



1、下载项目。

图 1

2、导入已有的 Maven 工程，点击 file，选择 import，然后出现图 2 所示。

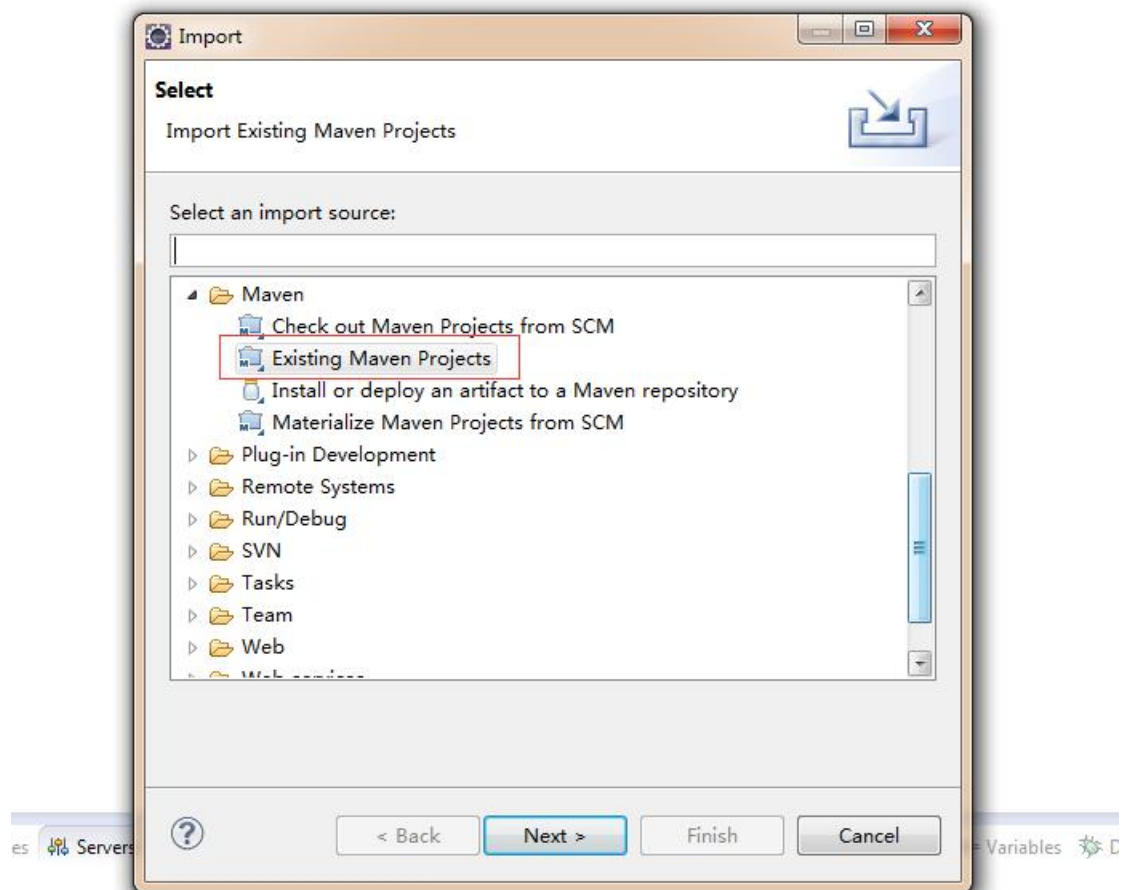


图 2

3、导入本地的工程，以我本地为例，选择 F:\tianti-master，点击 Browser，出现图 3 所示。

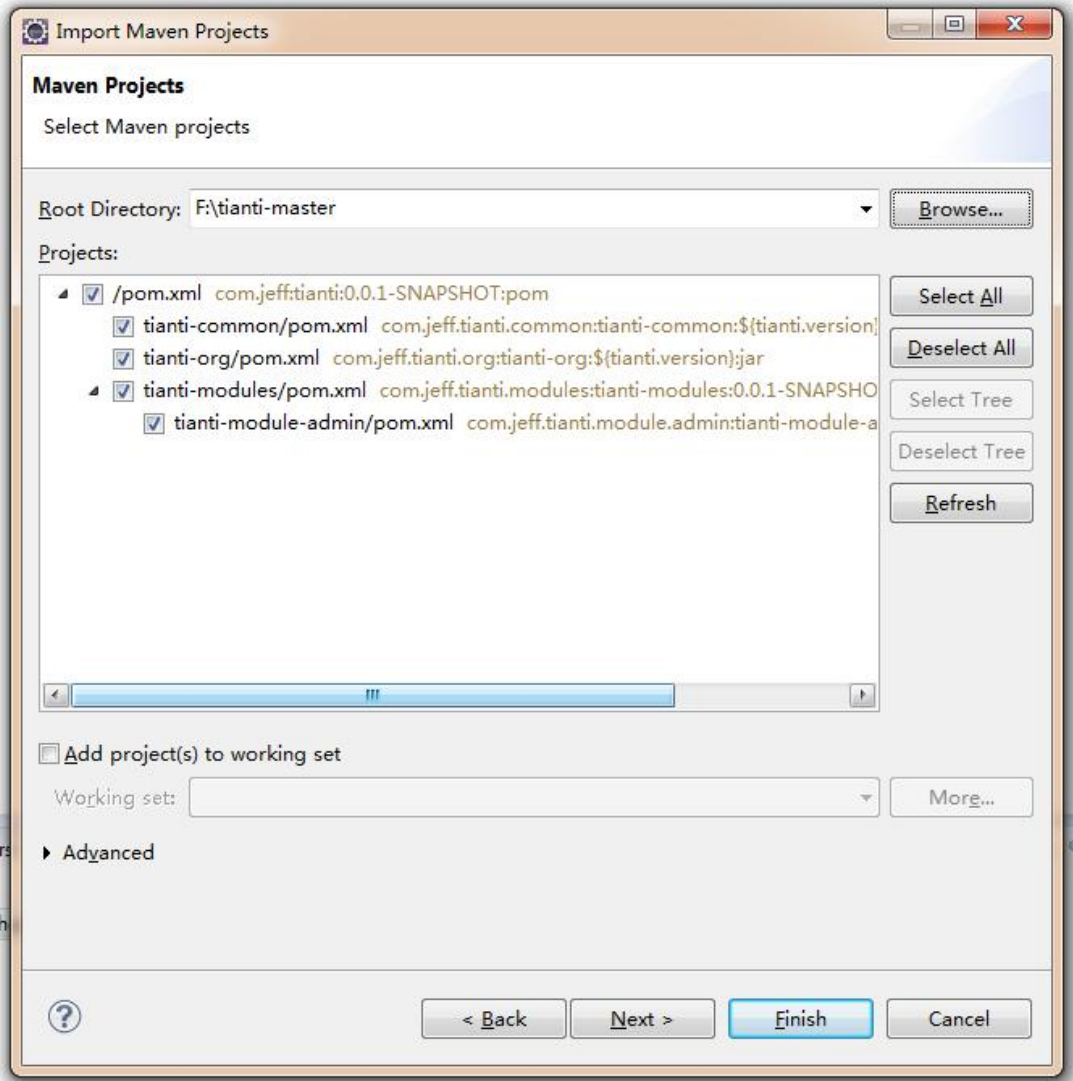


图 3

4、点击 Next、Finish 后，Eclipse 会出现图 4 所示。

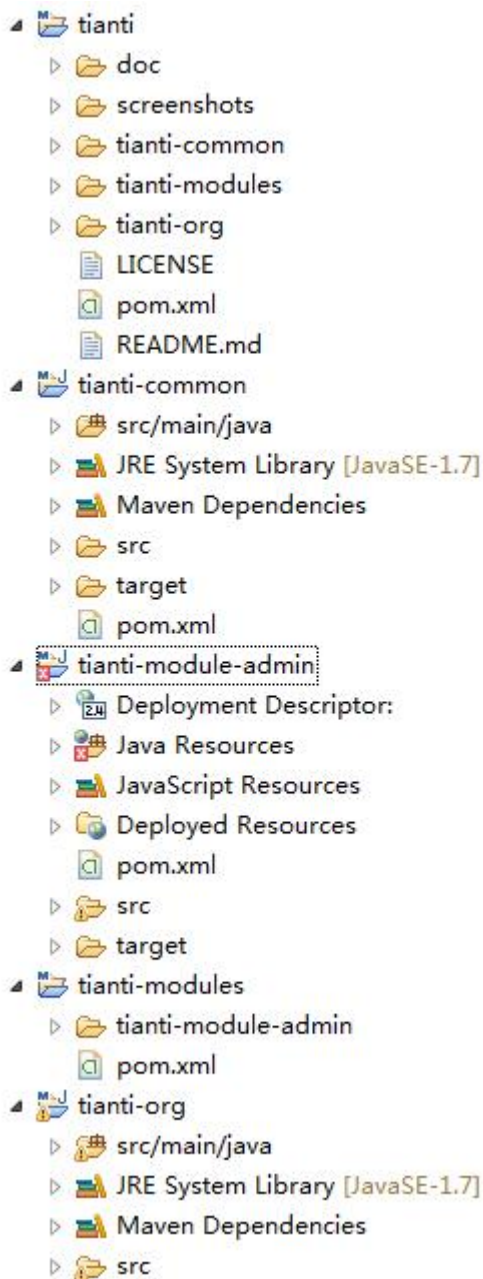


图 4

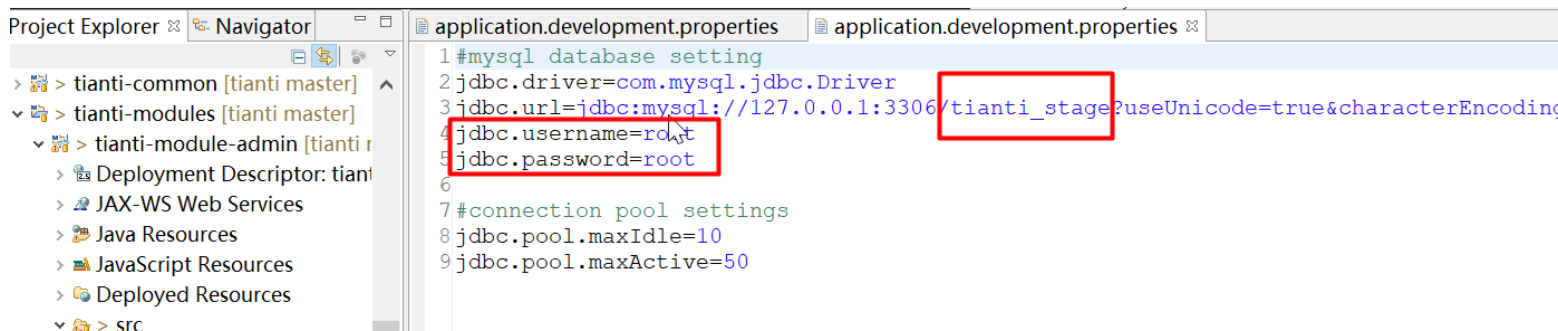
5、修改配置文件的数据库连接：

修改

/tianti-module-admin/src/main/resources/application.development.properties;

修改

/tianti-module-interface/src/main/resources/application.development.properties





使用说明：

0、本项目JDK默认显示的是1.7，用户可以自行选择1.7+。

1、数据库使用mysql，初始化脚本位于tianti-module-admin中的src/main/webapp/scripts/tianti_stage.sql。

2、后台的登陆路径为

`http://ip:端口/tianti-module-admin/login`

用户名为admin，初始密码为123456。

3、后台中所用到的图标是从font文件夹里面选择copy进去的，font文件夹在聊天群里面拥有。

4、编辑器的配置修改editor_config.js的var context_ =

“`http://127.0.0.1:8080/tianti-module-admin`”；

5、前端项目（tianti-module-gateway）是一个静态项目，可以直接部署到web容器，支持多端自适应，需要跑接口系统。

6、采用前后端分离机制获取数据，在/src/main/webapp/static/js/config.js文件里分别配置接口地址和后台项目地址；

7、前台项目可以直接部署到Nginx访问，也可以跑tianti-module-gateway应用。

8、如果是tomcat跑前端应用，访问路径为

`http://ip:端口/tianti-module-gateway/tianti/index/index.html`。

6、鼠标右键点击 tianti, 选择 Run As->Maven install,控制台会输出 BUILD 如图 5 所示。

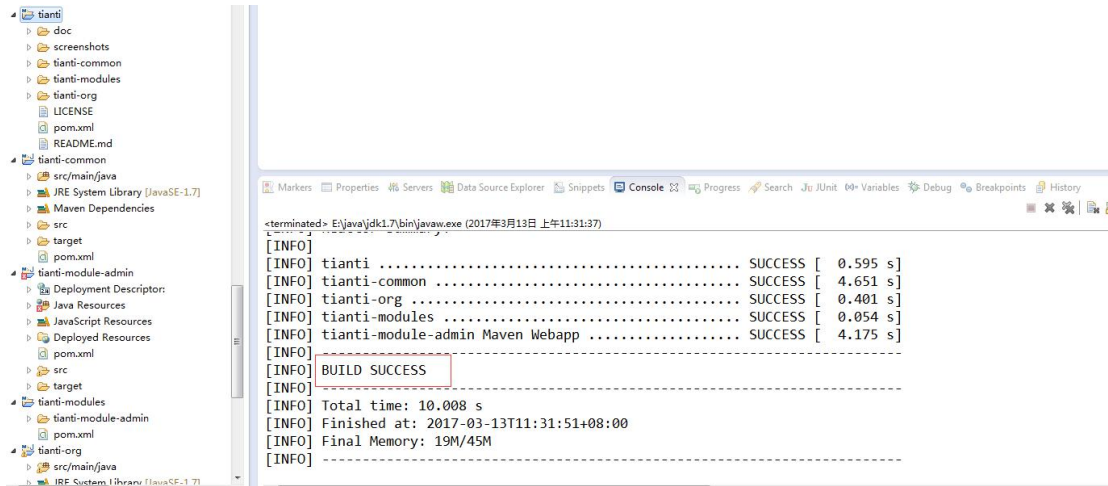


图 5

7、将 tianti-module-admin、tianti-module-gateway、tianti-module-interface, 添加到 Tomcat, 然后启动, 如图 6 所示。

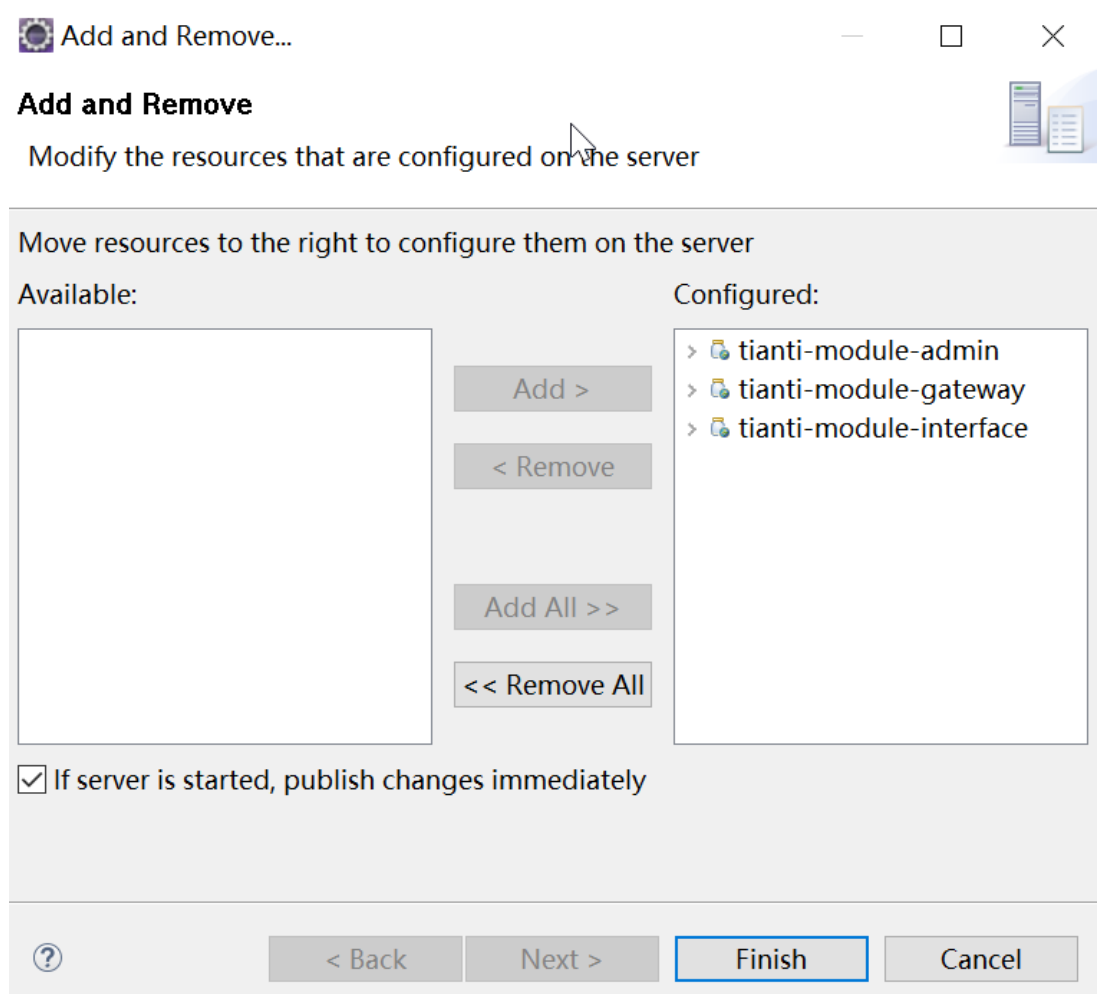


图 6



8、启动成的如图7所示

```
Markers Servers Console Problems Error Log History Search Debug Progress JUnit Call Hierarchy Sync
Tomcat v8.0 Server at localhost (3) [Apache Tomcat] C:\Program Files\Java\jdk1.8.0_131\bin\javaw.exe (2019年11月16日 下午2:44:06)
log4j:WARN No appenders could be found for logger (org.jboss.logging).
log4j:WARN Please initialize the log4j system properly.
log4j:WARN See http://logging.apache.org/log4j/1.2/faq.html#noconfig for more info.
十一月16, 2019 2:44:22 下午 org.apache.catalina.core.ApplicationContext log
信息: Set web app root system property: 'web.interface.root' = [D:\Workspace\MyWorkspacel\.metadata\
十一月16, 2019 2:44:22 下午 org.apache.catalina.core.ApplicationContext log
信息: Initializing Spring FrameworkServlet 'springmvc'
十一月16, 2019 2:44:29 下午 org.apache.jasper.servlet.TldScanner scanJars
信息: At least one JAR was scanned for TLDs yet contained no TLDs. Enable debug logging for this lo
十一月16, 2019 2:44:29 下午 org.apache.catalina.core.ApplicationContext log
信息: No Spring WebApplicationInitializer types detected on classpath
十一月16, 2019 2:44:33 下午 org.apache.jasper.servlet.TldScanner scanJars
信息: At least one JAR was scanned for TLDs yet contained no TLDs. Enable debug logging for this lo
十一月16, 2019 2:44:33 下午 org.apache.catalina.core.ApplicationContext log
信息: No Spring WebApplicationInitializer types detected on classpath
SLF4J: Failed to load class "org.slf4j.impl.StaticLoggerBinder".
SLF4J: Defaulting to no-operation (NOP) logger implementation
SLF4J: See http://www.slf4j.org/codes.html#StaticLoggerBinder for further details.
十一月16, 2019 2:44:33 下午 org.apache.catalina.core.ApplicationContext log
信息: Initializing Spring root WebApplicationContext
log4j:WARN No appenders could be found for logger (org.jboss.logging).
log4j:WARN Please initialize the log4j system properly.
log4j:WARN See http://logging.apache.org/log4j/1.2/faq.html#noconfig for more info.
十一月16, 2019 2:44:43 下午 org.apache.catalina.core.ApplicationContext log
信息: Set web app root system property: 'web.manage.root' = [D:\Workspace\MyWorkspacel\.metadata\p
十一月16, 2019 2:44:43 下午 org.apache.catalina.core.ApplicationContext log
信息: Initializing Spring FrameworkServlet 'springmvc'
十一月16, 2019 2:44:45 下午 org.apache.coyote.AbstractProtocol start
信息: Starting ProtocolHandler ["http-apr-8080"]
十一月16, 2019 2:44:45 下午 org.apache.coyote.AbstractProtocol start
信息: Starting ProtocolHandler ["ajp-apr-8009"]
十一月16, 2019 2:44:45 下午 org.apache.catalina.startup.Catalina start
信息: Server startup in 36674 ms
```

图 7