with a README' checkbox.
- Clicking the 'Add .gitignore: Ada' dropdown menu.
- Clicking the 'Add a license: None' dropdown menu.
- Clicking the 'Create repository' button.

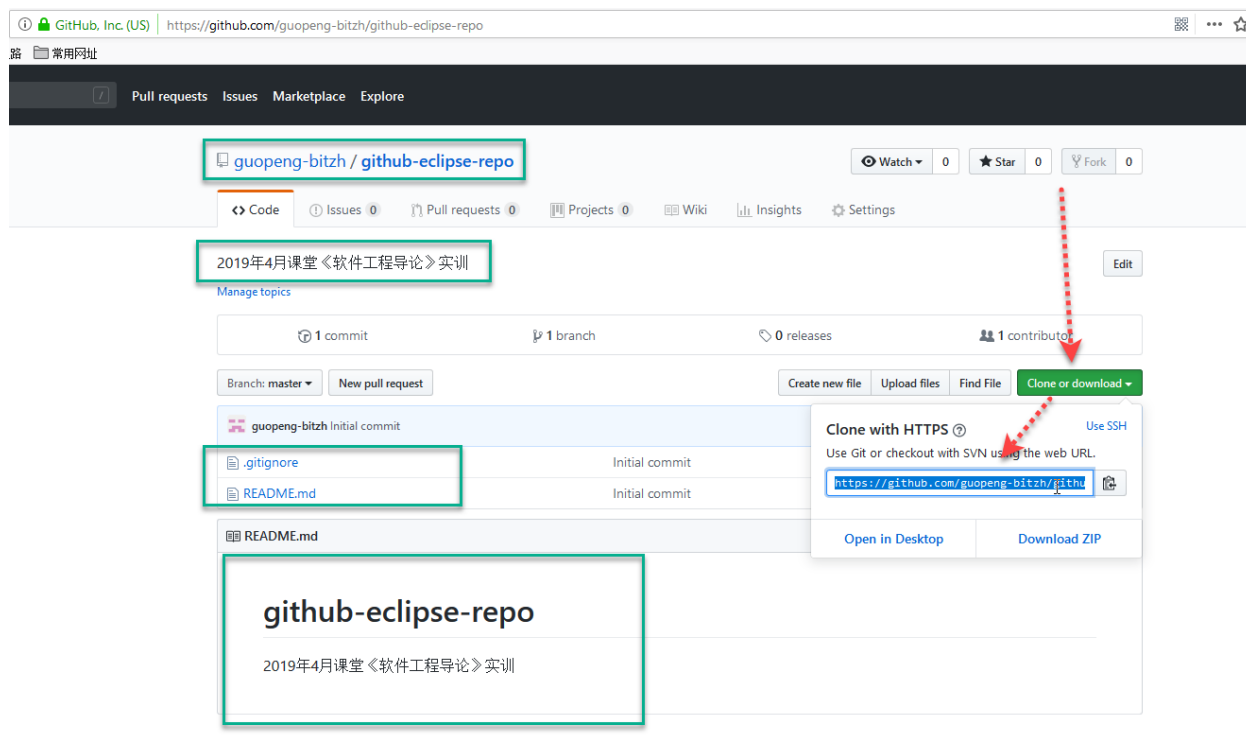
The form fields are filled with:

- Owner: guopeng-bitzh
- Repository name: github-eclipse-repo
- Description: 2019年4月课堂《软件工程导论》实训
- Visibility: Public
- Initialize with README: ☒
- .gitignore: Ada
- License: None



步骤 8: 一个空的 github 代码仓库就是这个样子, 可以得到代码仓库的地址如下:

`https://github.com/guopeng-bitzh/github-eclipse-repo.git`



4. 以运行维护人员 administrator 的角色在 eclipse 开设一个软件项目的各个 project，然后上传到 Github 的仓库（代号： **github-eclipse-repo**），这里以软件开发生命周期的 6 个阶段建立 eclipse project，在第 4 阶段的“程序编写”中，笔者建立四种不同类型的 eclipse project，1）一般 Project，2）一般 Java Project，3）Static project 和 4）Java Web Project，读者如需增加，可以自行参考笔者的 project 命名规则根据自己的需要而增添删。

要求：

1. eclipse work space 工作空间使用的目录在 SE001\Eclipse-WS\github-eclipse-ws\users 之下的各个不同角色：

dev-x ： 是程序员 developer x 的 workspace （SDLC 第 4th 阶段）

adm-x ： 是运维人员 administrator x 的 workspace （SDLC 第 6th 阶段）

test-x ： 是测试员 tester （或工业界称 QA，Quality Analysist 质量分析员）x 的 workspace。

2. github-eclipse-ws 中的 projects 连接 Git 时，本地的代码仓库是在使用目录在 SE001\Eclipse-WS\github-eclipse-repo\users 之下的各个不同角色：

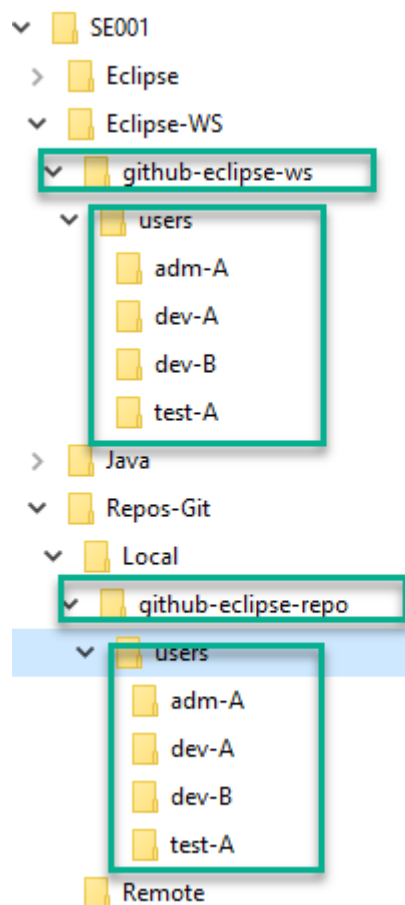
dev-x ： 是程序员 developer x 的本地代码仓库 repository （SDLC 第 4th 阶段）

adm-x ： 是运维人员 administrator x 的本地代码仓库 repository （SDLC 第 6th 阶段）

test-x ： 是测试员 tester （或工业界称 QA，Quality Analysist 质量分析员）x 的本地代码仓库 repository

3. 8 个 projects 命名如下：这 8 个 eclipse projects 组成一个工程。

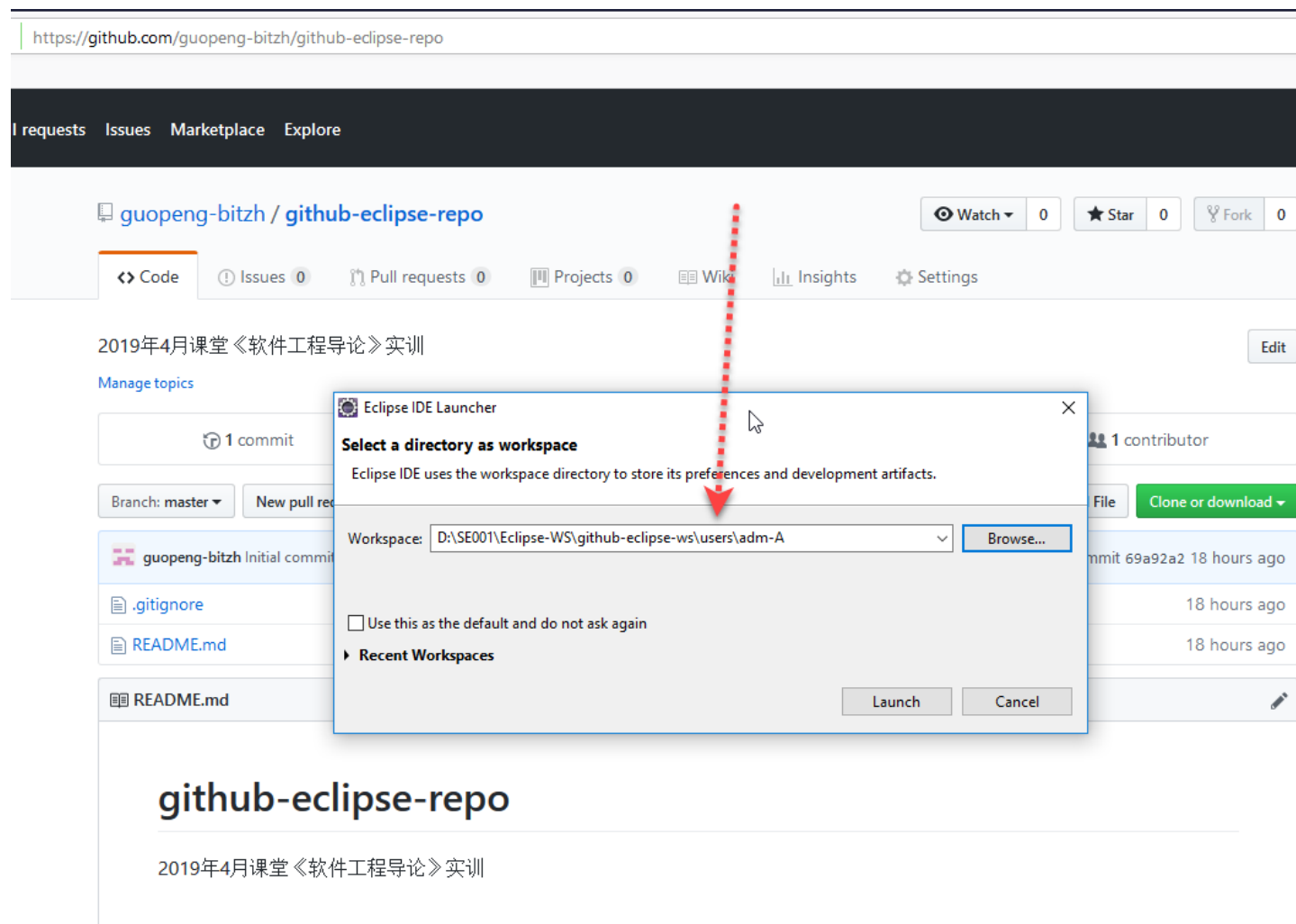
	(软件生命周期) 6 阶段	SDLC 6 phases	8 个 Eclipse project 名称	Eclipse project 性质	内容
1	计划	Planning	phase-01-planning	一般 Eclipse project	存放“软件可行性分析报告”
2	需求分析	Analysis	phase-02-analysis	一般 Eclipse project	存放“需求分析报告”文档
3	设计	Design	phase-03-design	一般 Eclipse project	存放“软件设计”文档
4	编程	Implementation	phase-04-implementation-001-java	Eclipse java project	存放标准 Java 代码
			phase-04-implementation-002-static	Eclipse static project	存放前端 html/JavaScript 代码
			phase-04-implementation-003-javaWeb	Eclipse Java web project	存放后端 JSP/Servlet 代码
5	测试	Testing	phase-05-testing	一般 Eclipse project	存放测试文档和缺陷报告
6	维护	Maintenance	phase-06-maintenance	一般 Eclipse project	存放运维文档



5. 在 eclipse 上用运维管理员 adm-A 的身份建立 eclipse projects , 然后上传到 github 仓库, 作为一个新工程项目的开始版本, 可以给团队中所有人按各自的角色去使用。

步骤 1: 新建 adm-A 的 workspace :

D:\SE001\Eclipse-WS\github-eclipse-ws\users\adm-A



步骤 2： 在 adm-A 的 workspace 中建立一般 Eclipse project：

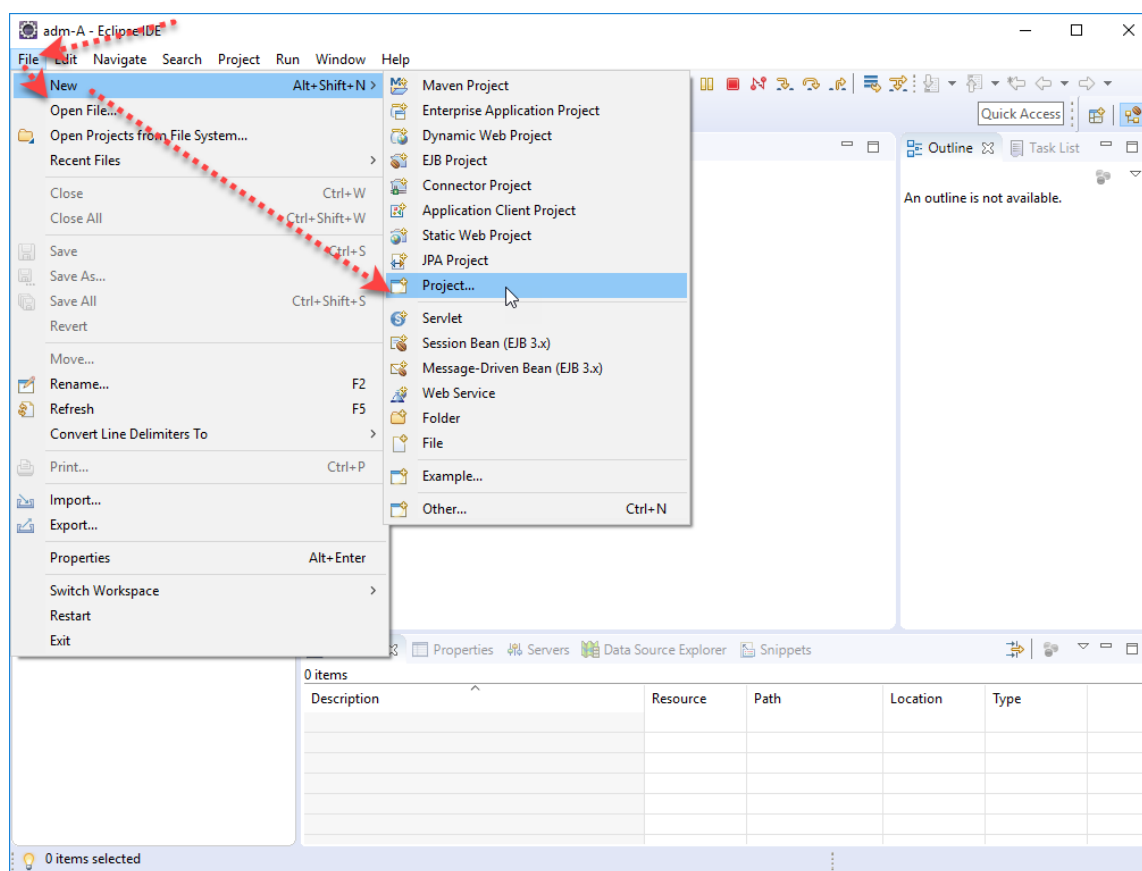
phase-01-planning

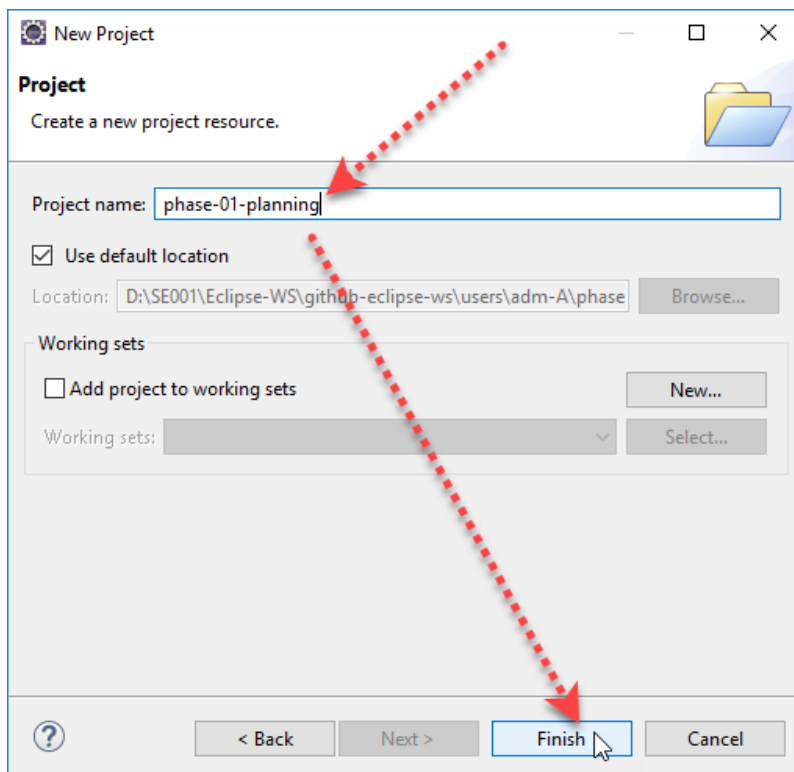
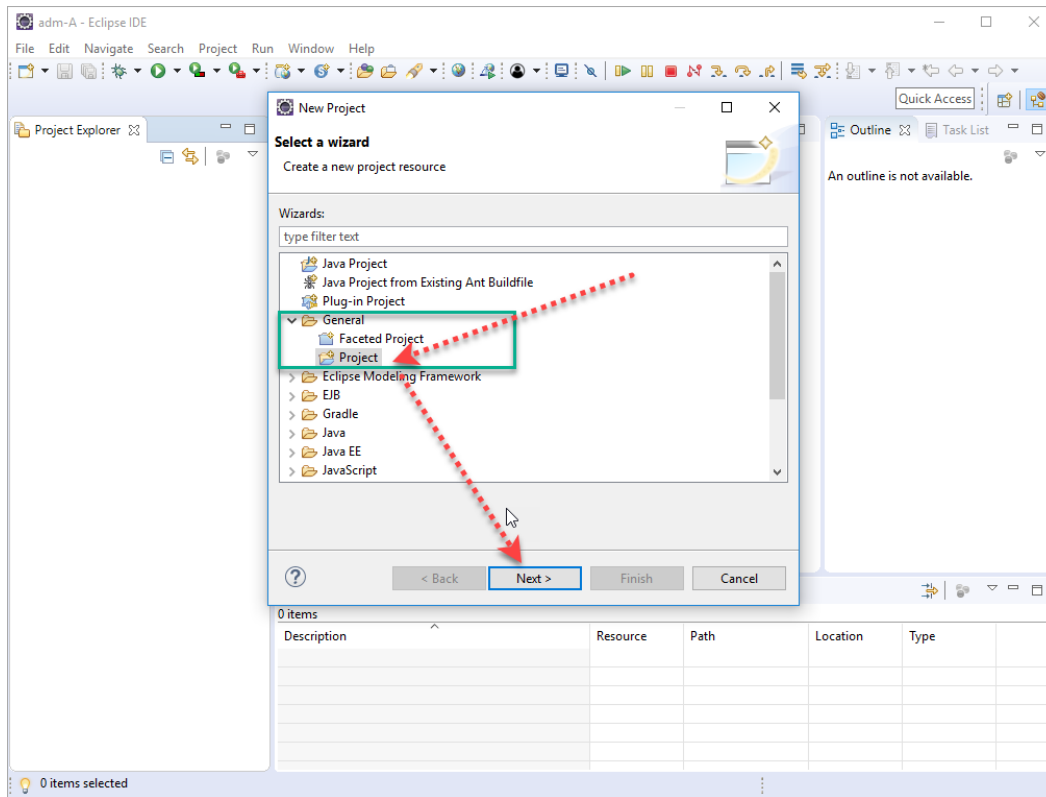
phase-02-analysis

phase-03-design

phase-05-testing

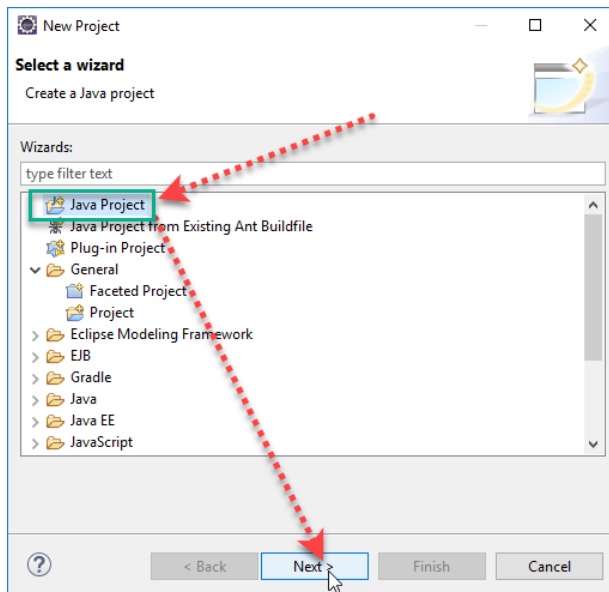
phase-06-maintenance

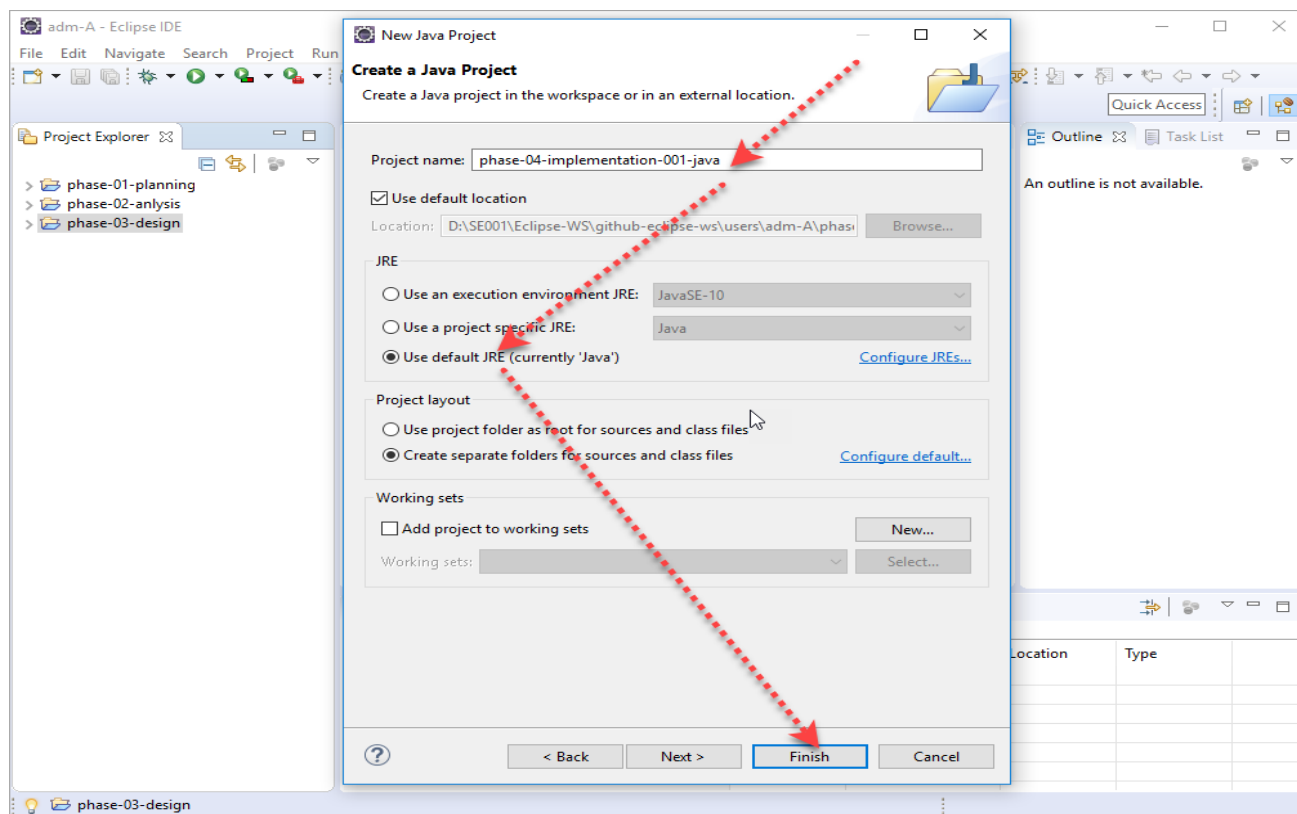




步骤 3： 在 adm-A 的 workspace 中建立一 Eclipse Java Project：

phase-04-implementation-001-java （这种 project 用途呢？是存放动态代码的共享 utility 工具代码）

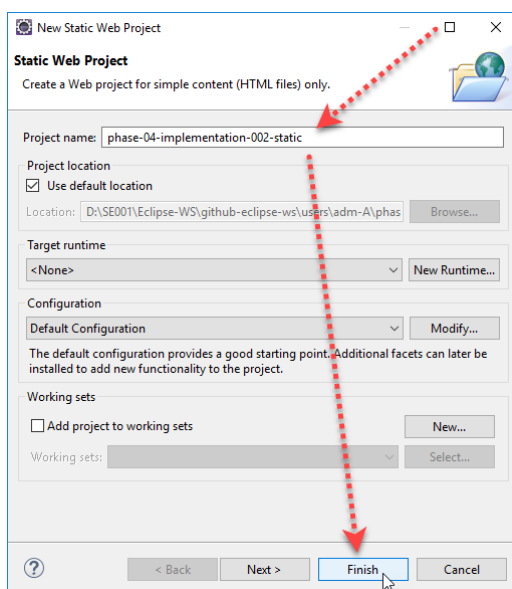
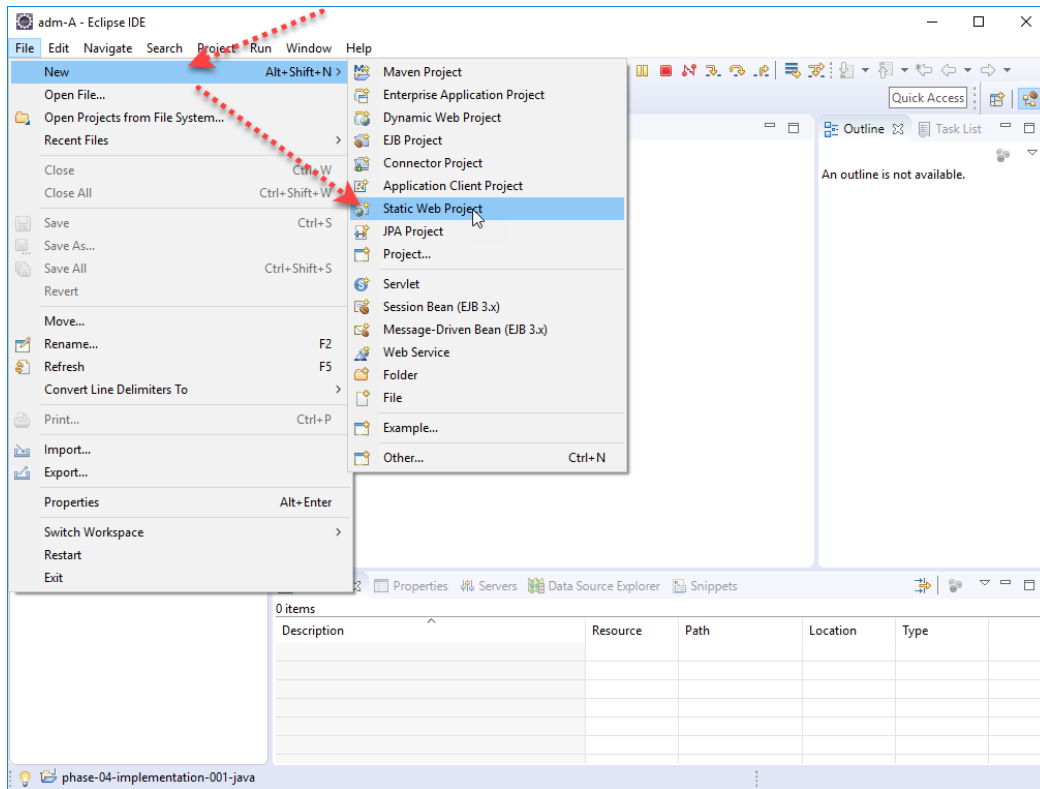






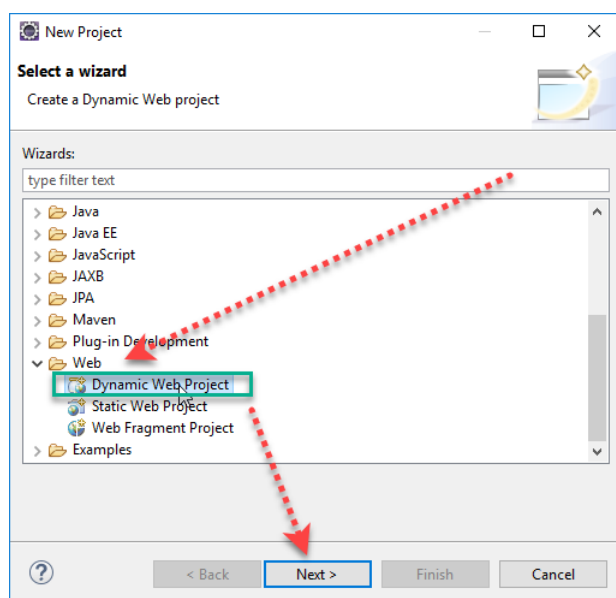
步骤 4: 在 adm-A 的 workspace 中建立一 Eclipse **static** project :

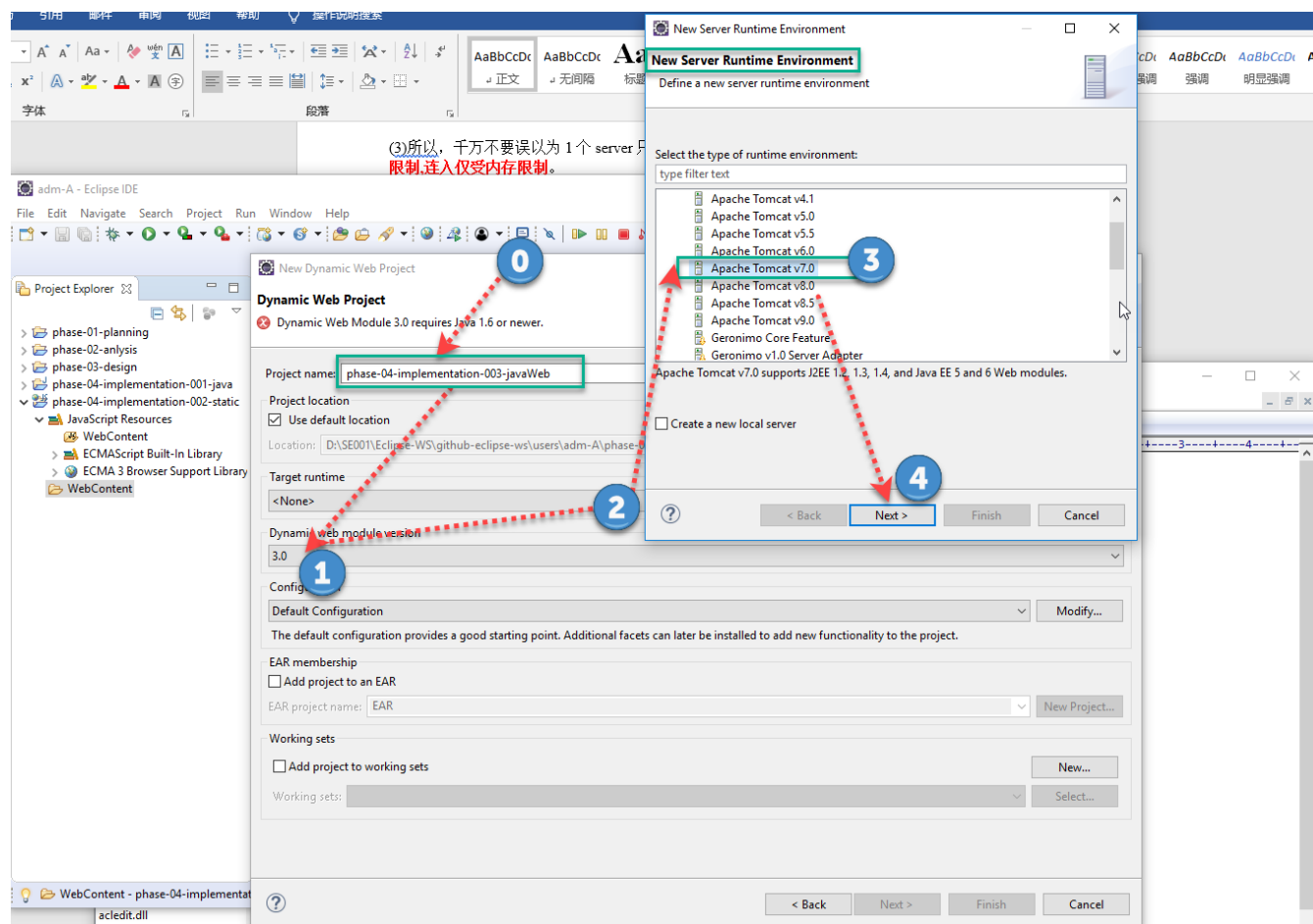
phase-04-implementation-002-static (这种 project 用途呢? 是存放静态代码的共享 utility 工具代码, CSS, javascript, html, 图片, 声音, 视屏等)

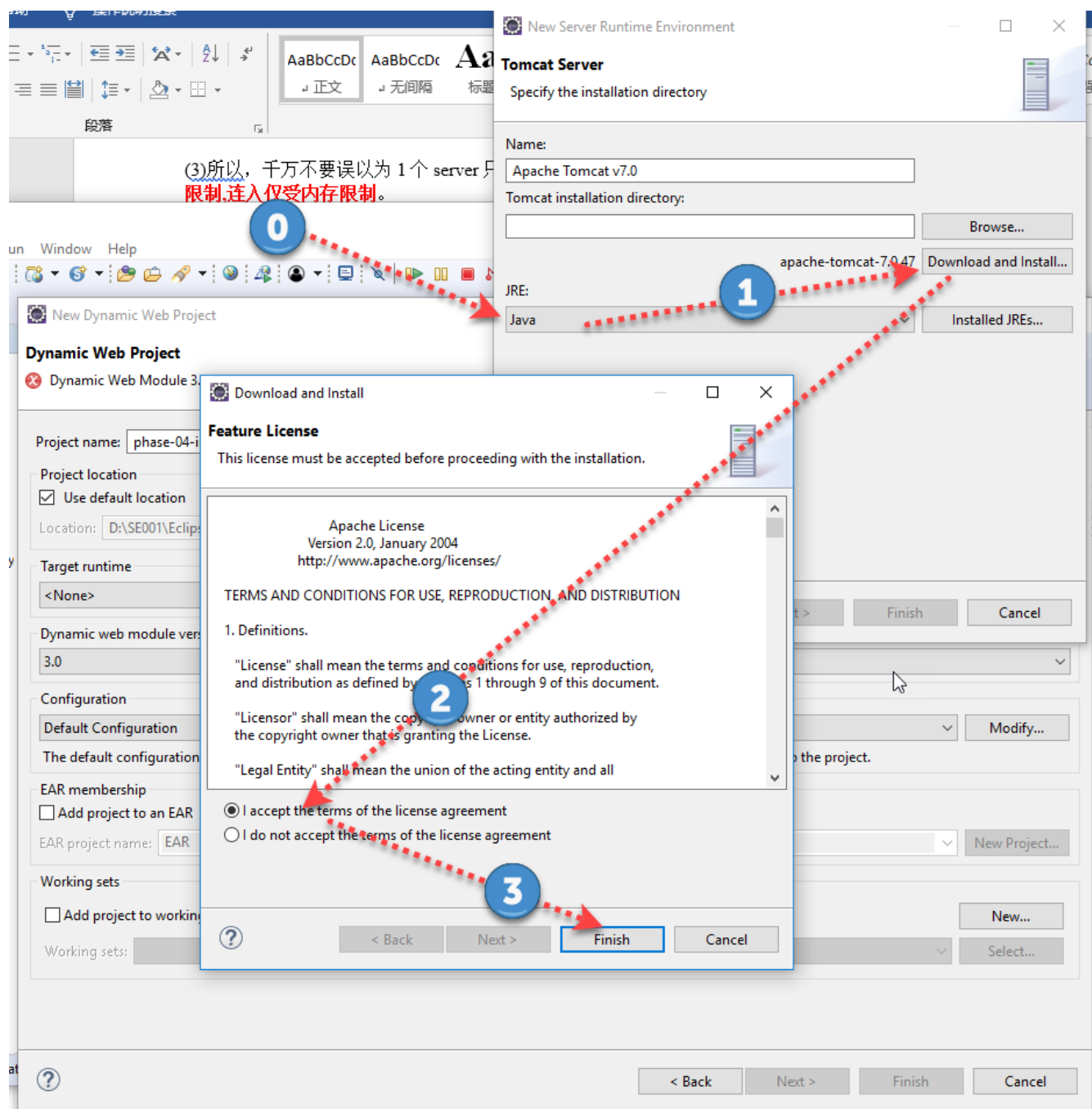


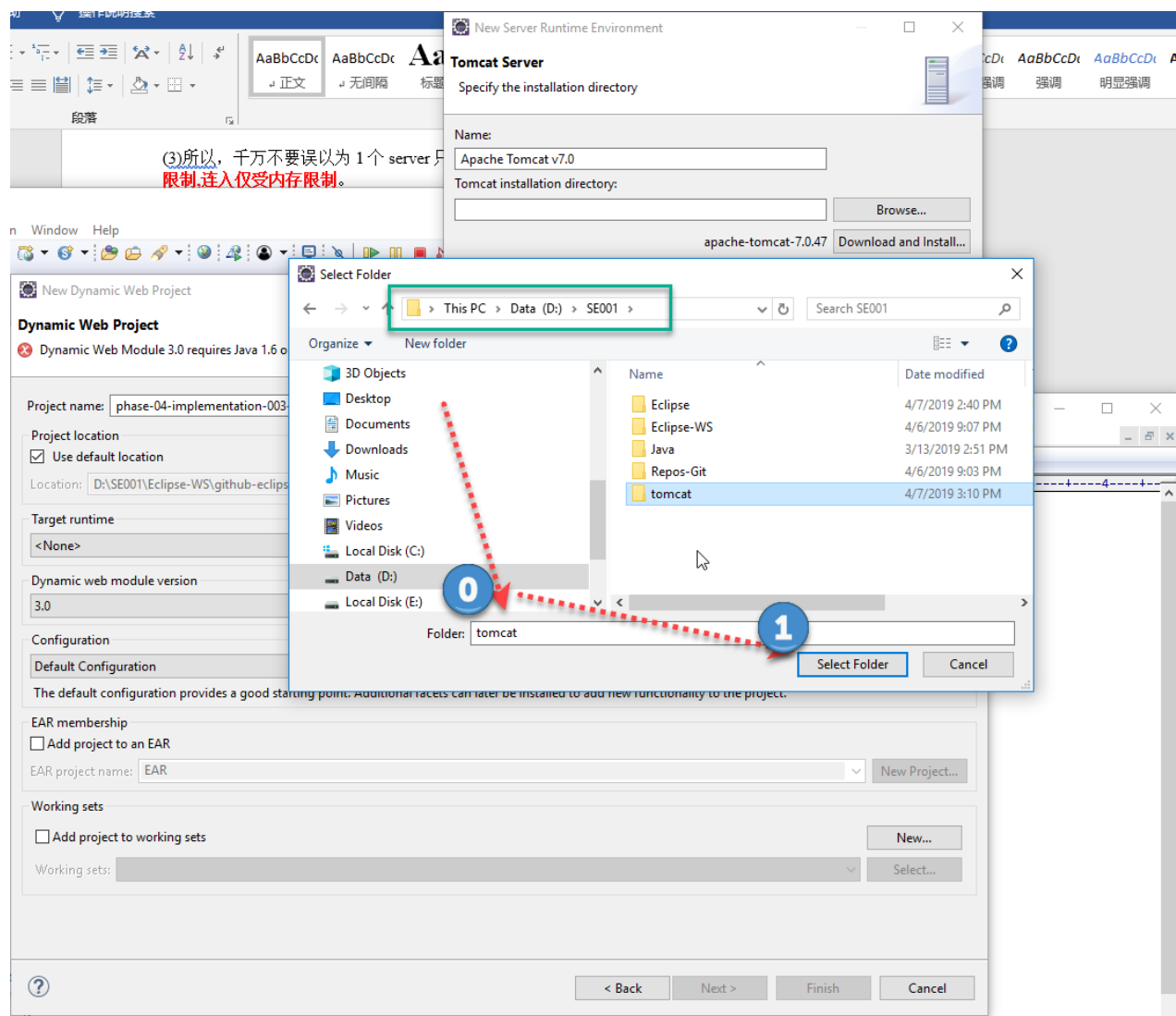
**步骤 5：** 在 adm-A 的 workspace 中建立一 Eclipse dynamic web project ：

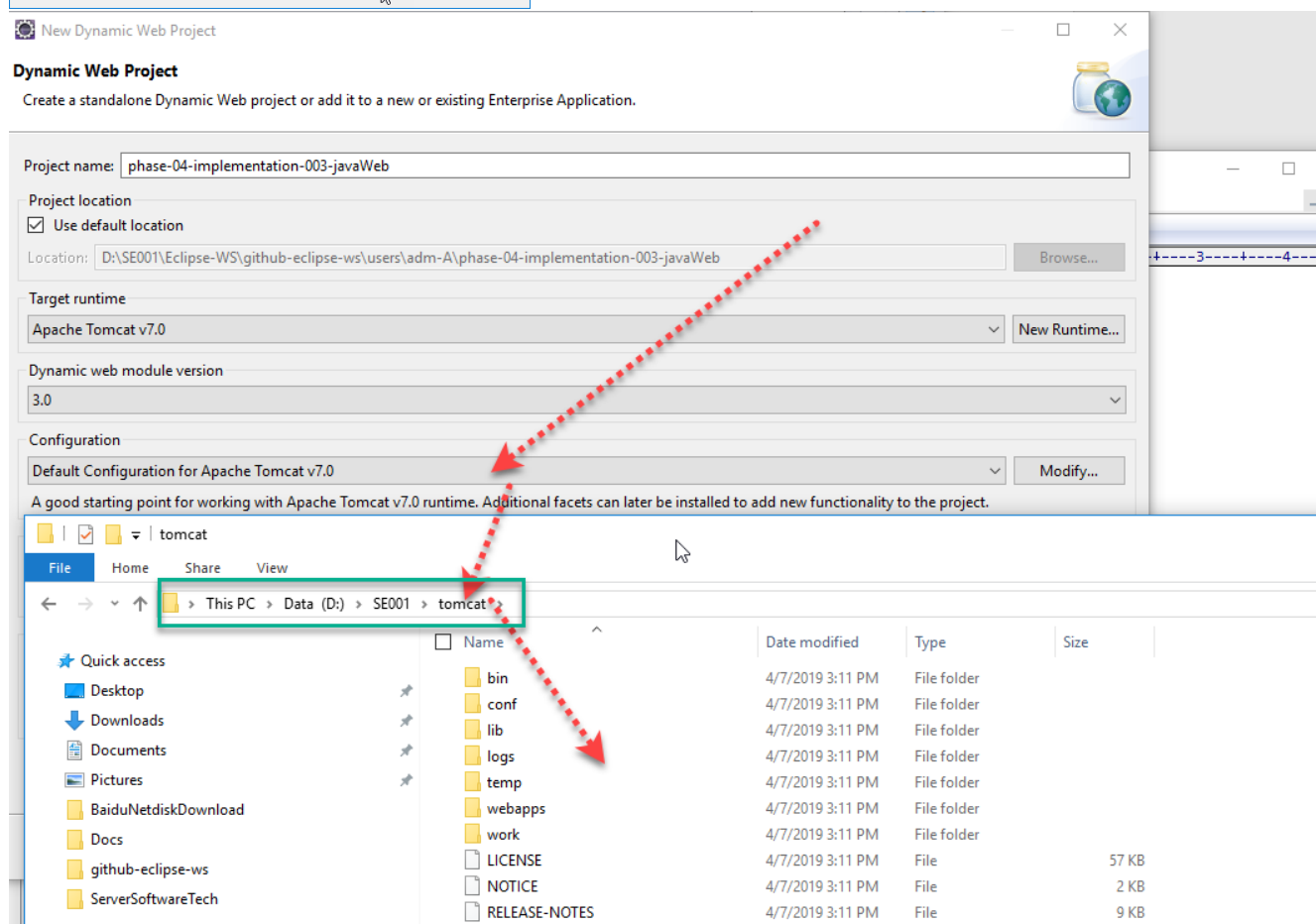
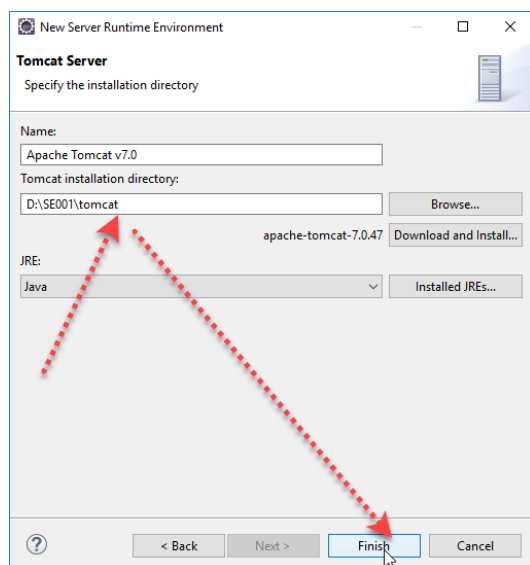
phase-04-implementation-003-javaWeb（这种 project 用途呢？是存放动态代码如 JSP，Servlet 等，这些可以运行在一个 servlet 容器上，这里的容器，老师要指导大家安装 **Tomcat 7.0** 于目录 D:\SE001\tomcat，并将其使用于 eclipse 当做 server，一来可以运行 phase-04-implementation-003-javaWeb 的代码，在 eclipse 里开发，运行和调试，最后可以打包成 war 文件而部署到生产环境的服务器上，如云托管或其他服务器上）



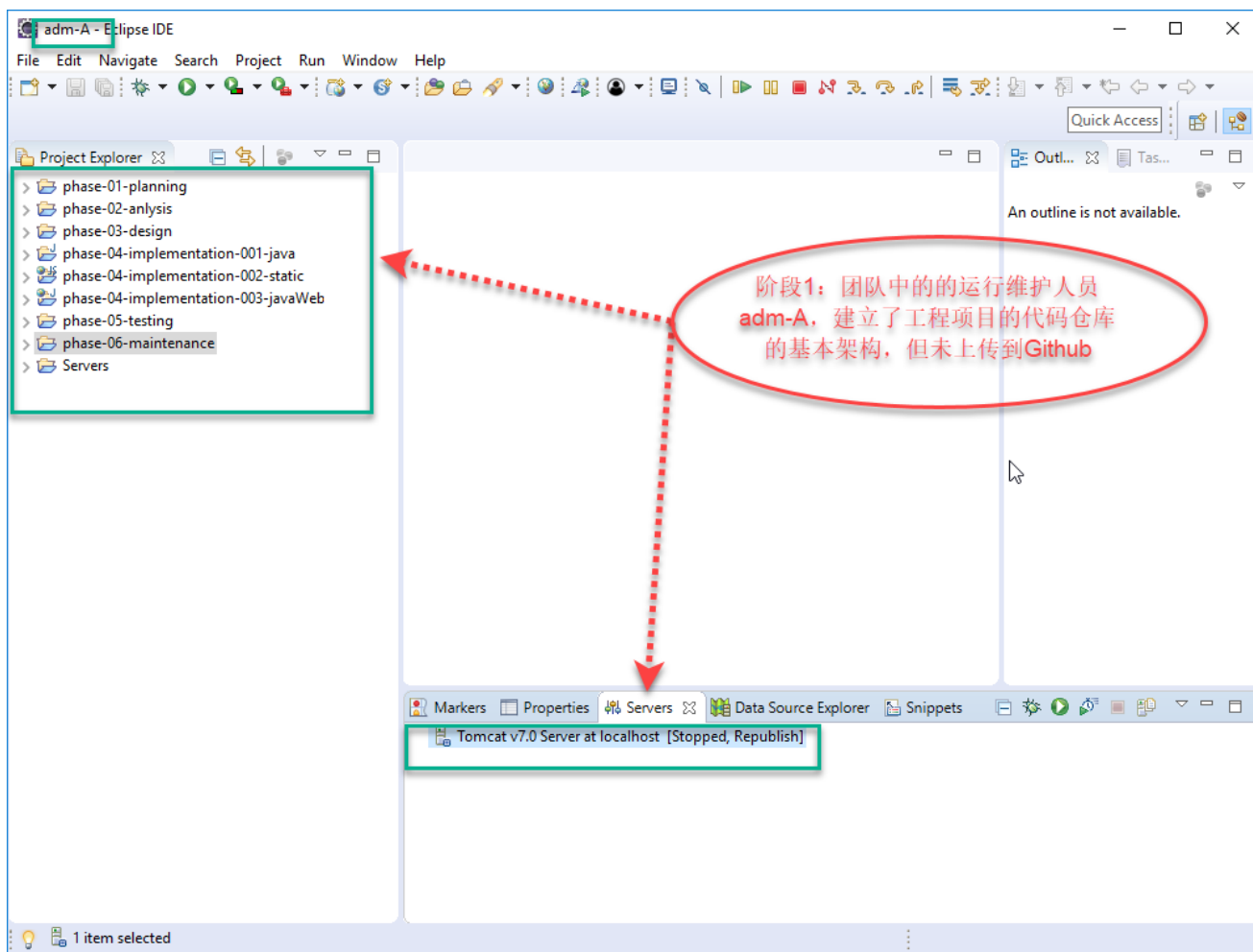








## 5. 里程碑 1：团队中的运行维护人员 adm-A，建立了工程项目的代码仓库的基本架构，但未上传到 Github



6. 里程碑 2: 上传 Eclipse 中的 1 到 3 个到 Github, 而且去 Github 验证一下, 这里笔者先上传以下 1 个 porject, 操作基本相似:

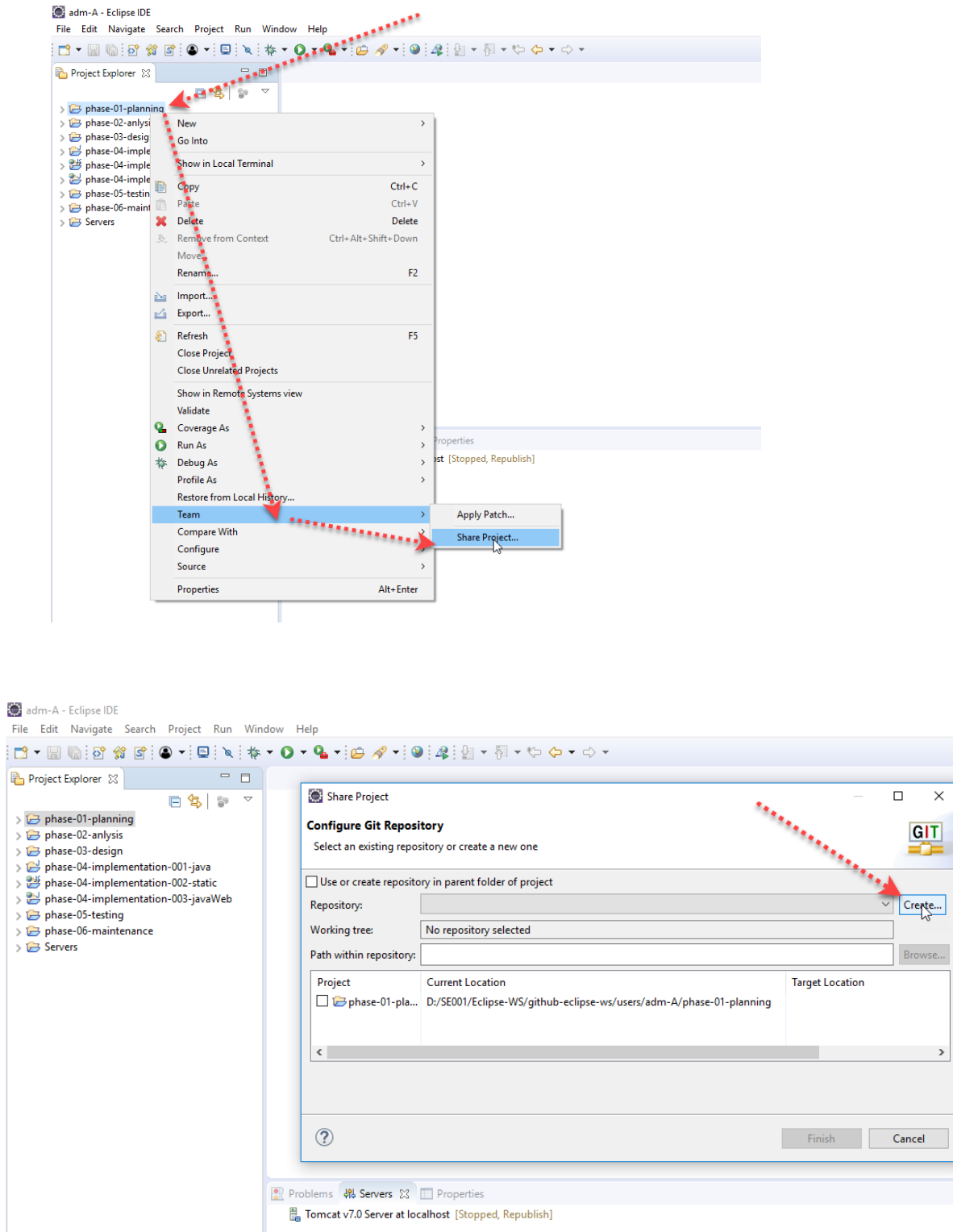
phase-01-planning

大概步骤如下 :

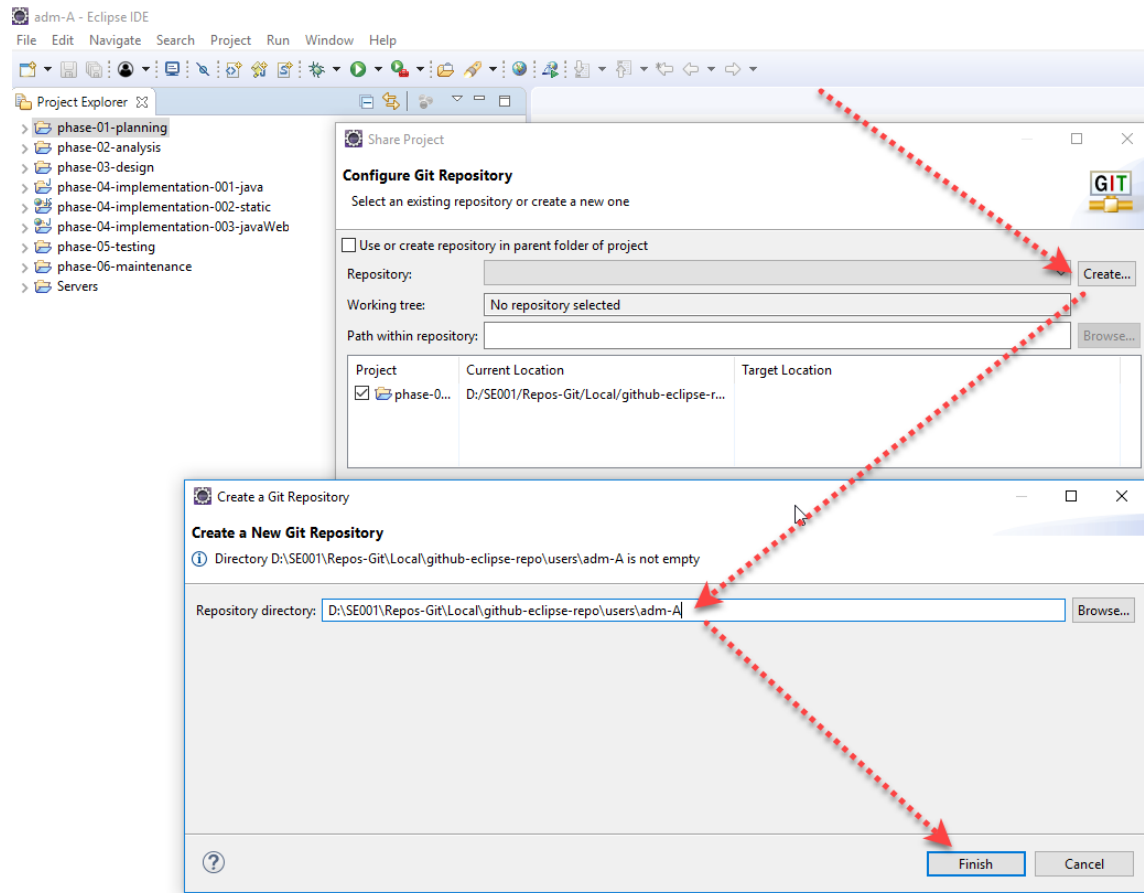
- 1) Eclipse 中 Team ->Add to index
- 2) Team-> Commit
- 3) Team->Remote->Push 这步是真正将代码从 Eclipse 本地仓库上传到 github
- 4) 到 Github 去查 phase-01-planning
- 5) 很好 ! 成功了
- 6) 重新再开一个程序员 dev-A 的工作空间 workspace, 看是否能下载下来  
同时 dev-A 在 phase-01-planning 里建立一个 doc 目录, 并放一份 word 文件  
《流浪地球》项目意向书.docx



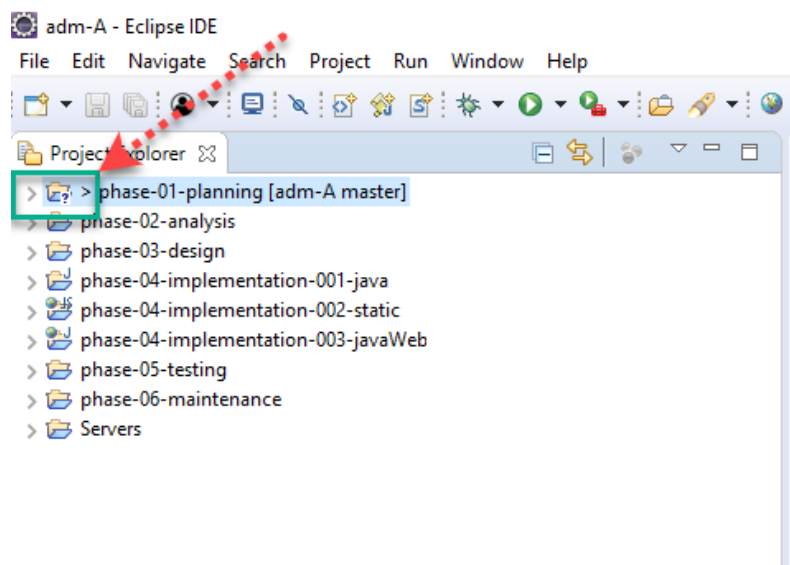
## 步骤 1) Eclipse 中 Team -&gt;Add to index



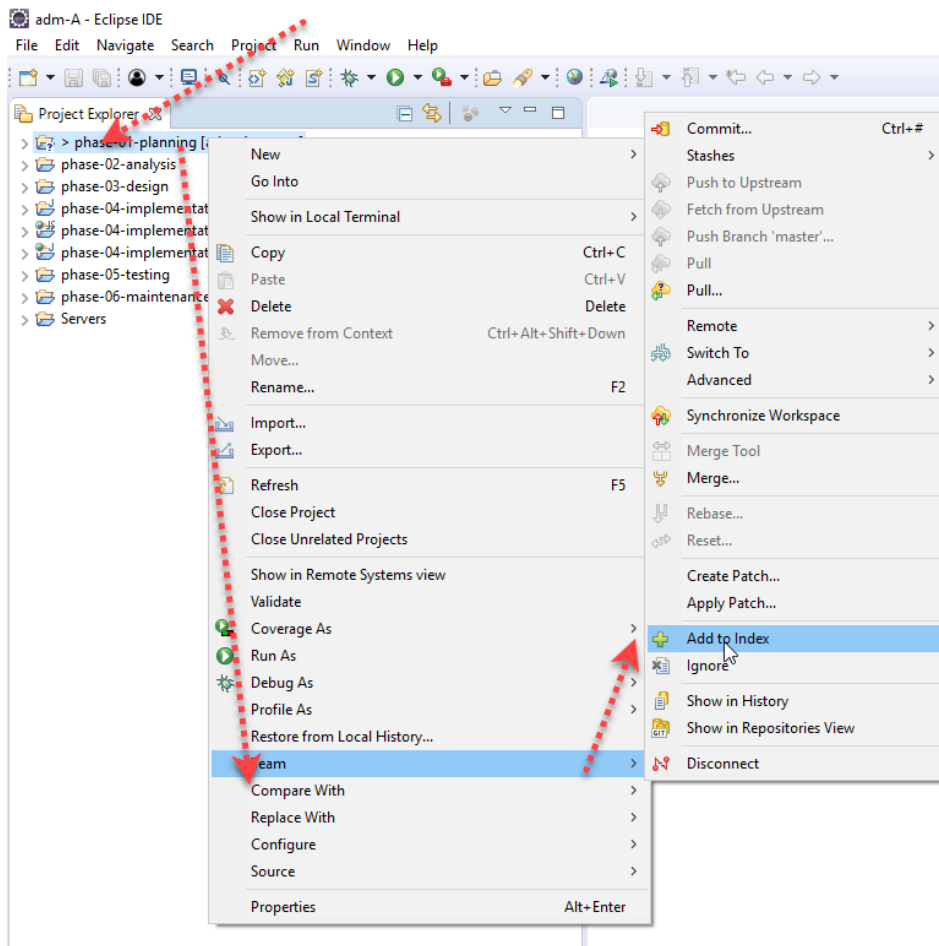
在 Repository directory 一栏输入： `D:\SE001\Repos-Git\Local\github-eclipse-repo\users\adm-A`



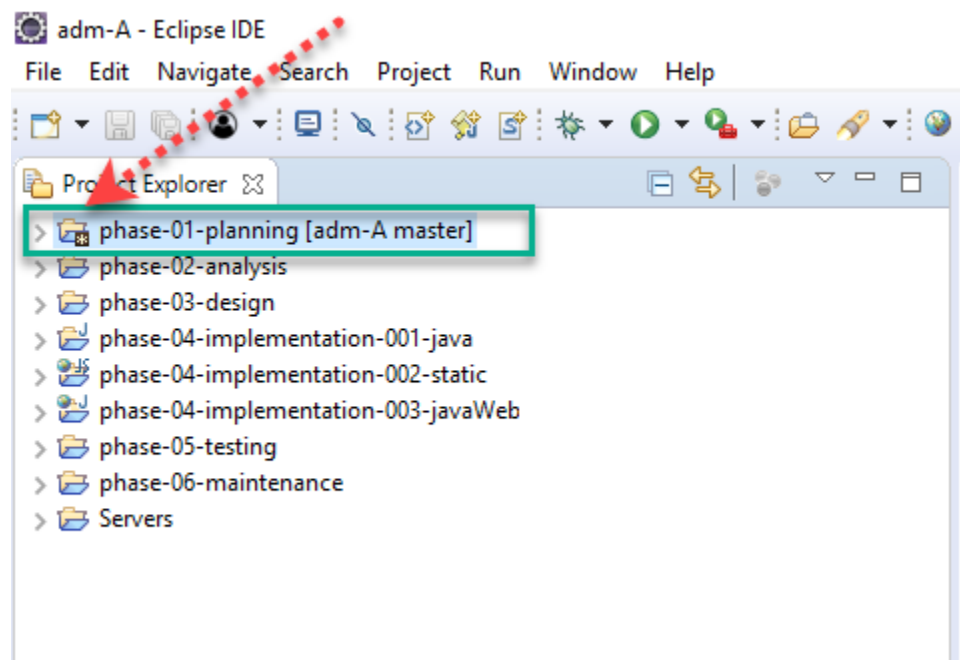
建立本地代码仓库后图标变化如下:



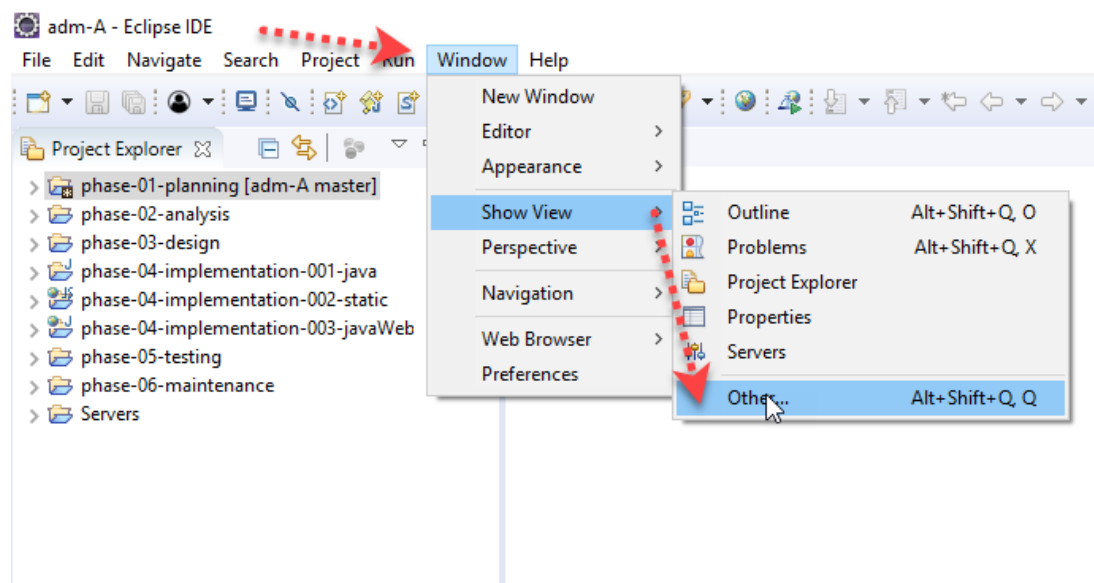
执行加入索引库操作： Project -> Team->Add to index

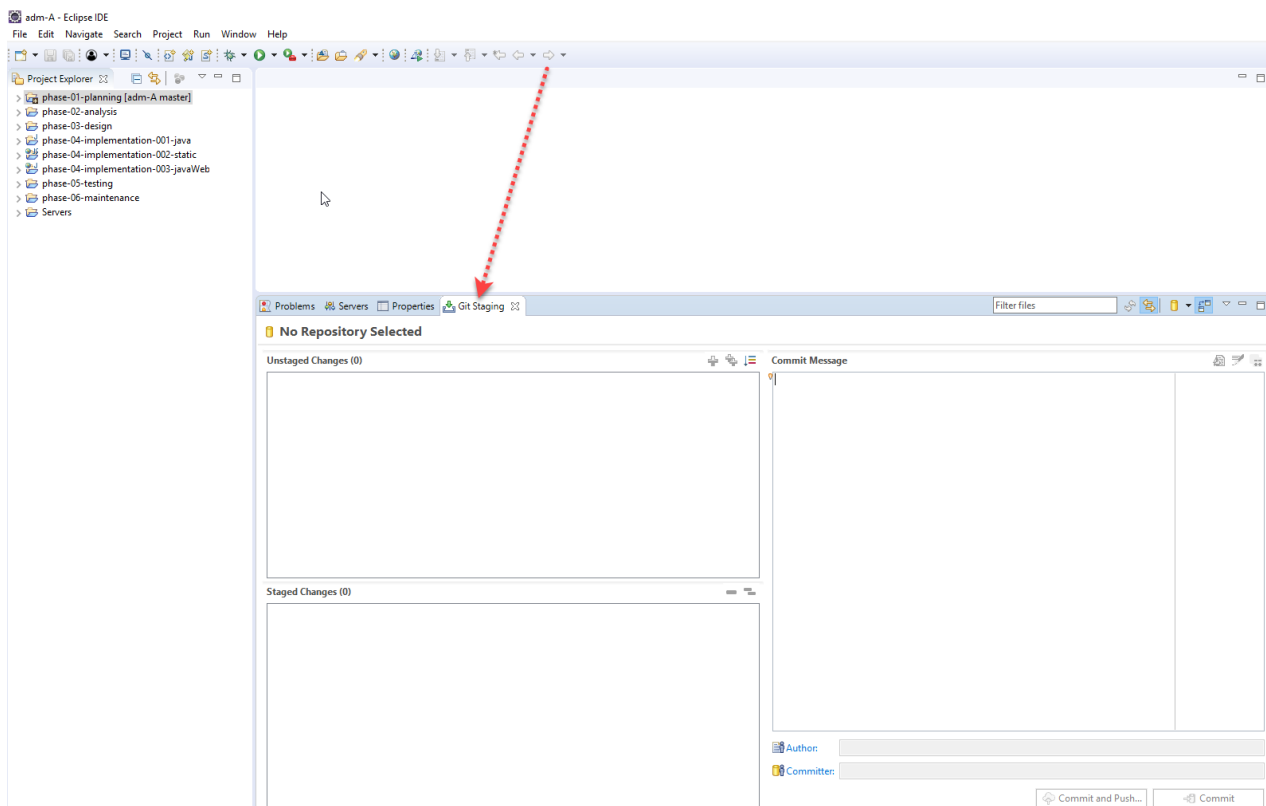
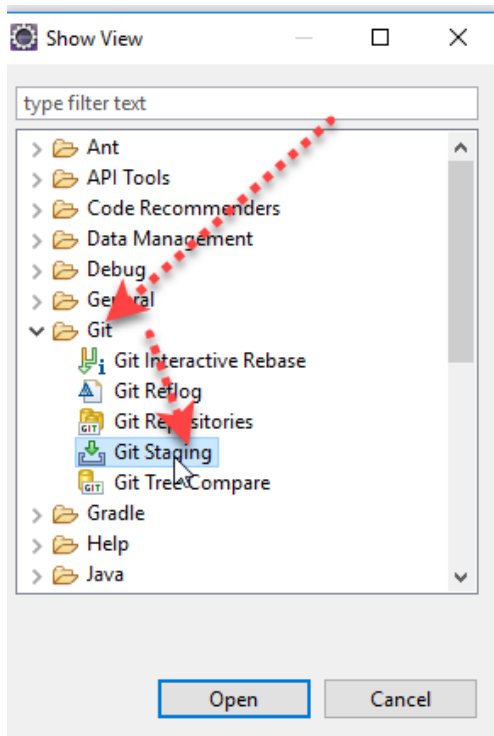


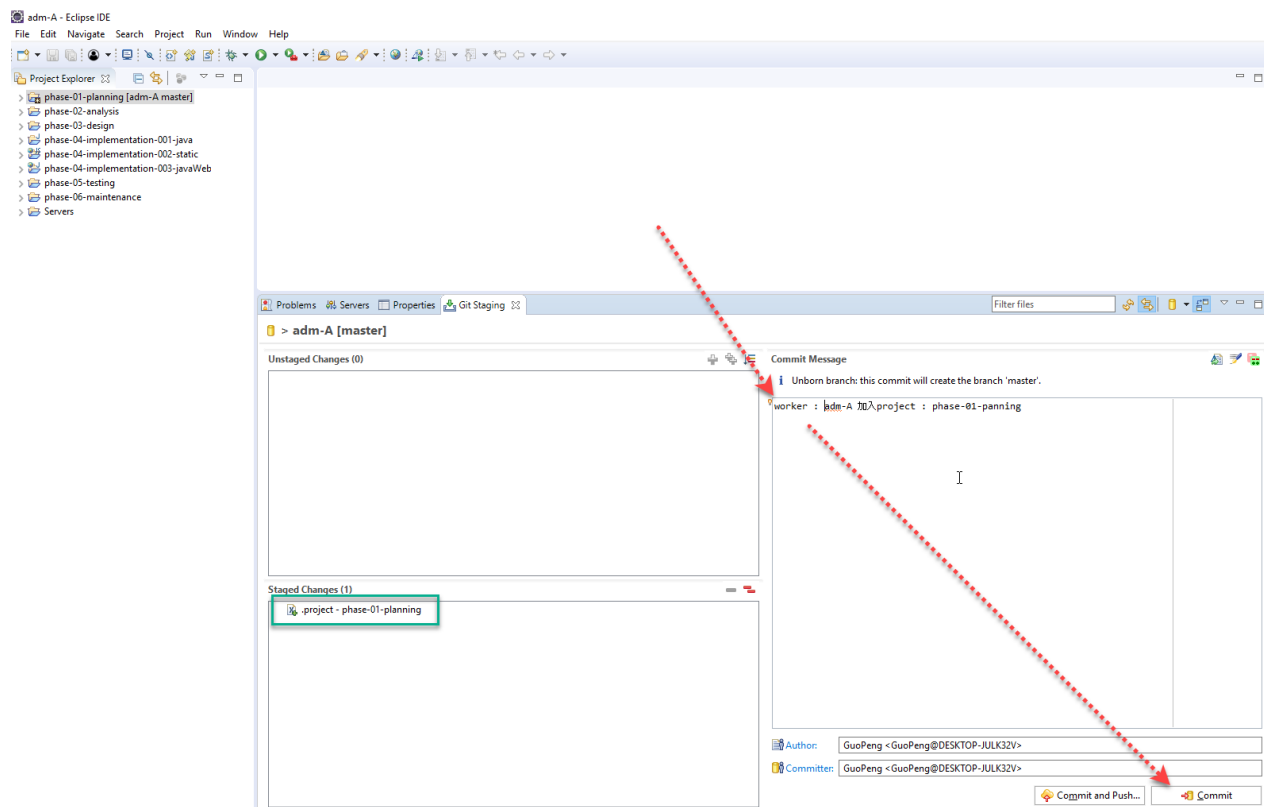
加入索引库后图标变化如下:



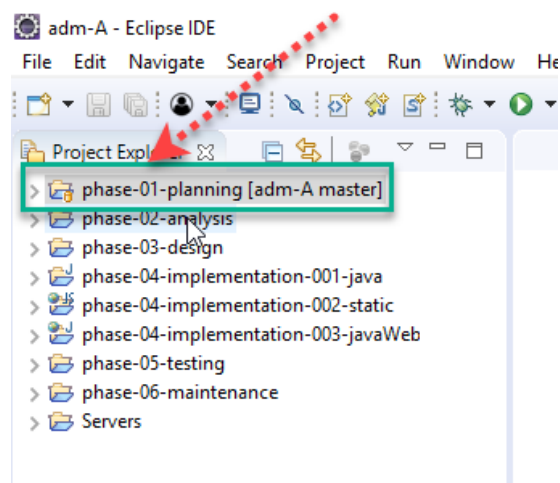
在 Eclipse 打开仓库 Git 工作台的视图 (Git Staging View)



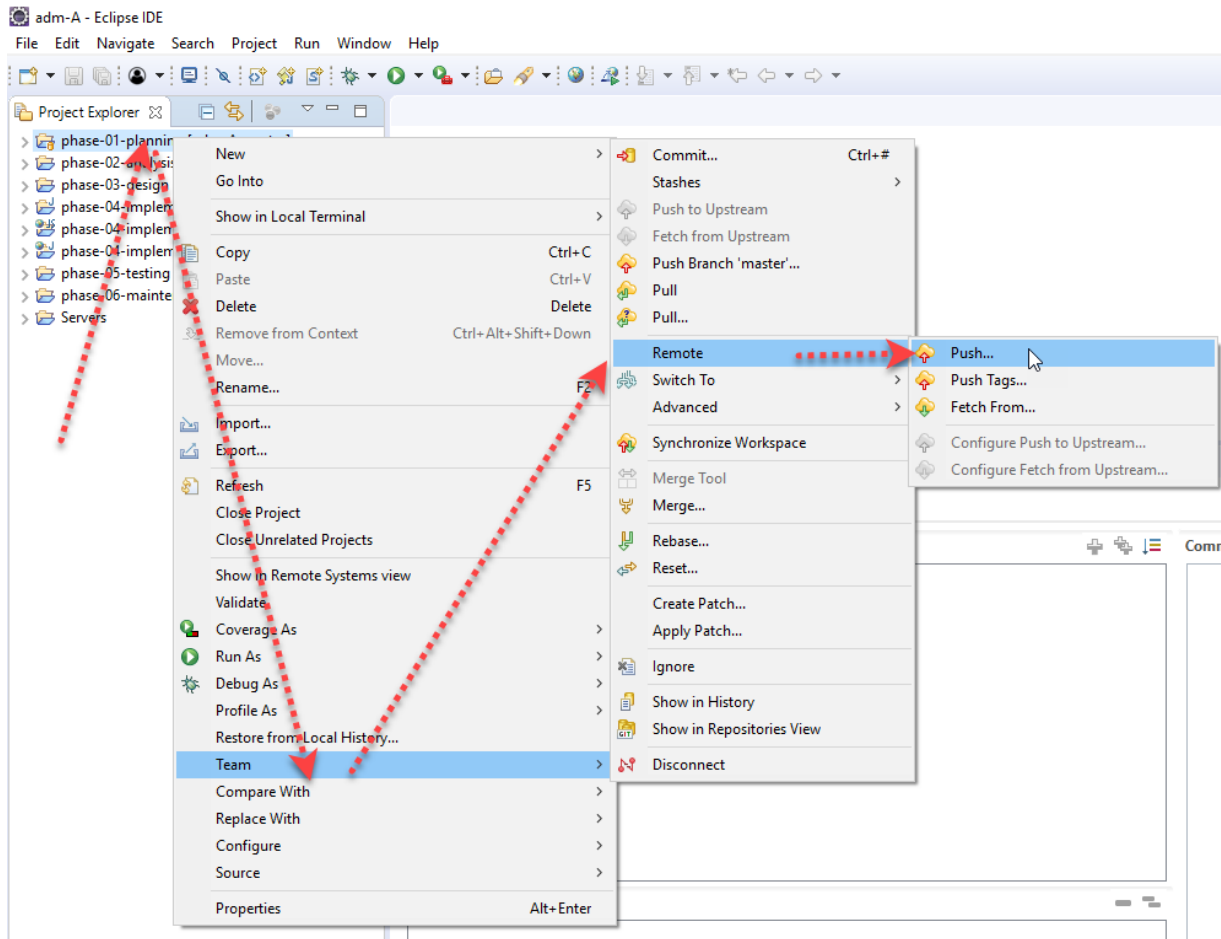




提交本地仓库（commit）后图标变化如下：

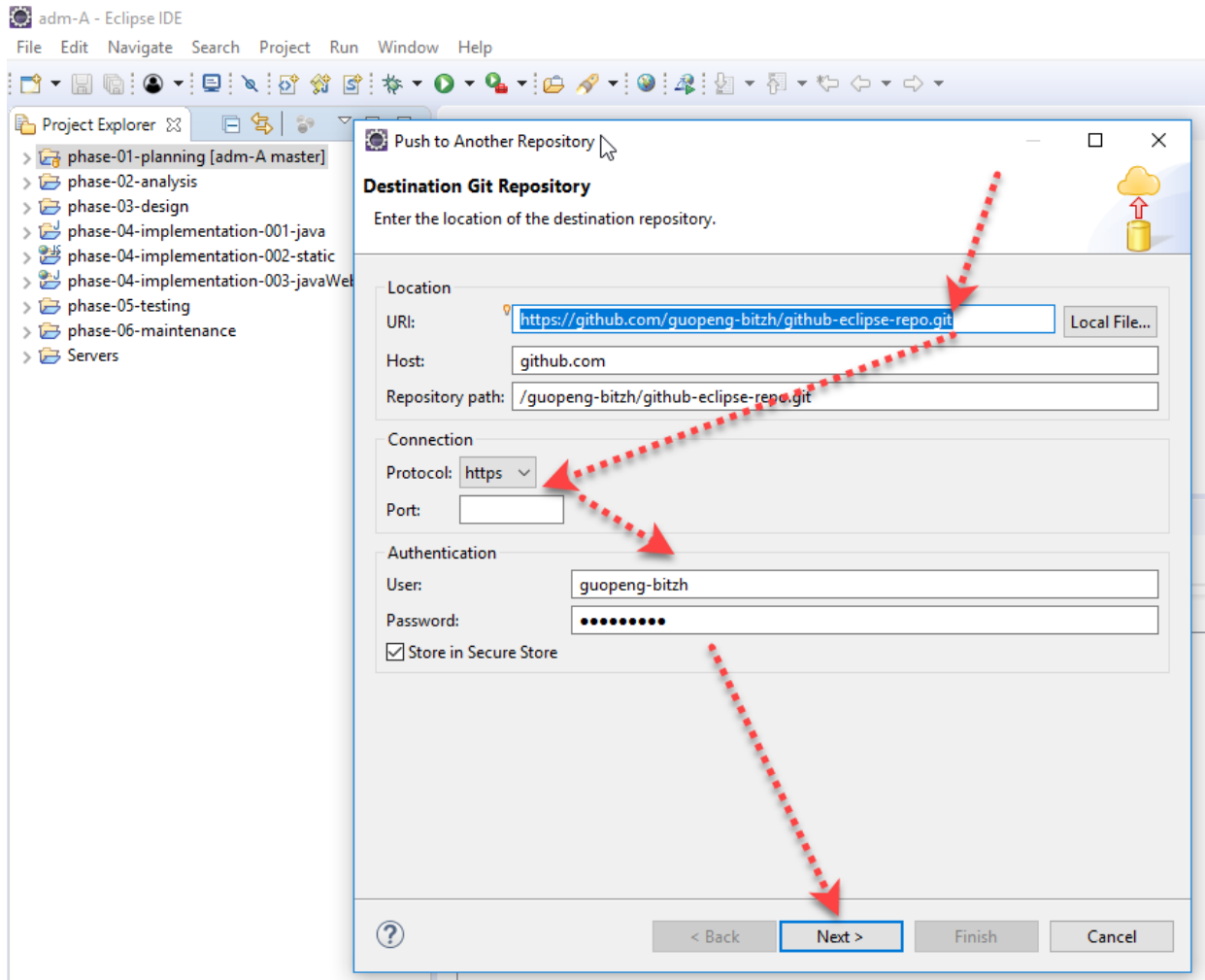


## 如何推 (Push) 本地仓库到远方仓库 (Remote) 的 Github

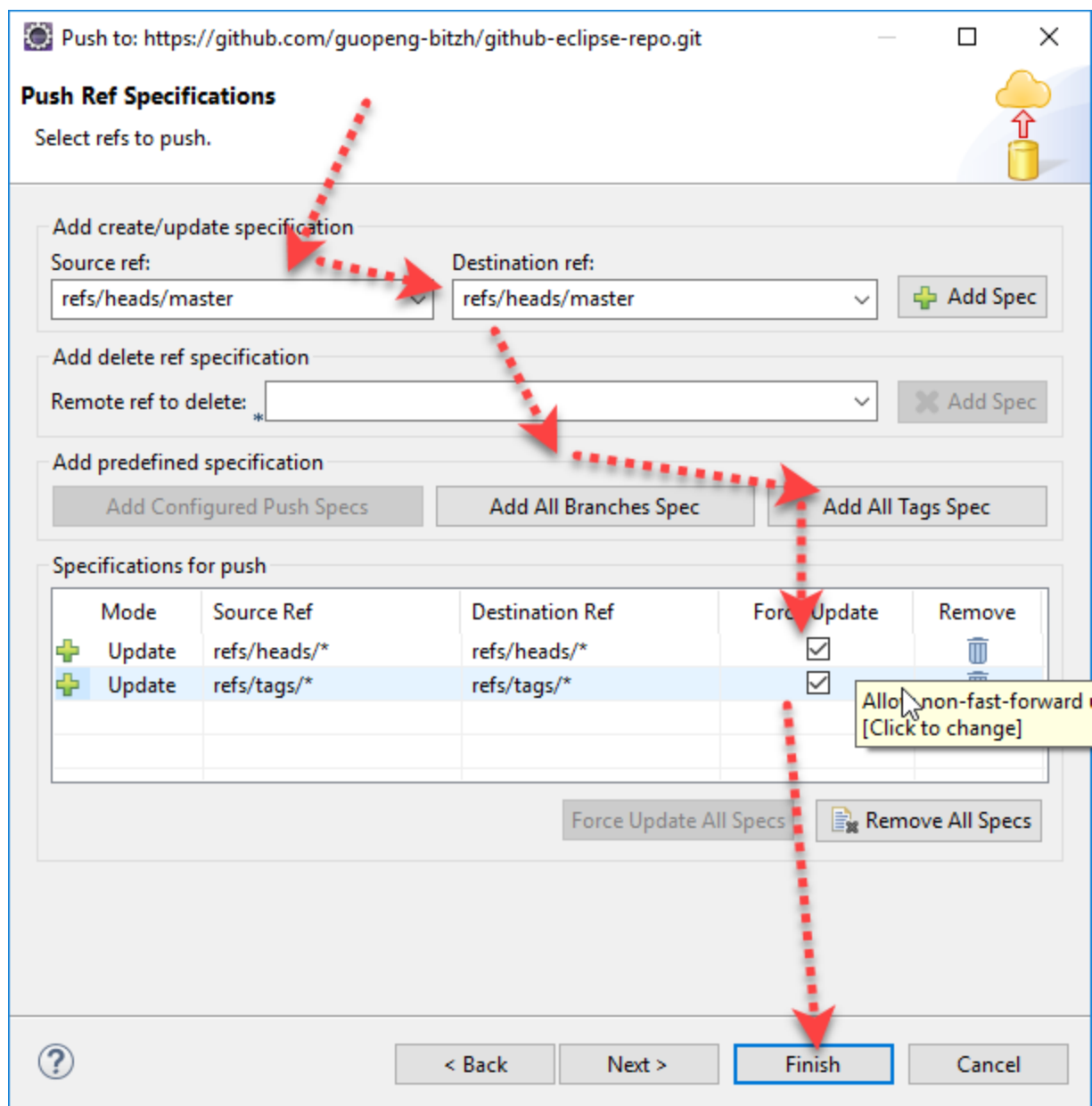




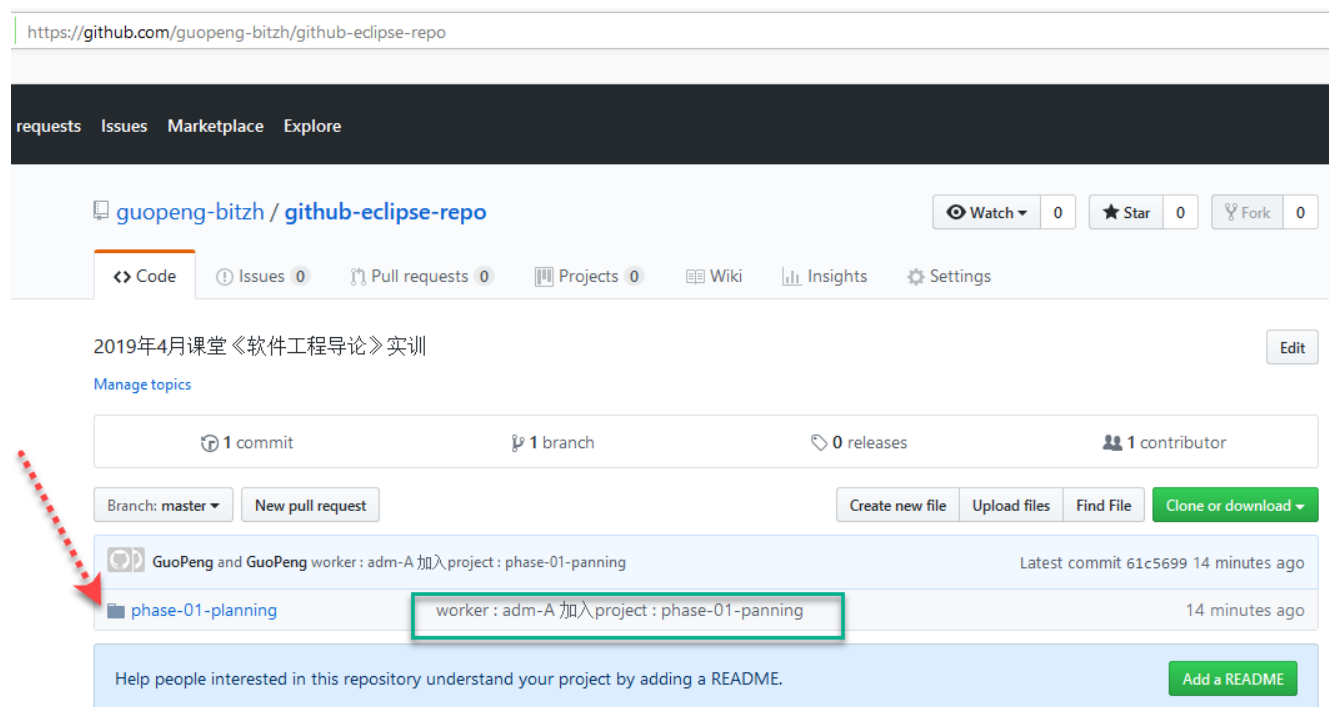
将 <https://github.com/guopeng-bitzh/github-eclipse-repo.git> 填写于 URI 一栏:



适当地配置“Push”：



检查一下自己的 Github，变化如下：phase-01-planning project 已经被加到仓库 github-eclipse-repo 了！



## 第一阶段的实验任务完成！