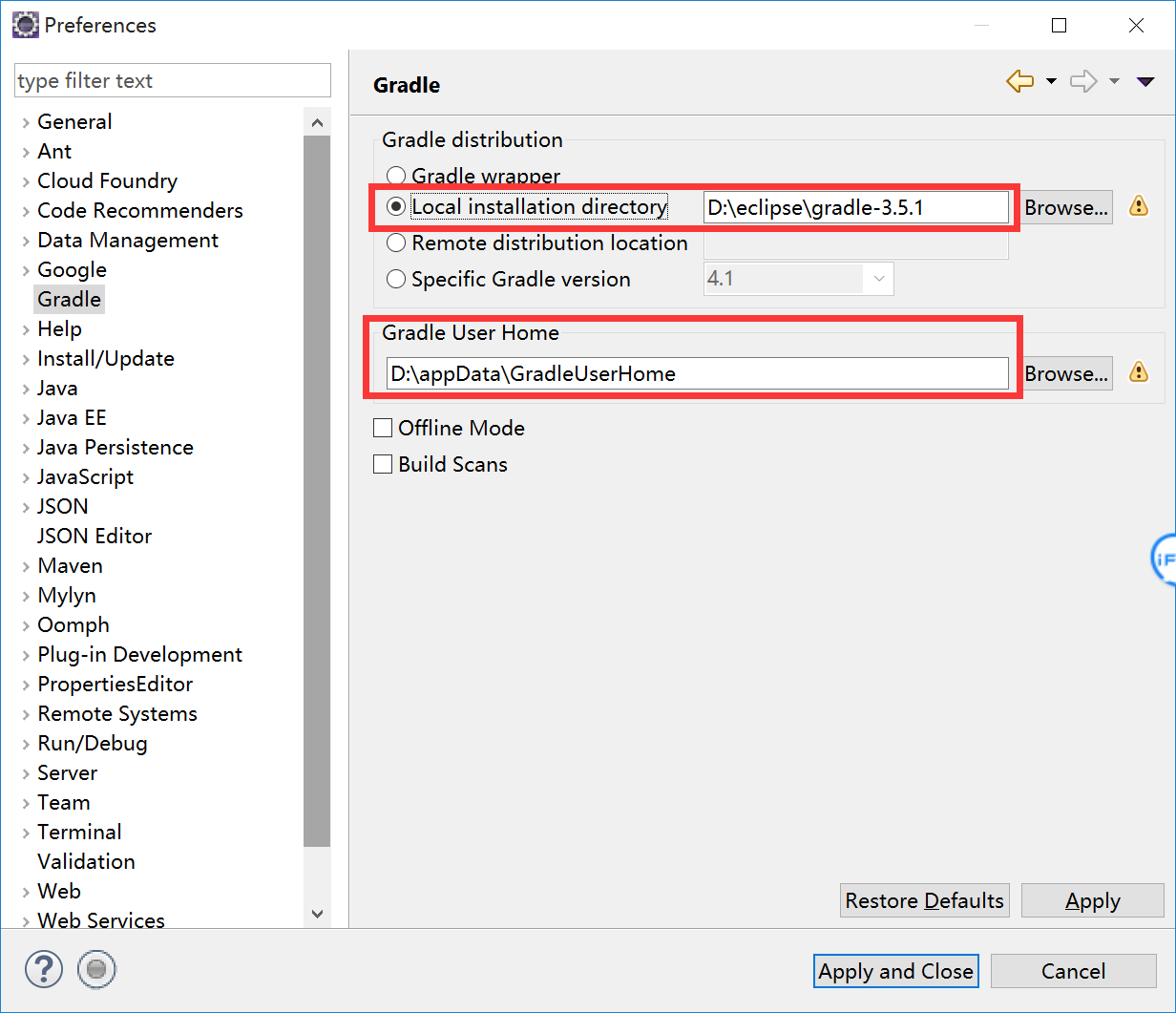
eclipse集成环境配置

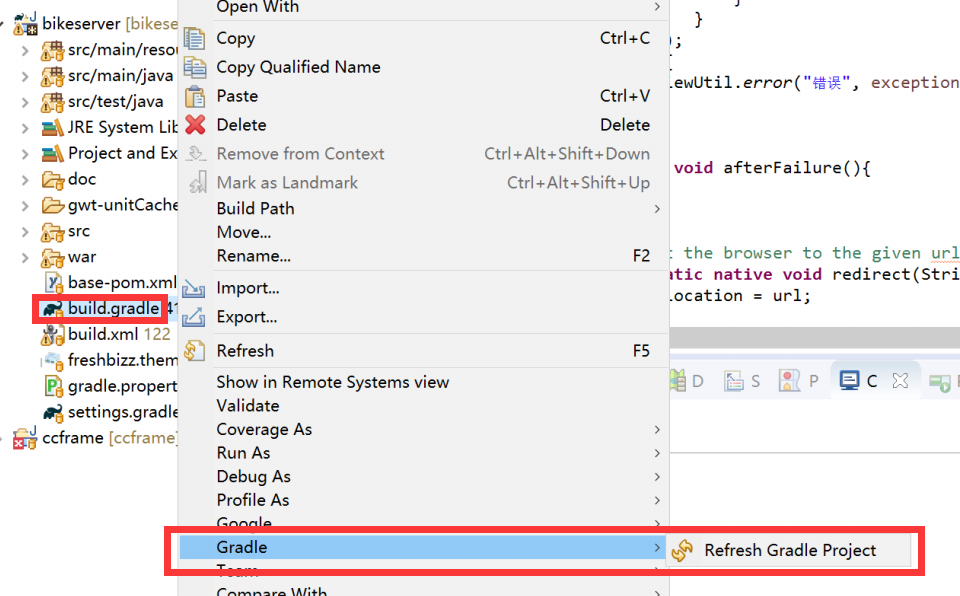
STEP1 解压eclipse-47-oxgen-all.7z

STEP2 配置gradle路径，指向eclipse下的gradle、并还可以指定gradle user home地址（用于存放从maven repository下载的jar）

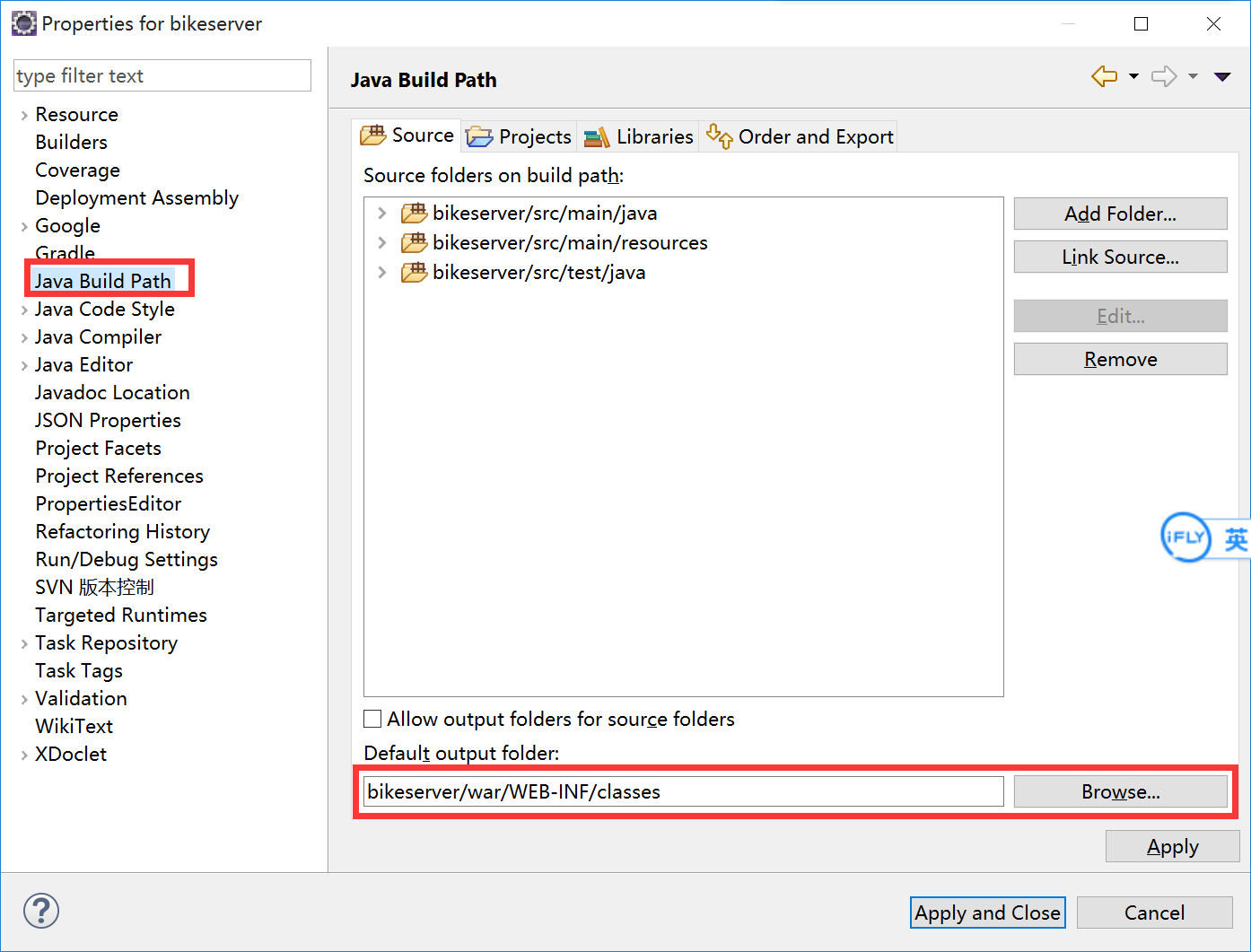


STEP3 连接SVN并checkout项目

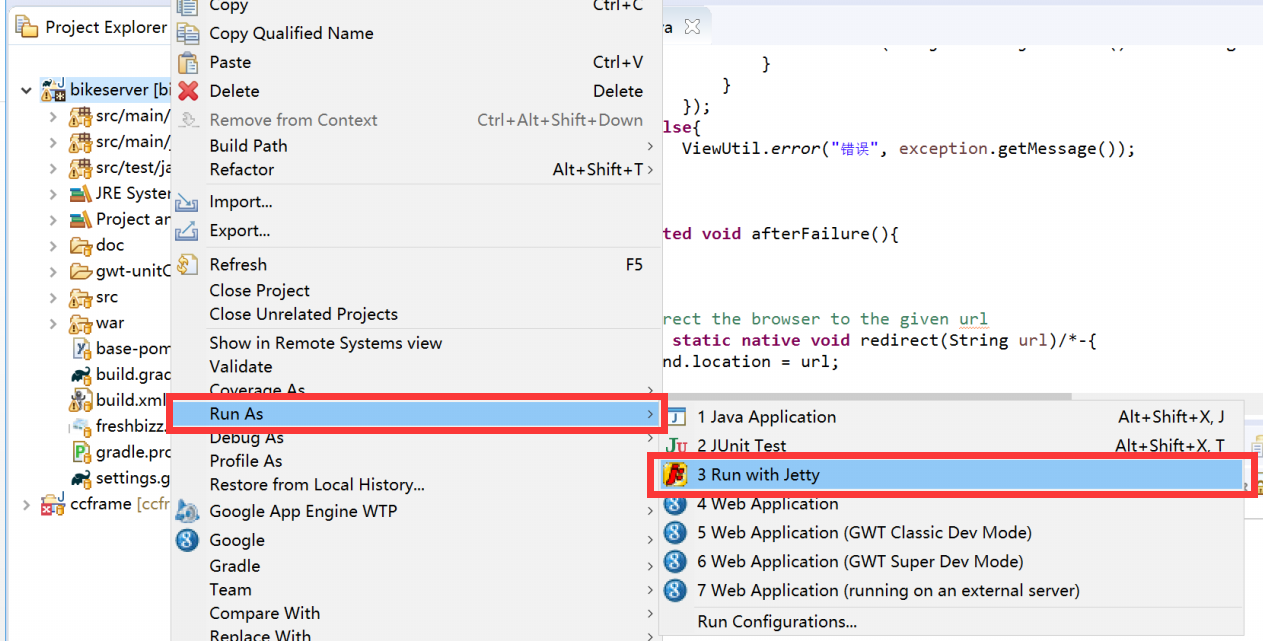
STEP4 在项目的build.gradle上右键选择gradle->refresh gradle project



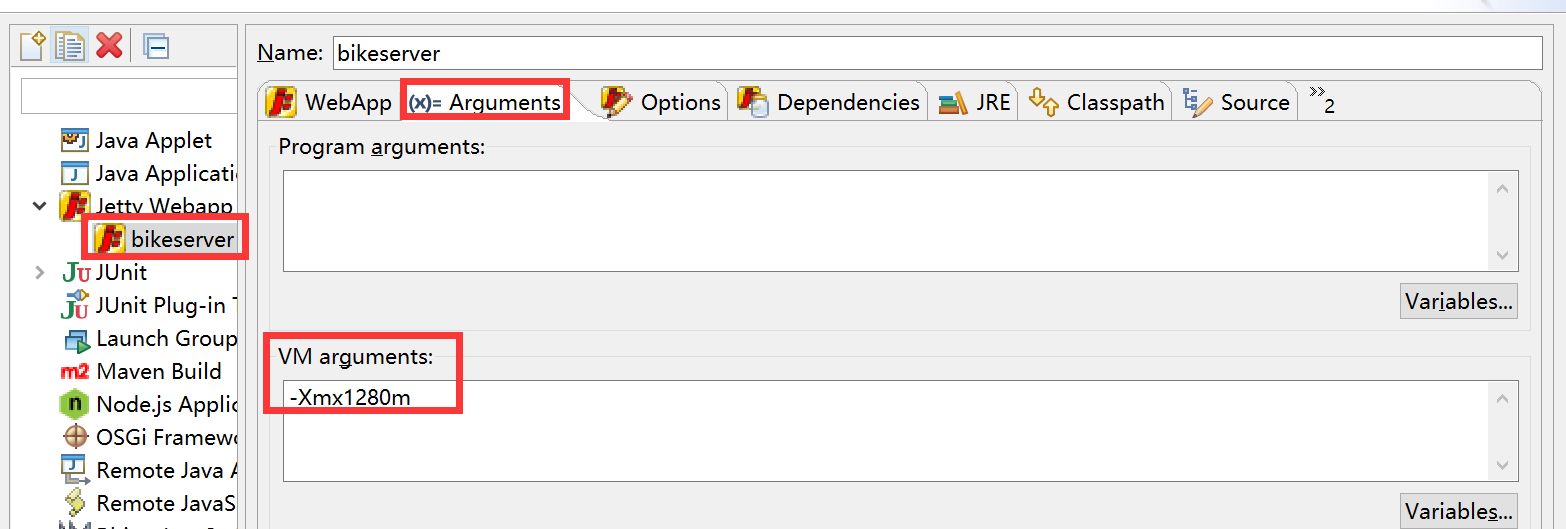
STEP5 等待所有的JAR下载完毕后，在项目的properties->Java Build Path将refresh导致的路径错误修正到项目war的位置

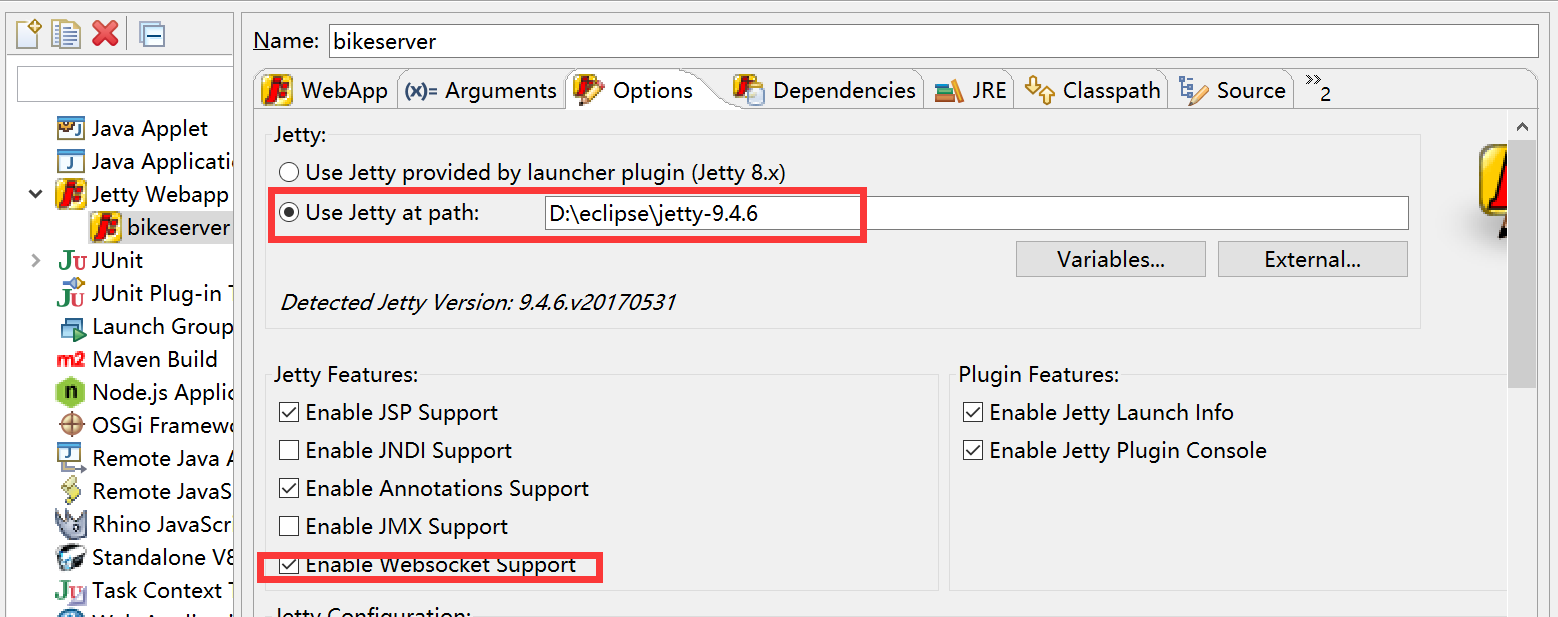


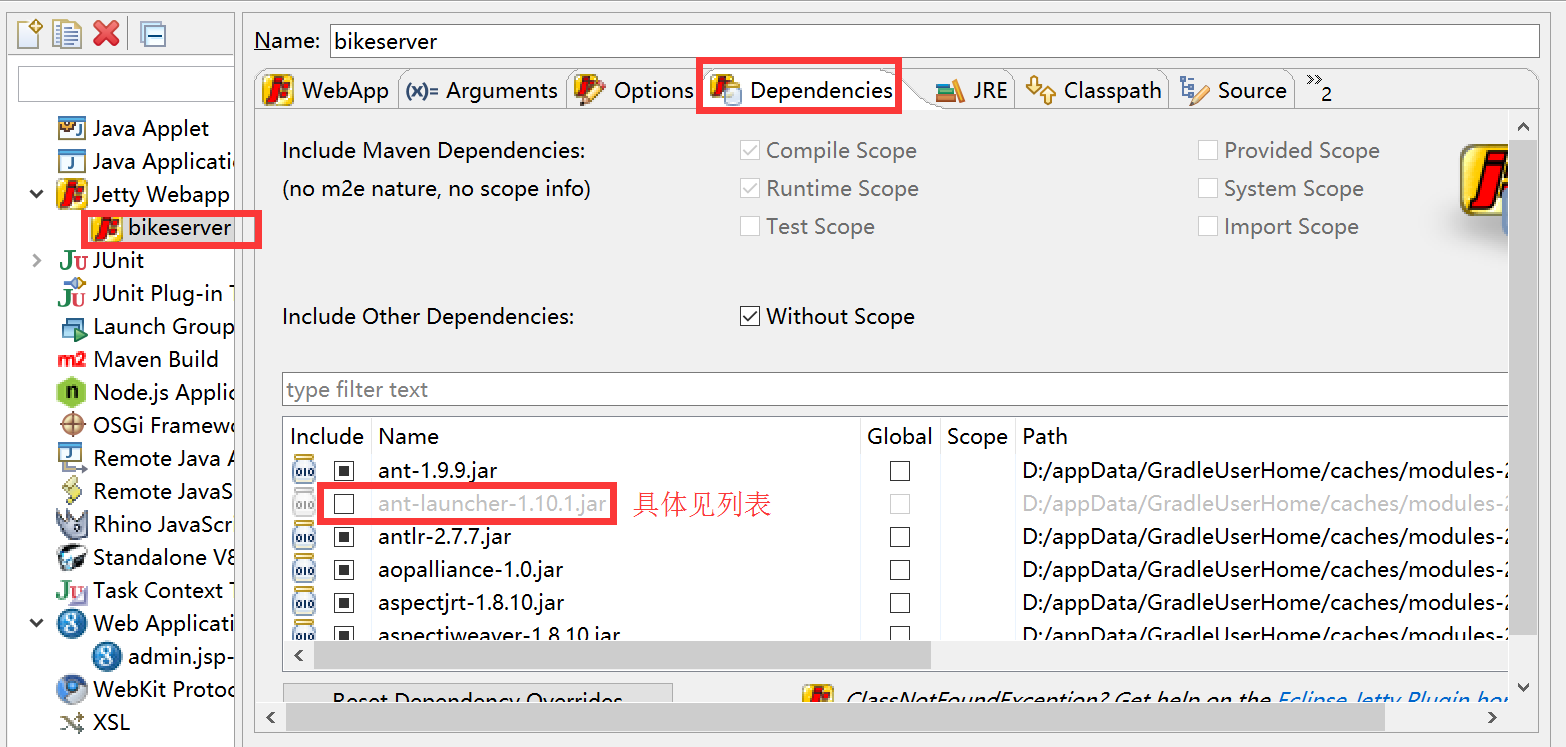
STEP6 配置项目运行的JETTY环境，先在项目上点击Run As -> Run With Jetty，启动后立即停止配置具体的运行项



STEP7 配置JETTY参数，内存，还有gradle里标记为providedCompile及testImplementation的jar







具体列表忽略以下勾选：

ant-launcher-1.10.1.jar

gin-2.1.2.jar

guice-3.0.jar

guice-assistendinject-3.0.jar

gwt-dev-2.6.1.jar

gwt-servlet-2.6.1.jar

gwt-servlet-deps-2.6.1.jar

gwttheme-freshbizz-1.0.jar

gwt-user-2.6.1.jar

gxt-3.1.1.jar

gxt-chart-3.1.1.jar

gxt-legacy-3.1.1.jar

junit-4.8.jar

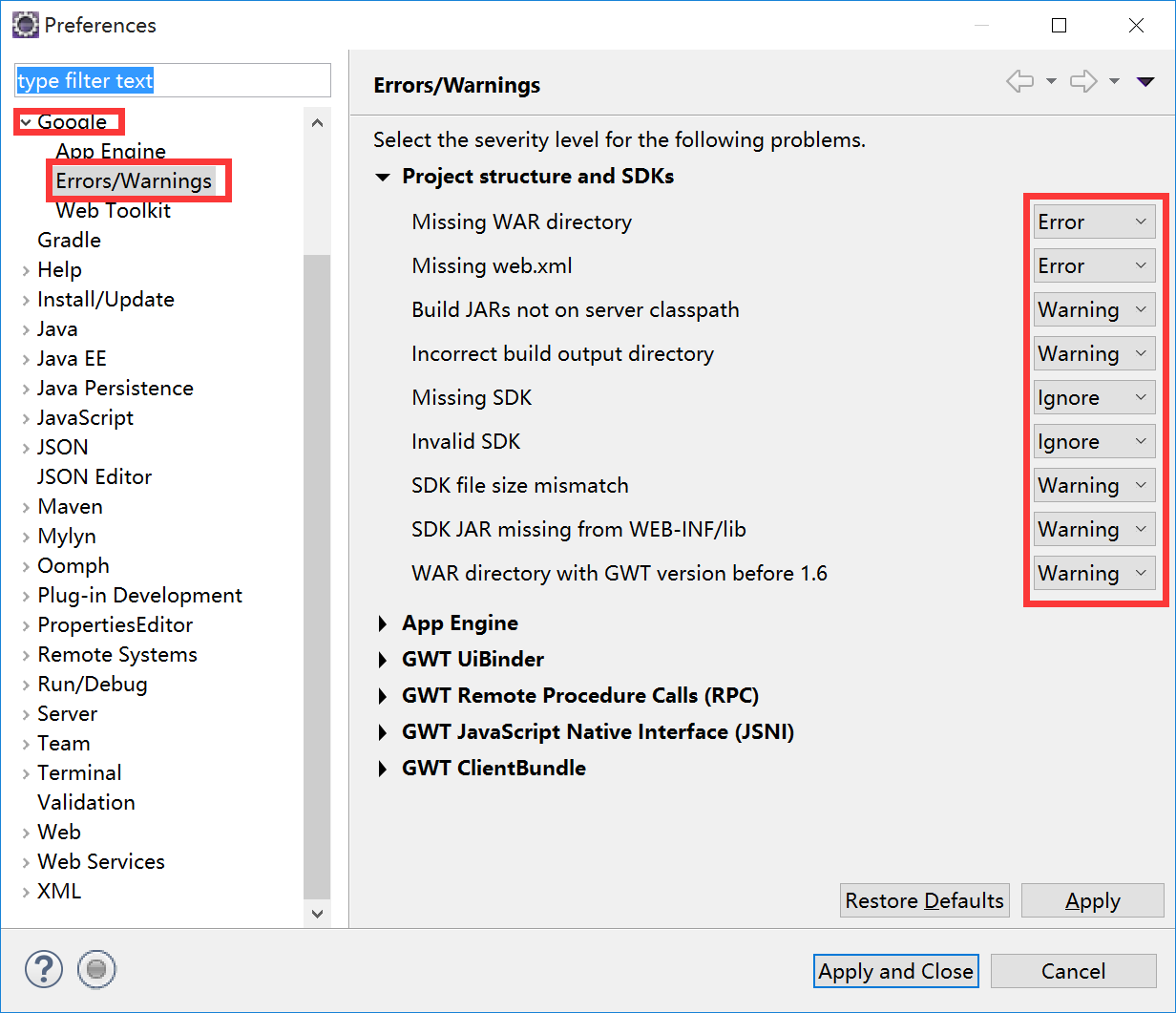
restygwt-2.1.1.jar

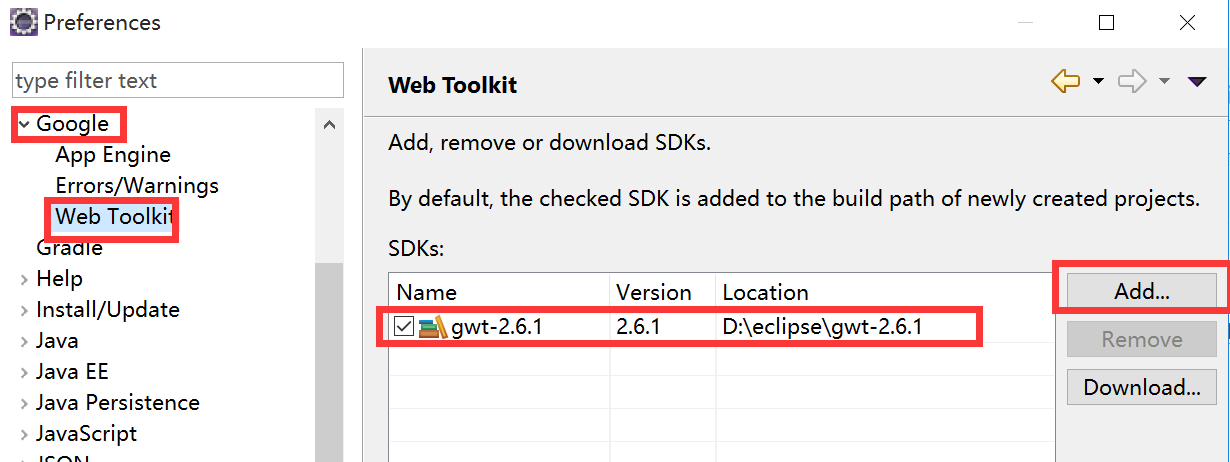
servlet-api-2.5.jar

validation-api-1.0.0.GA.jar

validation-api-1.0.0.GA-source.jar

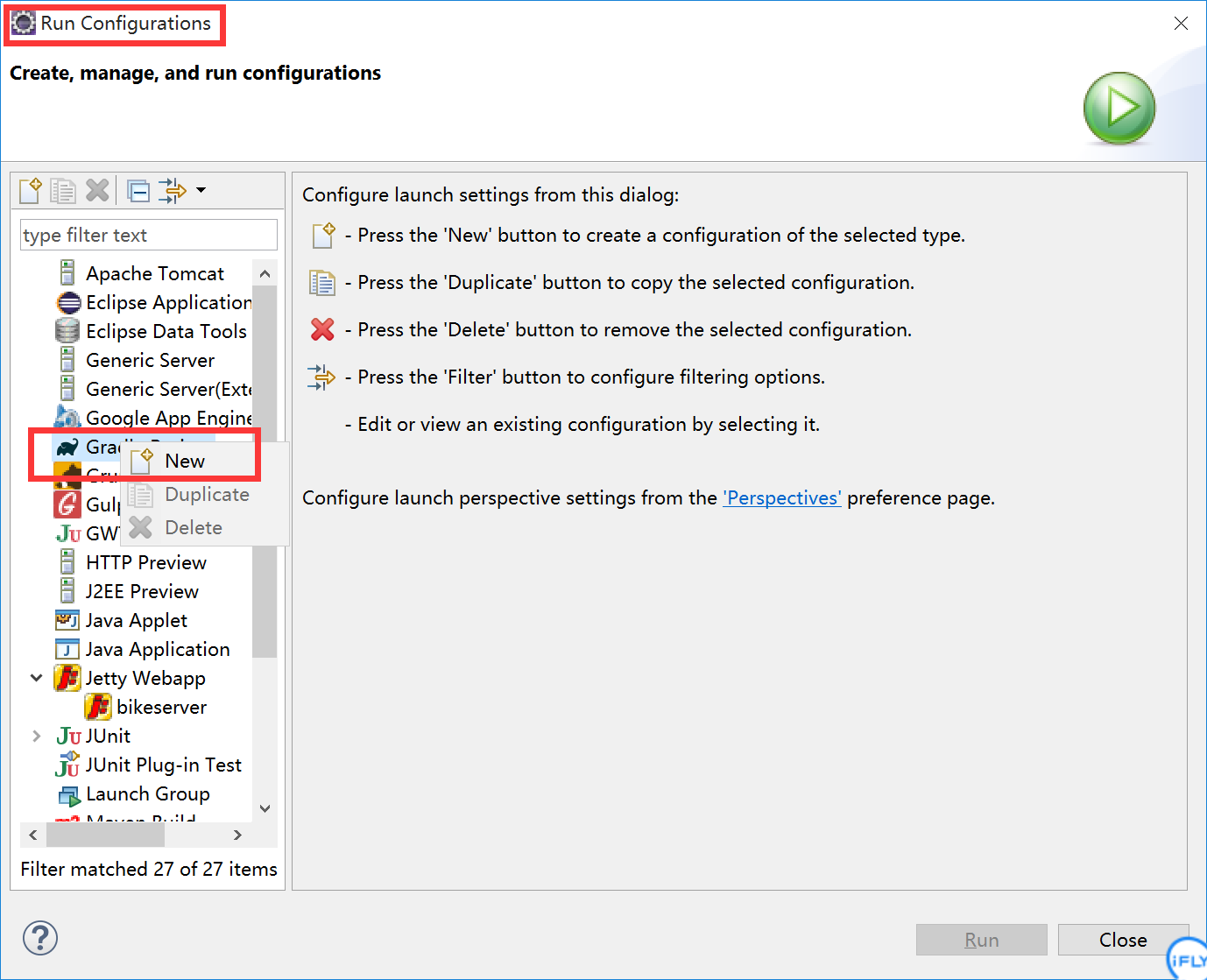
STEP8 配置GWT SDK的路径为eclipse目录下的GWT，以及设置GWT的忽略设置

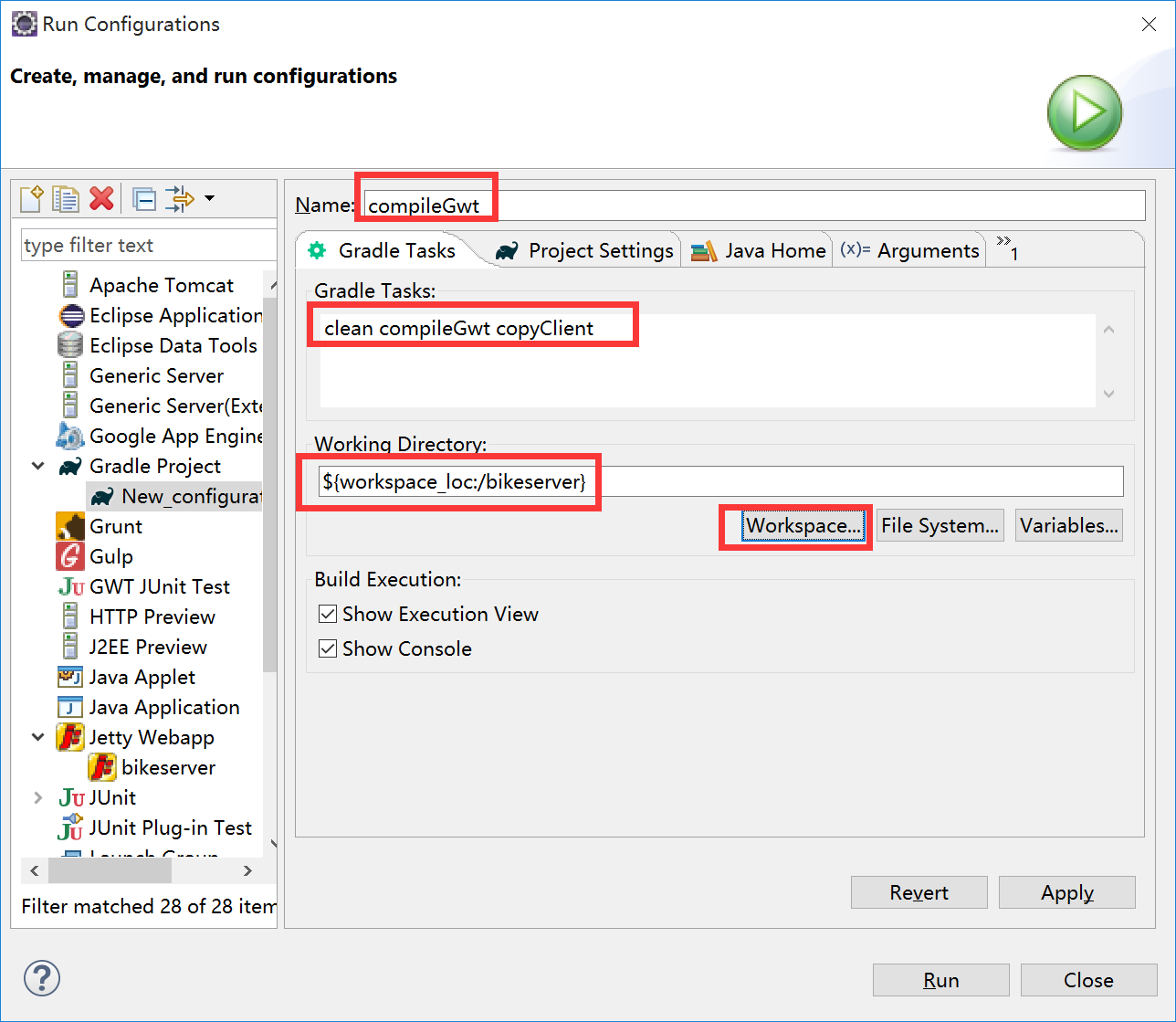


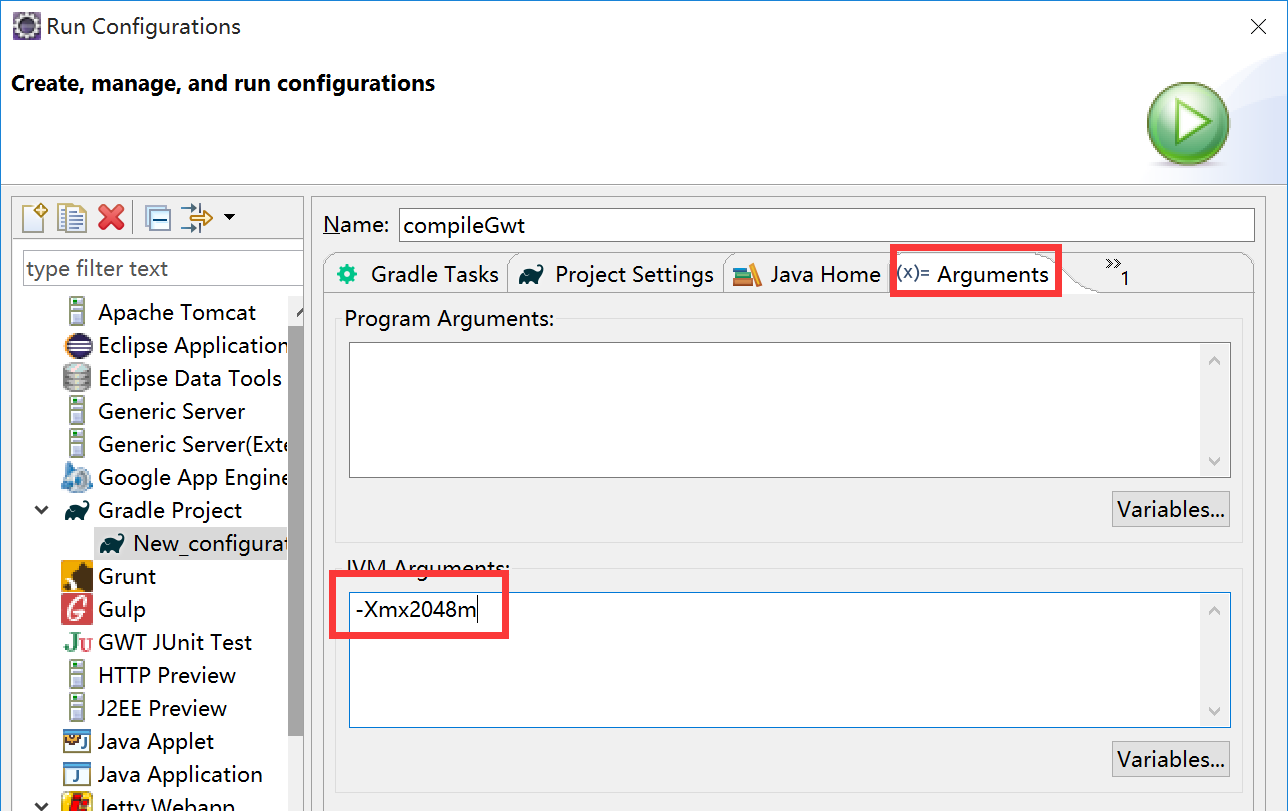


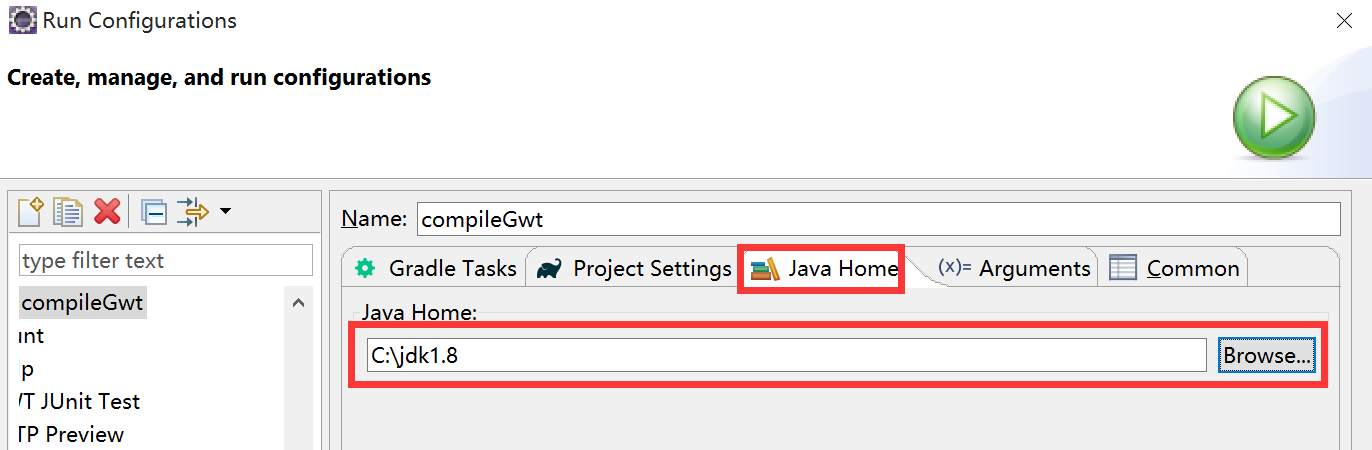
STEP9 配置GWT编译环境。

GWT编译不再用ant，而是要添加一个gradle的编译插件配置。先新建一个gradle gwt编译运行，然后配置好内存和运行参数以及JAVA\_HOME地址即可。









剩下的GWT debug的配置和GWT教程一致，在此不再赘述。

注意：

1. 更新依赖的snapshot jar需要refresh gradle project;
2. refresh gradle project后，项目的Default output folder会重新变为bin目录，需要手动更改为项目war/WEB-INF/classes