IDEA 使用全解

IDEA 简介

Intellij IDEA 是 JetBrains 公司研发的又一款伟大的 IDE 工具，主要面向广大的 Java 开发人员。为什么说是“又一款伟大的 IDE 工具”呢？，因为该公司已经推出过 PHPStorm、PyCharm、WebStorm 等优秀的开发工具。JetBrains 是一家捷克的软件开发公司，该公司位于捷克的布拉格，并在俄国的圣彼得堡及美国麻州波士顿都设有办公室。根据国内的使用习惯，在本 Chat 中，我们将使用 IDEA 作为 Intellij IDEA 的简称。对于 IDEA，JetBrans 公司提出的最新口号是“Capable and Ergomomic IDE for JVM”，翻译过来就是面向 JVM、功能强大且符合人体工程学的 IDE。面向 JVM 的是指该 IDE 支持 Java、Scale 等运行于 JVM 之上的所有语言。

IDEA 在业界被公认为最好的 Java 开发工具之一，尤其在智能代码助手、代码自动提示、重构、 J2EE 支持、各类版本工具（ Git、 SVN、 Github 等）、 JUnit、 CVS 整合、代码分析、 创新的 GUI 设计等方面的功能可以说是超常的。目前，该软件有两个主要的版本，即免费的社区版和付费的旗舰版。免费版只支持 Java 等少数语言和基本的 IDE 特性；旗舰版还支持 HTML、CSS、PHP、MySQL、Python 等语言和更多的工具特性，详情请见下面两张图。

安装插件后支持	SQL类	基本JVM
PHP	PostgreSQL	Java
Python	MySQL	Groovy
Ruby	Oracle	
Scala	SQL Server	
Kotlin		
Clojure		

支持的框架	额外支持的语言代码提示	支持的容器
Spring MVC	HTML5	Tomcat
GWT	CSS3	TomEE
Vaadin	SASS	WebLogic
Play	LESS	JBoss
Grails	JavaScript	Jetty
Web Services	CoffeeScript	WebSphere
JSF	Node.js	
Struts	ActionScript	
Hibernate		
Flex		

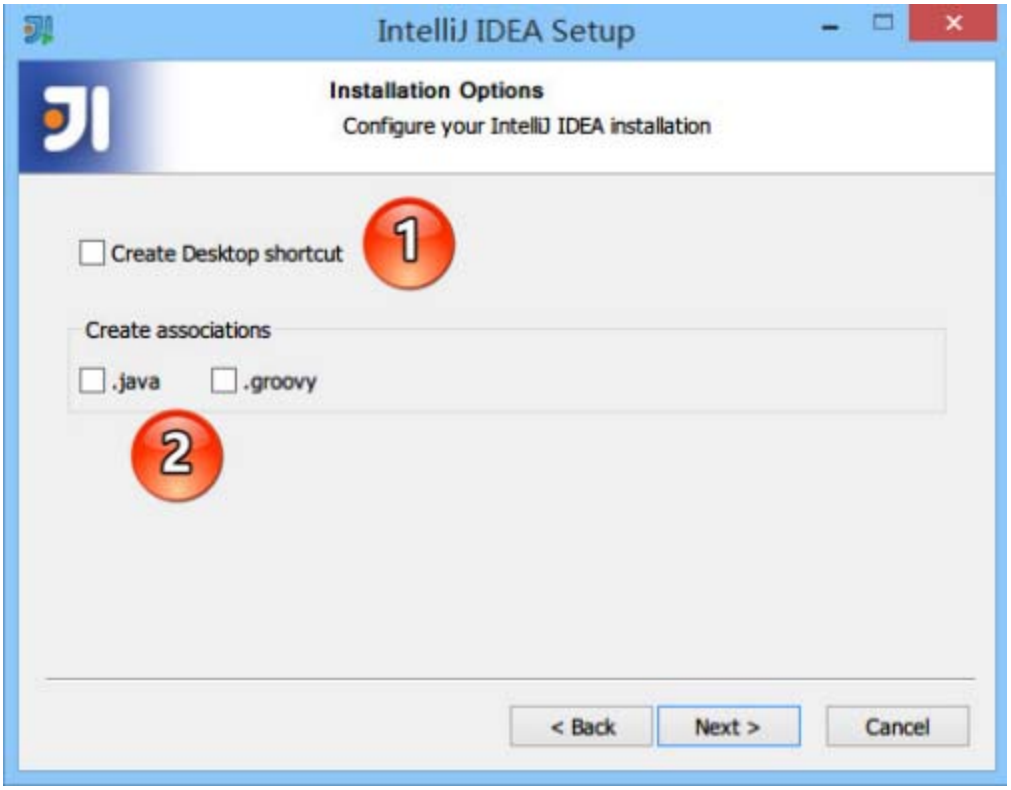
安装与配置

具体的安装步骤不在此赘述，需要的小伙伴可以参看这里。

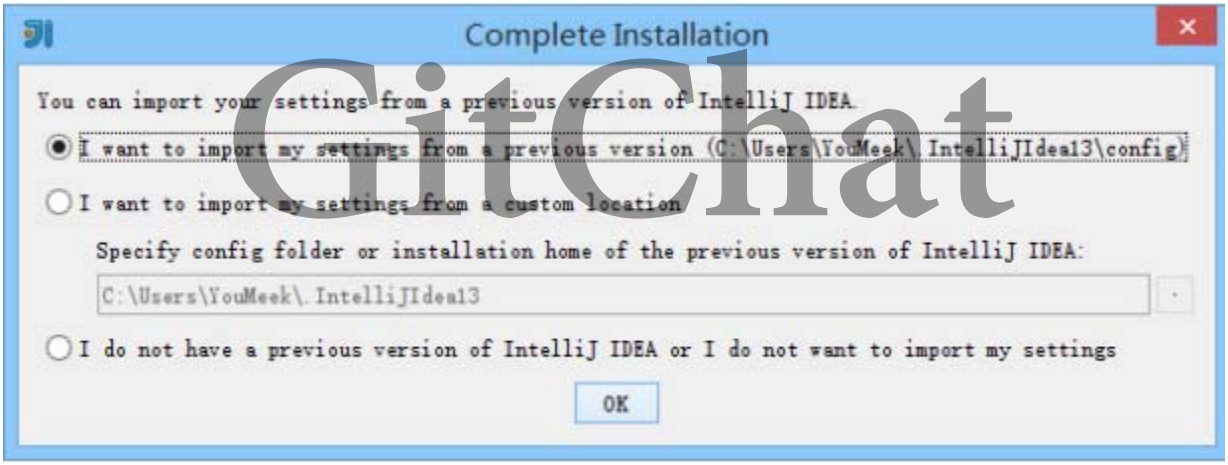
请注意以下两点：

1. 在桌面创建快捷方式。
2. 创建关联，不建议勾选。

如下图所示。

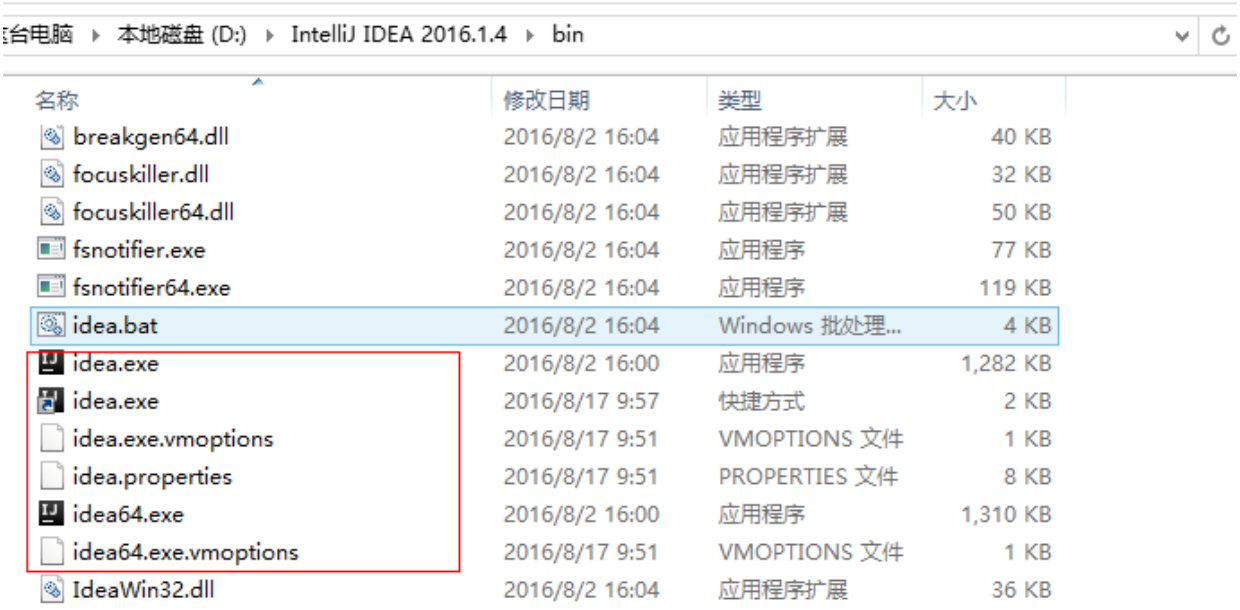


首次运行，如下图所示。



- 上图第一个单选表示选择默认路径 IntelliJ IDEA 旧配置；
- 上图第二个单选表示选择其他目录的 IntelliJ IDEA 配置；
- 上图第三个单选表示生成新的 IntelliJ IDEA 配置。

配置文件如下图所示。

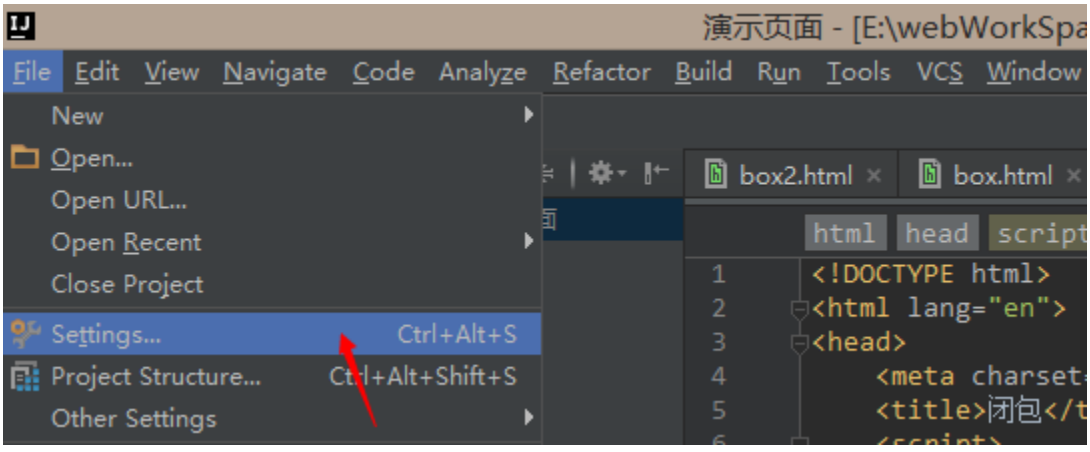


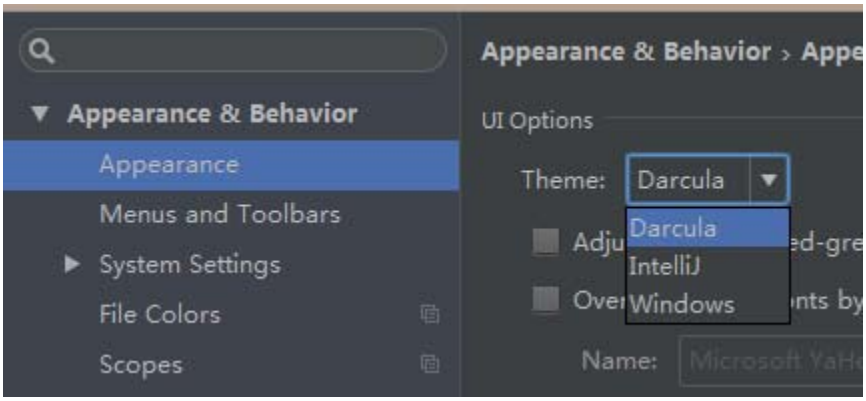
其中：

- idea.exe.vmoptions 文件是 IntelliJ IDEA 32 位的可执行文件的 VM 配置文件；
- idea64.exe.vmoptions 文件是 IntelliJ IDEA 64 位的可执行文件的 VM 配置文件，**通过适当修改里面参数，可以加快 idea 速度**；
- idea.properties 文件是 IntelliJ IDEA 的一些属性配置文件。

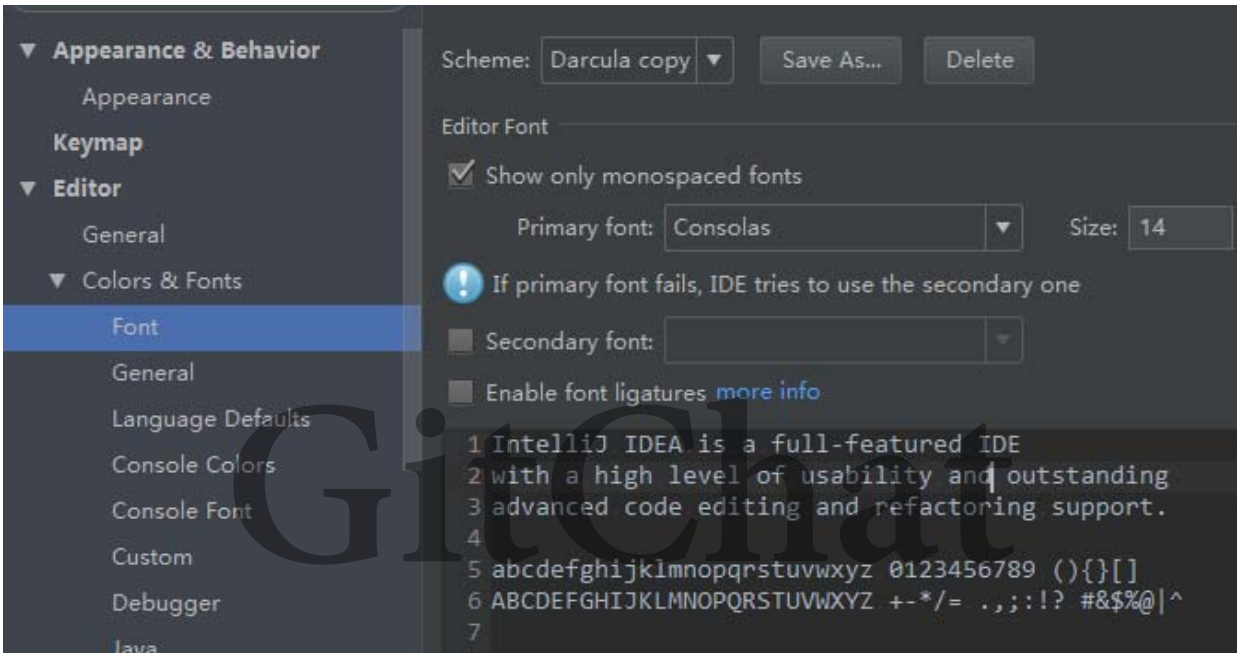
主题、字体、文件编码修改

在 Windows 系统上 IntelliJ IDEA 默认提供的主题有三套，即 Darcula 、 IntelliJ 、 Windows。除了 Darcula 是黑色主题，其他都是以白色为背景的，设置过程如下图所示。



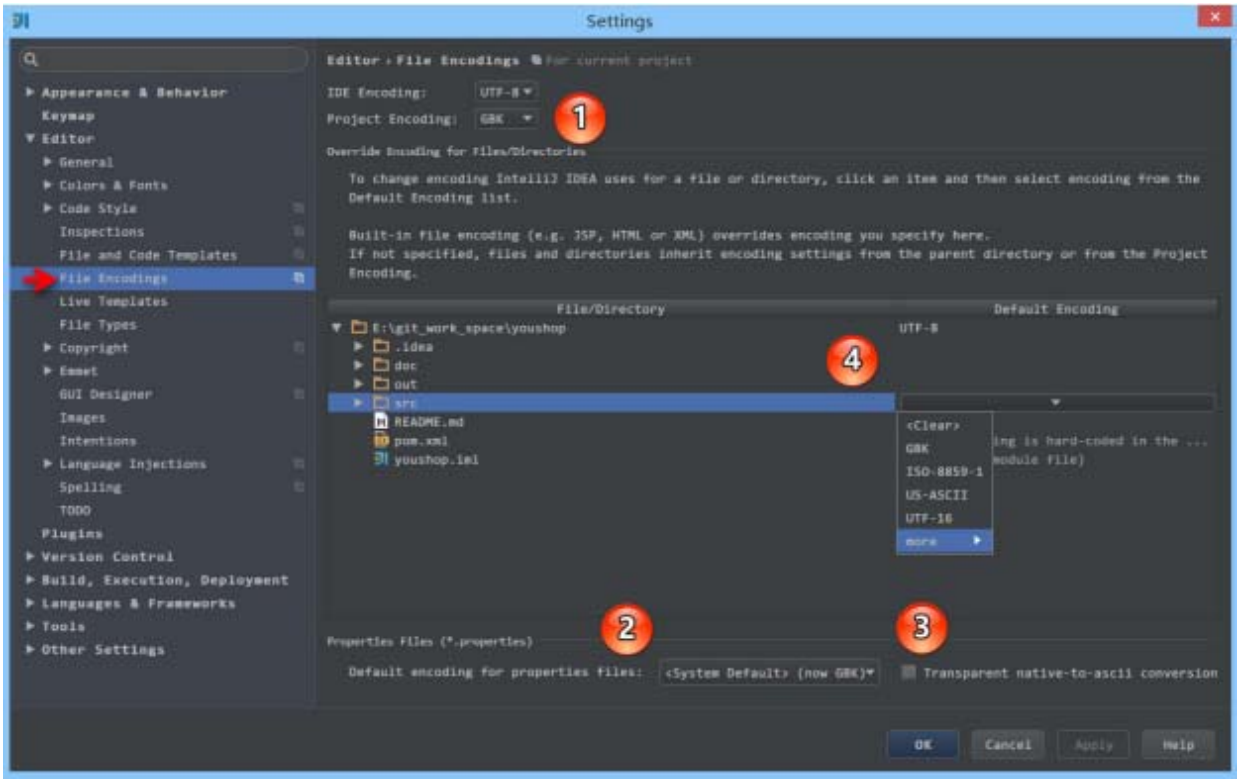


注意：默认 IntelliJ IDEA 是不能直接在默认的代码模板上修改字体的，需要先 Save As 一份出来，然后才可以修改，如下图所示。



文件编码的修改

可以按照下图的提示，对文章编码进行修改。



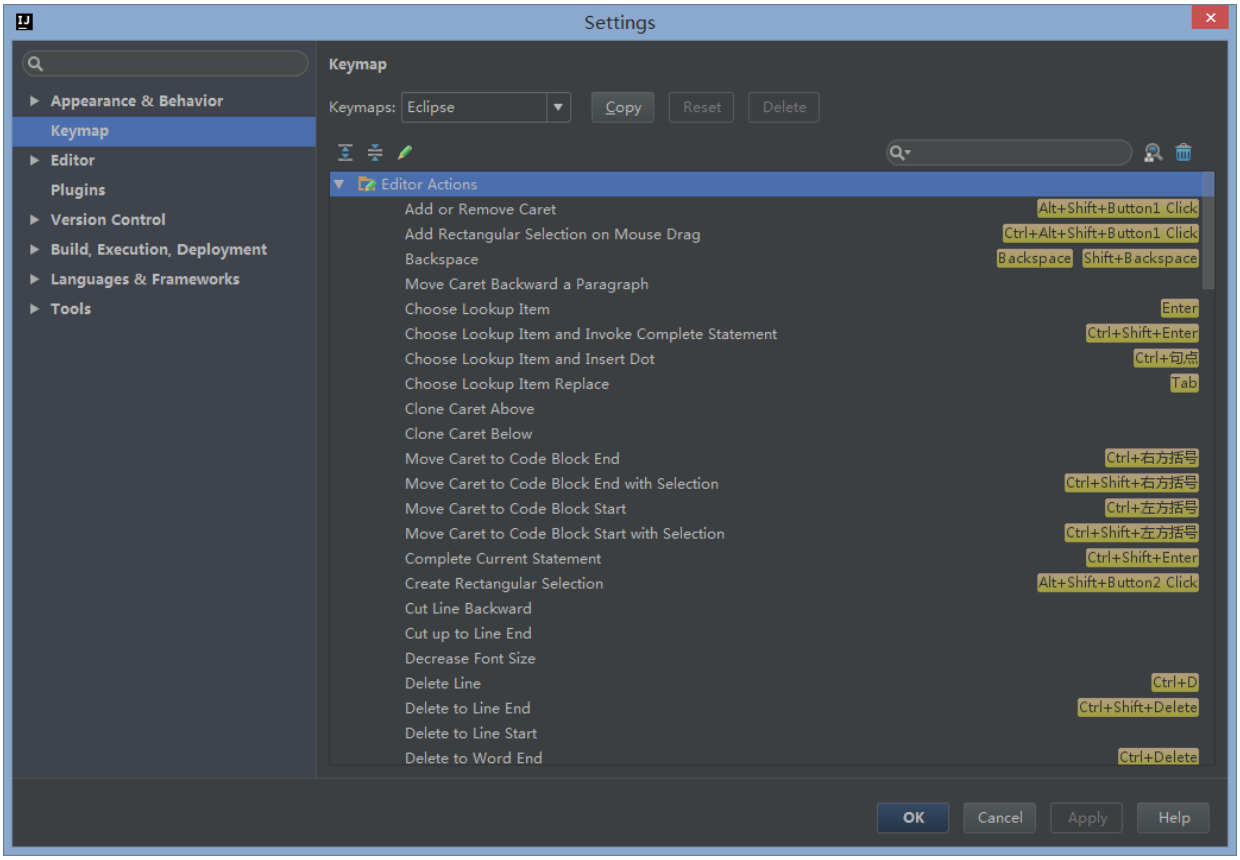
图中 ① 处设置项目编码，② 处设置 properties文件编码，③ 处的 Transparent native-to-ascii conversion 为重要属性，主要用于转换 ASCII，一般都要勾选，不然 properties 文件中的注释显示的都不会是中文。④ 处表示 IntelliJ IDEA 除了支持对整个 Project 设置编码之外，还支持对目录、文件进行编码设置。

注意：对单独文件的编码修改还可以点击右下角的编码设置区。如果代码内容中包含中文，则会弹出演示中的如下两种操作选择。

• Reload：表示不会改变文件和内容的编码格式，而是将 IDE 本身的解码格式改为新的编码格式；
• Convert：表示直接将文件格式和IDE的解码格式都一起改变。

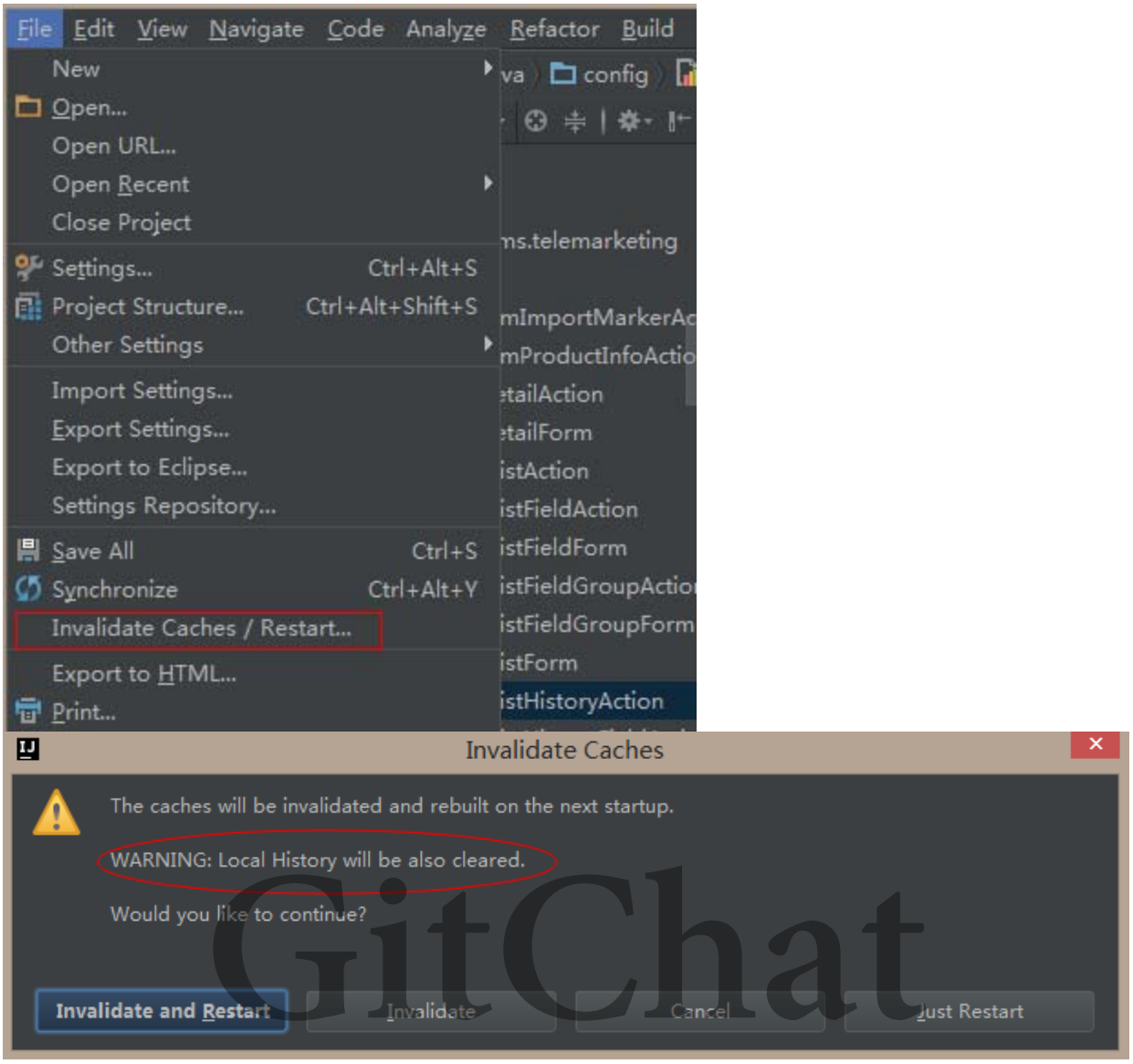
快捷键设置

我们还可以对快捷键进行设置，可选择 Eclipse 风格，并对具体快捷键进行修改，如下图所示。



缓存和索引清理

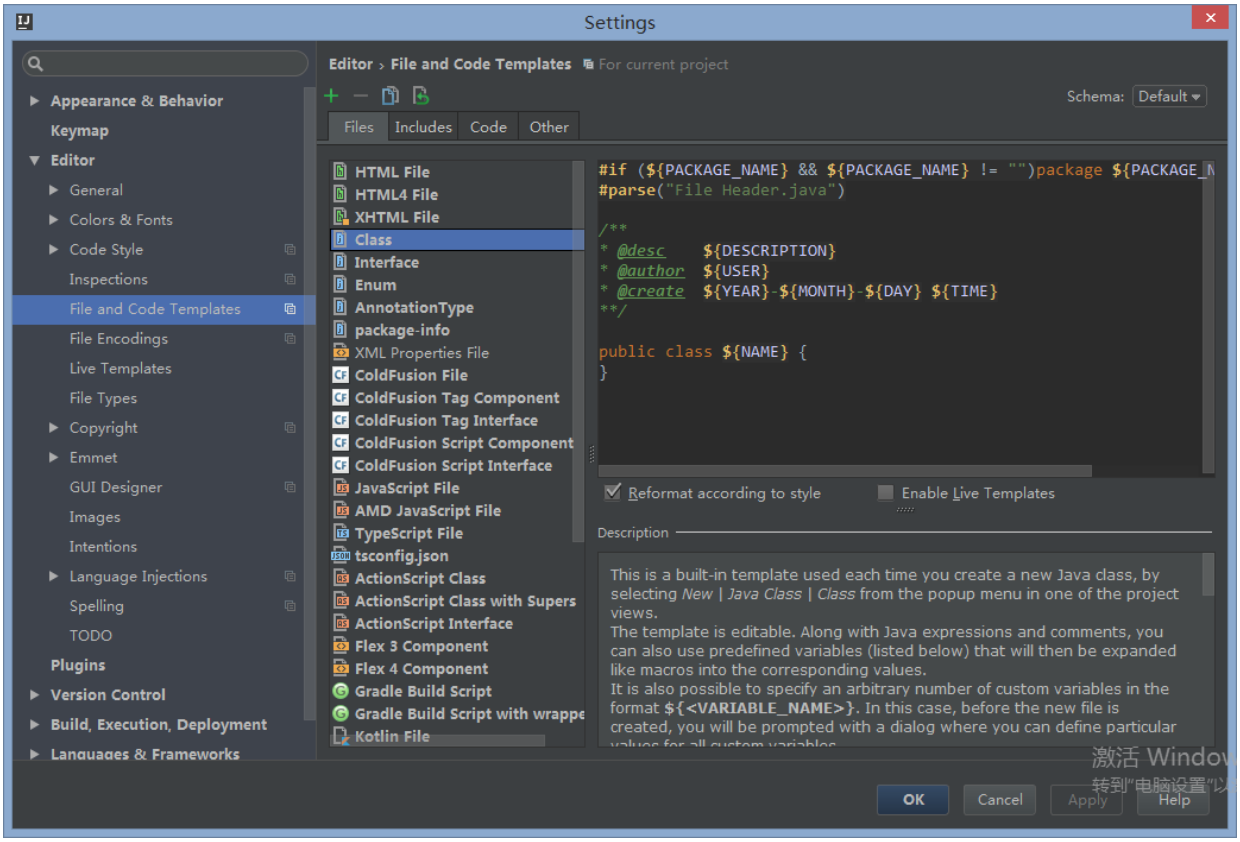
IntelliJ IDEA 的缓存和索引主要用来加快文件查询，从而加快各种查找、代码提示等操作的速度，当某些特殊条件下，IntelliJ IDEA 的缓存和索引文件也是会损坏的，比如断电、蓝屏引起的强制关机。当你重新打开 IntelliJ IDEA，基本上 IntelliJ IDEA 都会报各种莫名其妙错误，甚至项目打不开，IntelliJ IDEA 主题还原成默认状态。这很有可能是 IntelliJ IDEA 缓存和索引出了问题，这时可以清除缓存、索引，如下图所示。



通过上图方式清除缓存、索引本质也只是删除 C 盘下的 system 目录下的对应文件而已，如果你不用上述方法也可以删除整个 system 。当 IntelliJ IDEA 再次启动项目的时候会重新创建新的 system 目录以及对应项目缓存和索引（目录地址在： C:\Users\当前登录的系统用户名.IntelliJIdea16\system ）。如果你遇到了因为索引、缓存坏了以至于项目打不开，那也建议你可以直接删除 system 目录，一般这样都可以很好地解决你的问题。

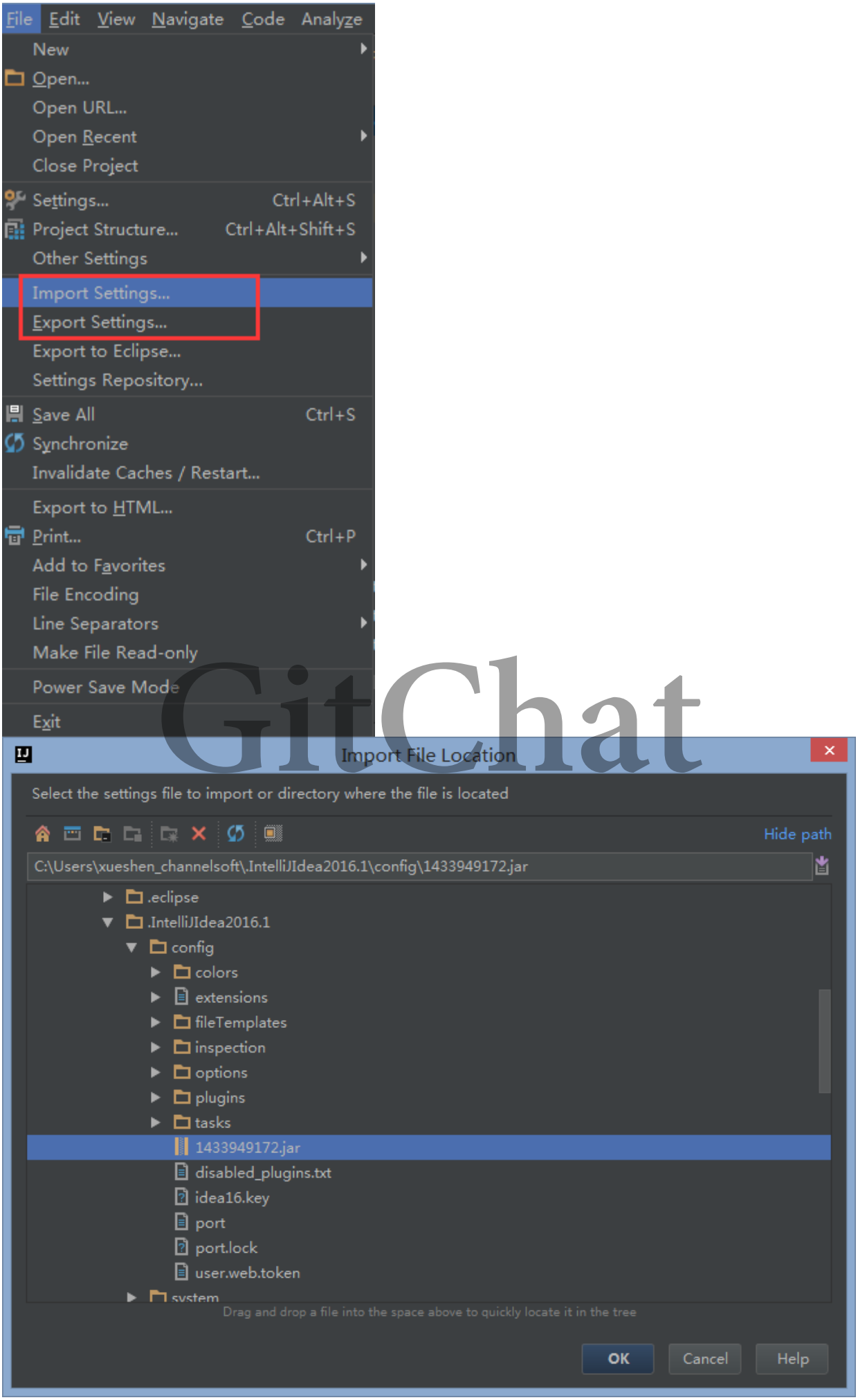
设置模板

File -> Settings -> Editor -> Code Style -> File and Code Templates，可以按照该流程进行操作，如下图所示。



导入/导出基本配置

操作流程为：File –> Import Settings –> 选择基本配置 jar 包。如下图所


File Edit View Navigate Code Analyze
New
Open...
Open URL...
Open Recent
Close Project
Settings... Ctrl+Alt+S
Project Structure... Ctrl+Alt+Shift+S
Other Settings
Import Settings...
Export Settings...
Export to Eclipse...
Settings Repository...
Save All Ctrl+S
Synchronize
Invalidate Caches / Restart...
Export to HTML...
Print... Ctrl+P
Add to Favorites
File Encoding
Line Separators
Make File Read-only
Power Save Mode
Exit
Import File Location
Select the settings file to import or directory where the file is located
Hide path
C:\Users\xueshen_channelsoft\.IntelliJIdea2016.1\config\1433949172.jar
.eclipse
.IntelliJIdea2016.1
config
colors
extensions
fileTemplates
inspection
options
plugins
tasks
1433949172.jar
disabled_plugins.txt
idea16.key
port
port.lock
user.web.token
system
Drag and drop a file into the space above to quickly locate it in the tree
OK
Cancel
Help

大家可以点击这里下载模板。

然后选择 File -> settings -> Colors & Fonts，设置 Scheme 为新导入的主题即可。

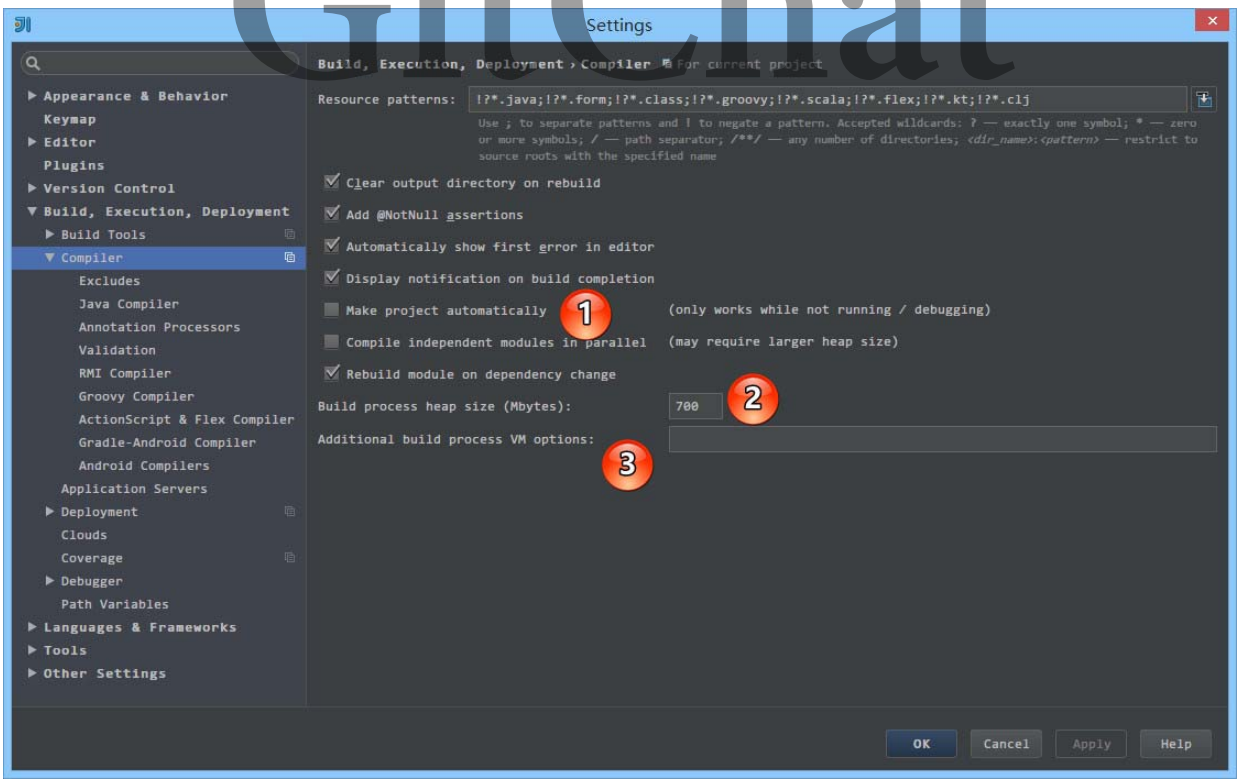
设置背景图片

按下 Ctrl+Shift+A，你会看到弹出了一个对话框，在里面输入一些操作的名字可以跳转到相应的操作界面（你可以通过这个方法找到你猜测 IntelliJ 可能会有的黑科技），输入 Set Background Image，弹出一个窗口，根据窗口的提示进行操作即可。还有实时预览。

编译方式

在 IntelliJ IDEA 里，编译方式一共有如下三种。

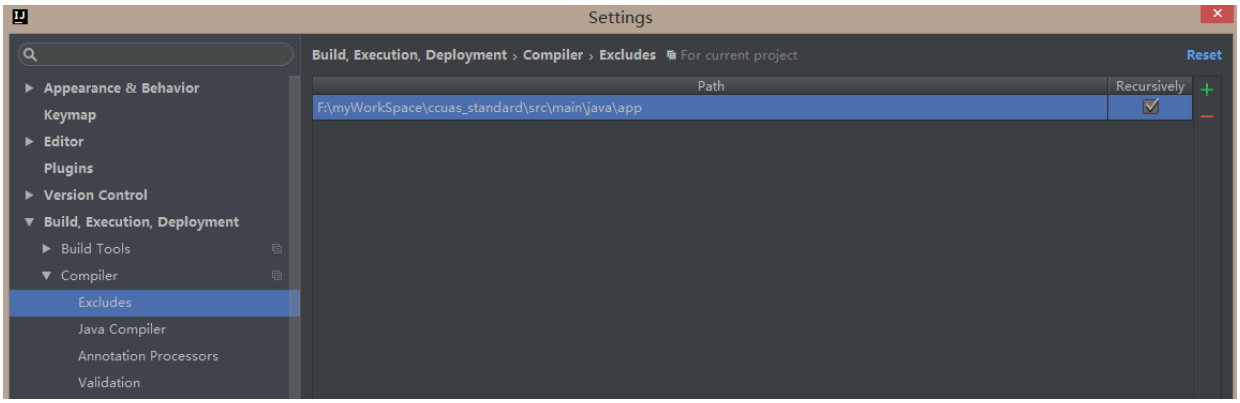
- Compile：对选定的目标（Java 类文件），进行强制性编译，不管目标是否是被修改过。
- Rebuild：对选定的目标（Project），进行强制性编译，不管目标是否被修改过。由于 Rebuild 的目标只有 Project，所以 Rebuild 每次花的时间会比较长。
- Make：使用最多的编译操作。对选定的目标（Project 或 Module）进行编译，但只编译有修改过的文件，没有修改过的文件不会编译，这样平时开发大型项目才不会在编译过程中浪费时间。



如上图所示，IntelliJ IDEA 是支持自动编译的，默认是不开启的，也建议不用开启。设置编译 heap 大小，默认是 700，建议使用 64 位的用户，在内存足够的情况下，建议改为 1500 或以上。如果你在编译的时候出错，报：OutOfMemoryError，一般也是要来改这个地方。还可以设置编译时的 VM 参数，这个你可以根据需求进行设置，一般用不上。

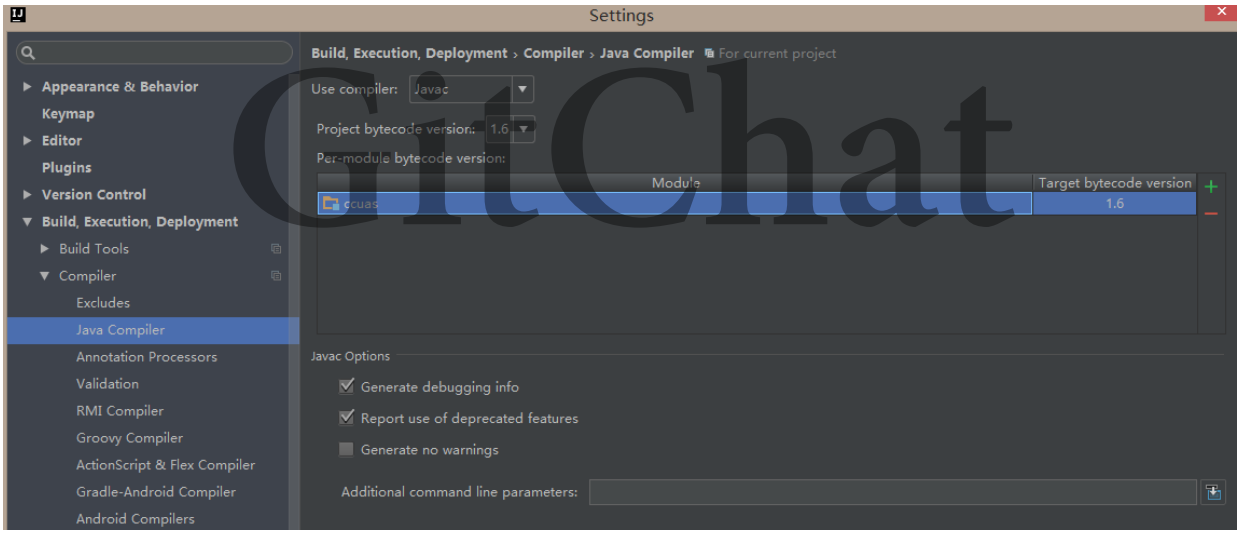
编译器

在项目中，如果有任何一个可编译的文件无法编译通过，则 IntelliJ IDEA 是无法运行起来的，必须等你解决全部问题，编译通过之后才可运行。但在开发过程中，可能某一个包目录的文件编译无法通过，但是我们又不急着改，那就可以考虑把该包加入到排除编译列表中，则项目就可以运行起来。设置过程，请见下面这张图。



Java 编译器

Java 编译器设置方法如下图所示。



其中有两点需要注意下：

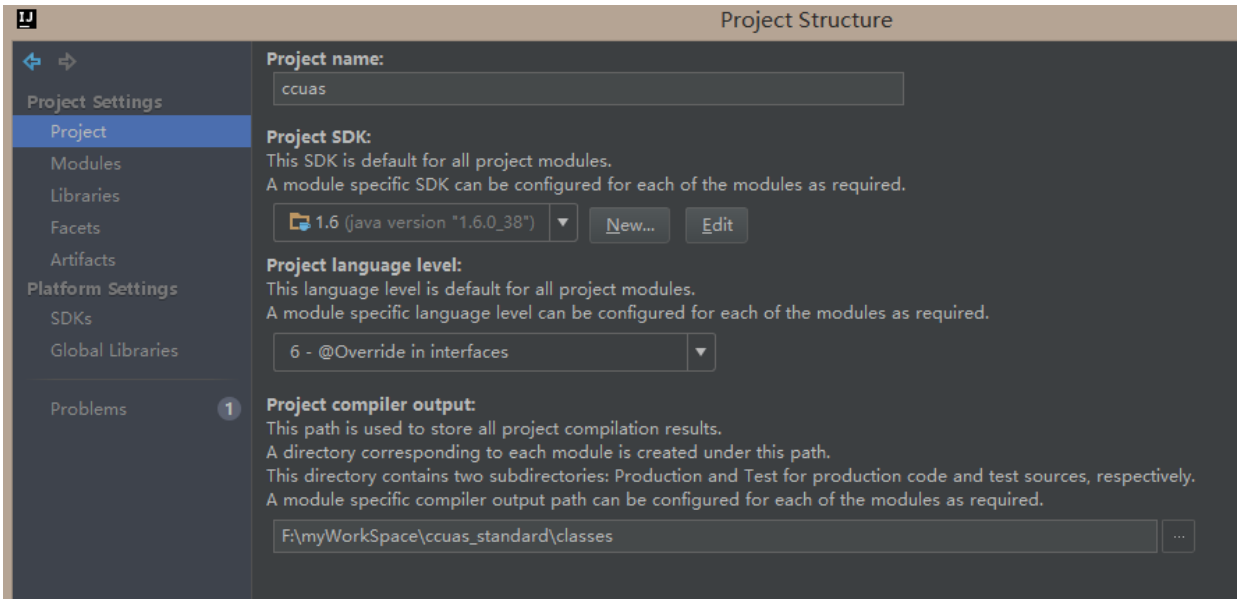
• Project bytecode version：针对项目字节码编译版本，一般选择的是当前项目主 JDK 的版本。
• Per-module bytecode version：可以针对 Project 下各个 Module 的特殊需求单独设置不同的 bytecode version，前提是电脑上必须安装对应的 JDK 版本。

项目构建

在 IntelliJ IDEA 中 Project 是最顶级的级别，次级别是 Module。

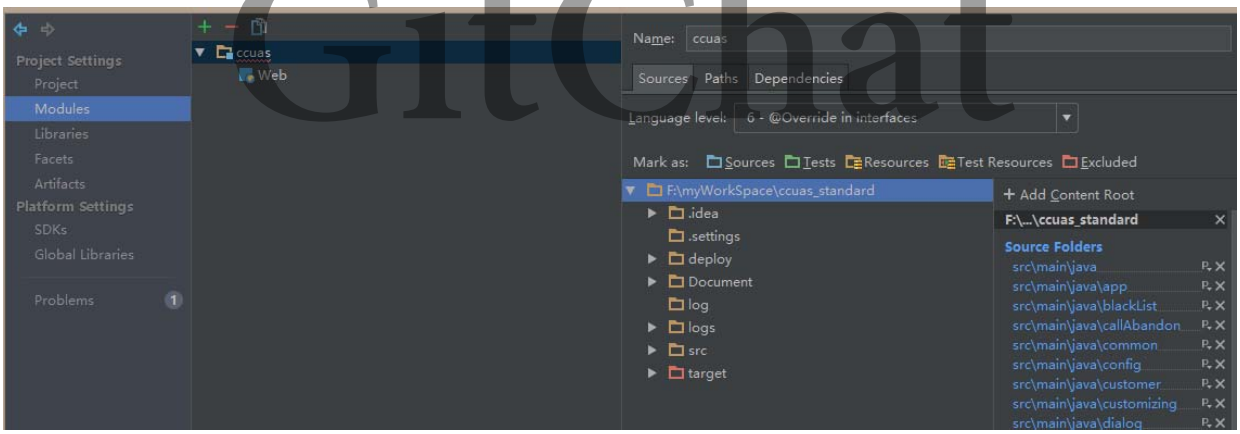
一个 Project可以有多个 Module。模块之间尽量是处在同一个项目业务的的情况下，彼此之间互相依赖关联。

Project Structure 如下图所示。

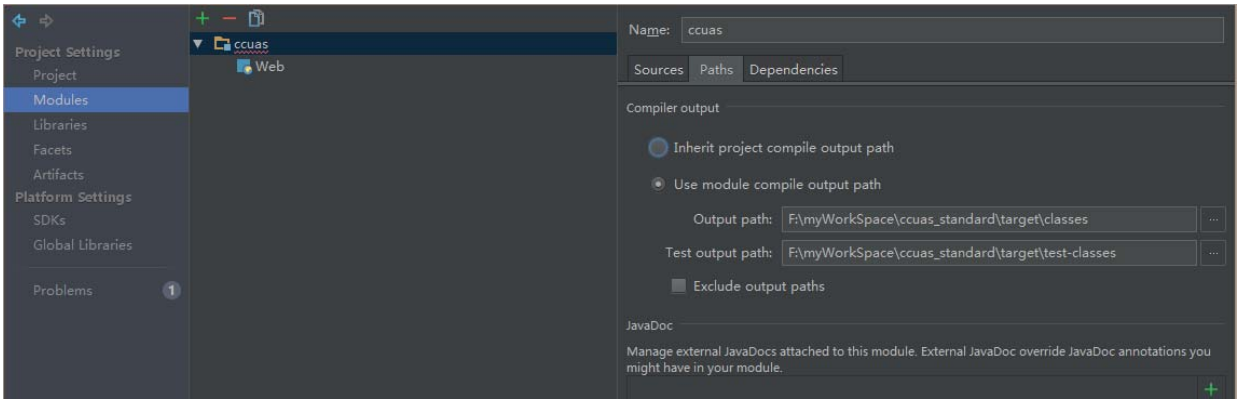


我们为整个 Project 选择 JDK 1.6作为SDK。

配置 Module 的 source，如下图所示。



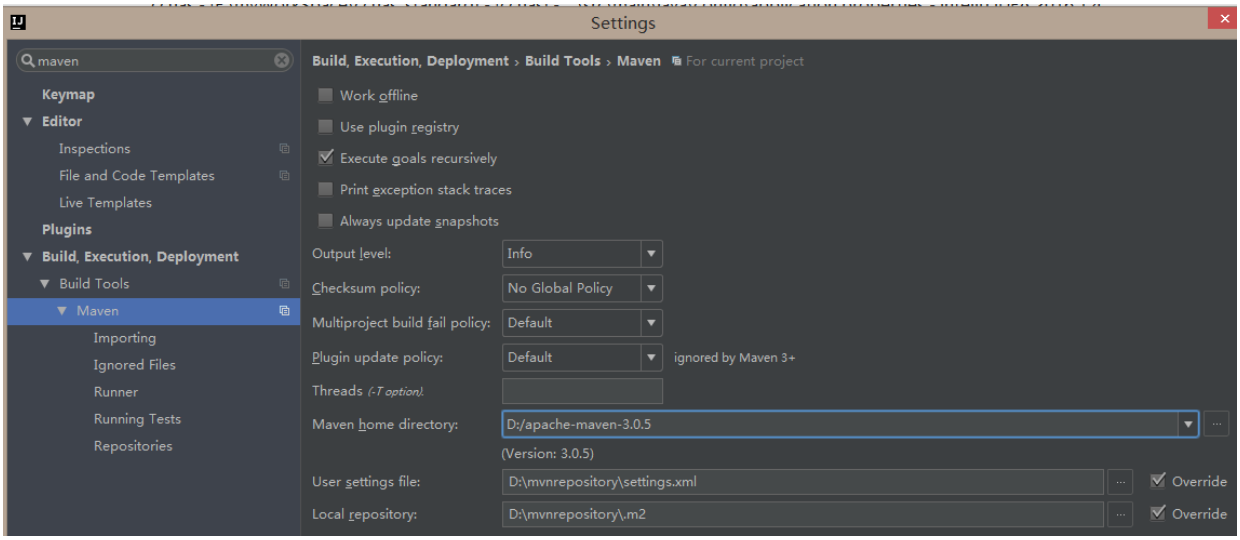
配置 Module 的 Path，如下图所示。



注意：如果使用 Resin 作为容器需要把 output path 改为 webapp/WEB-INF/classes。

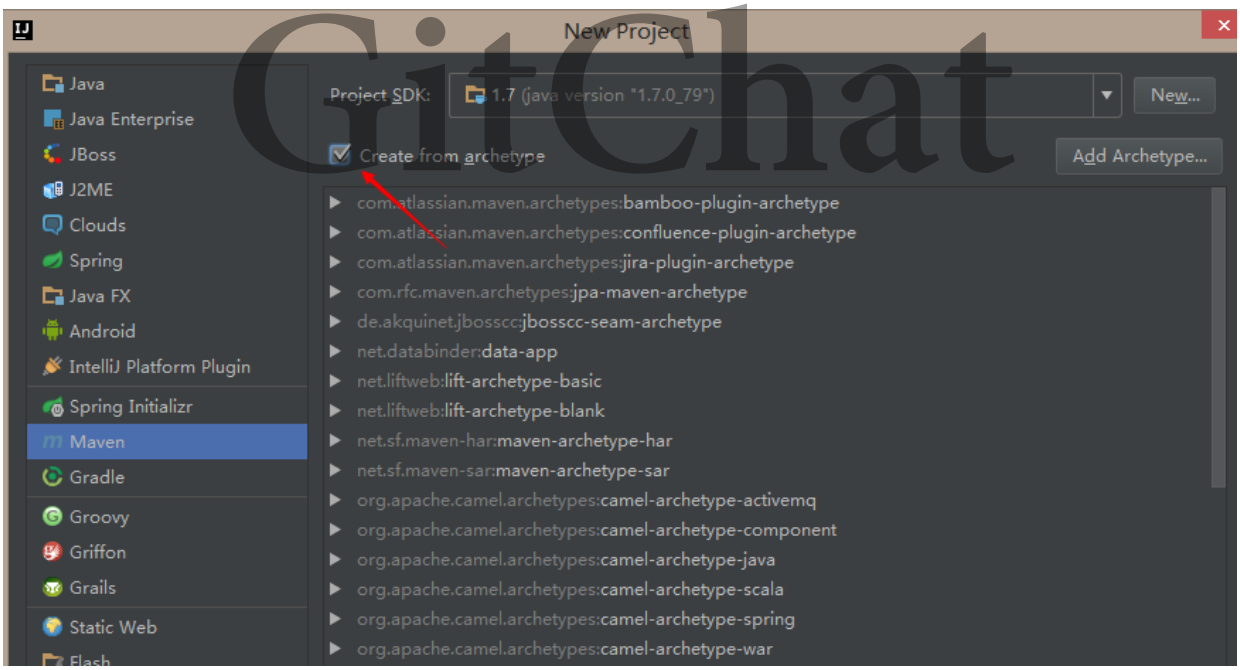
Maven 项目介绍

Maven 的常用设置，如下图所示。



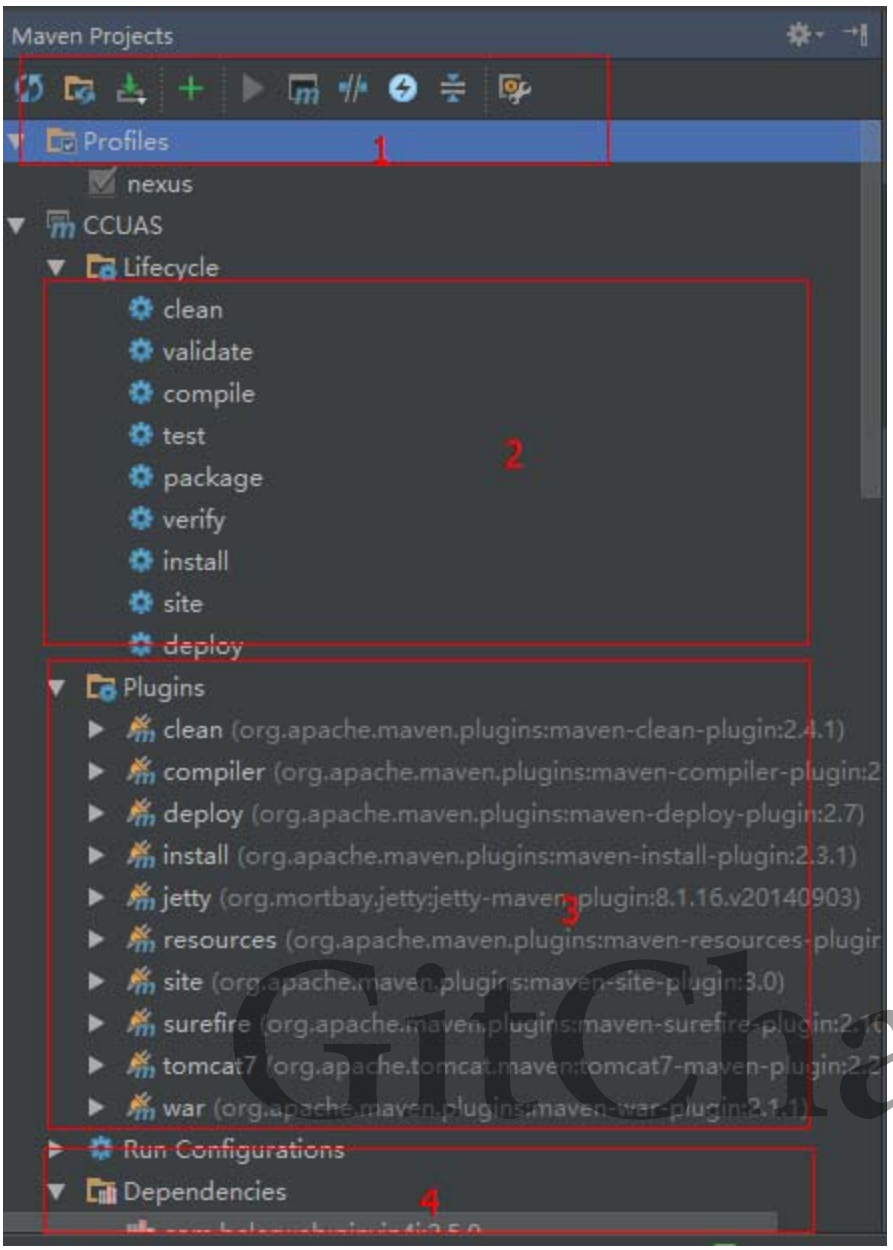
（1）配置 Maven home 与 setting file

Maven骨架如下图所示

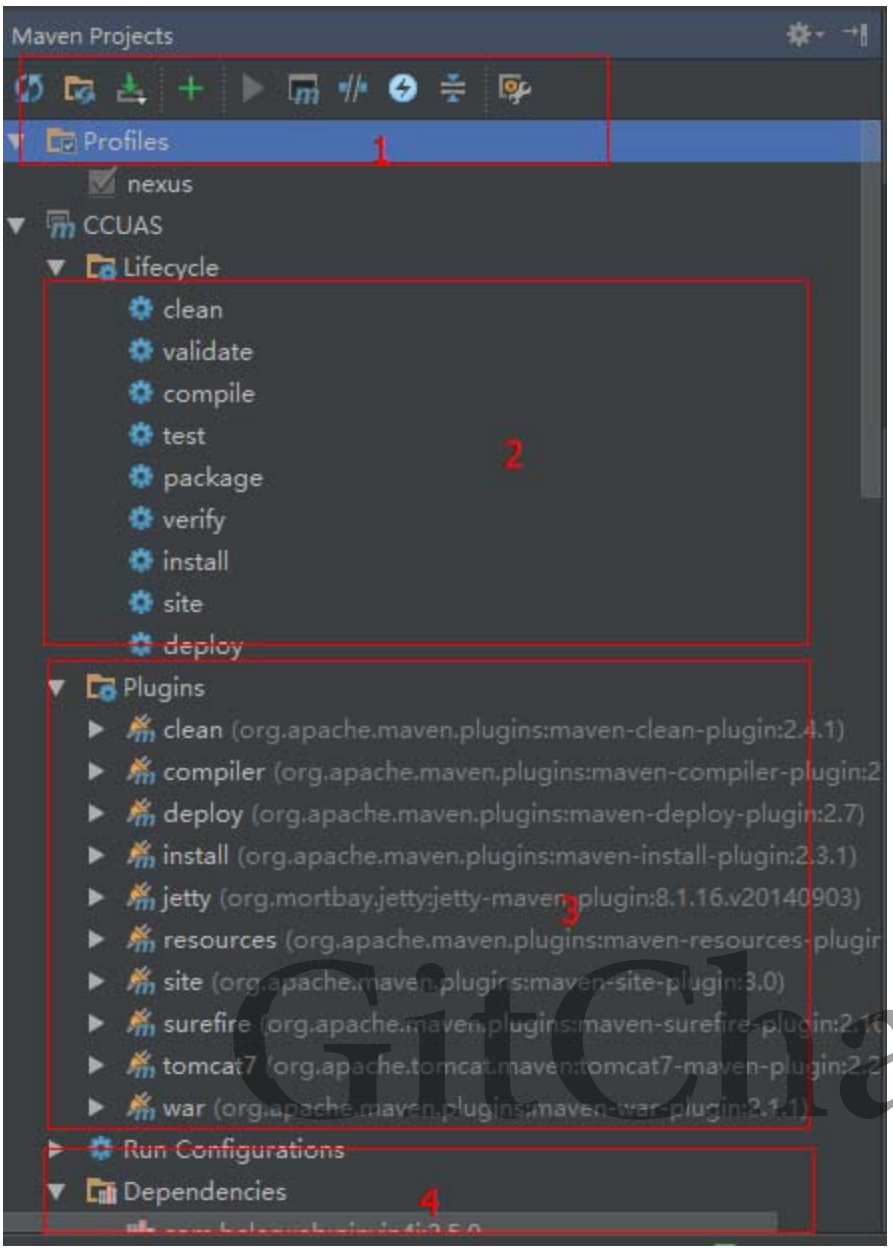


根据已有的 Maven 骨架进行 Java Web 项目创建。其中需要特别注意的是，在创建项目过程中 Maven 会去外网中央仓库中下载对应的依赖或是组件，这个过程根据自身网络环境决定其快慢。

Maven组件管理



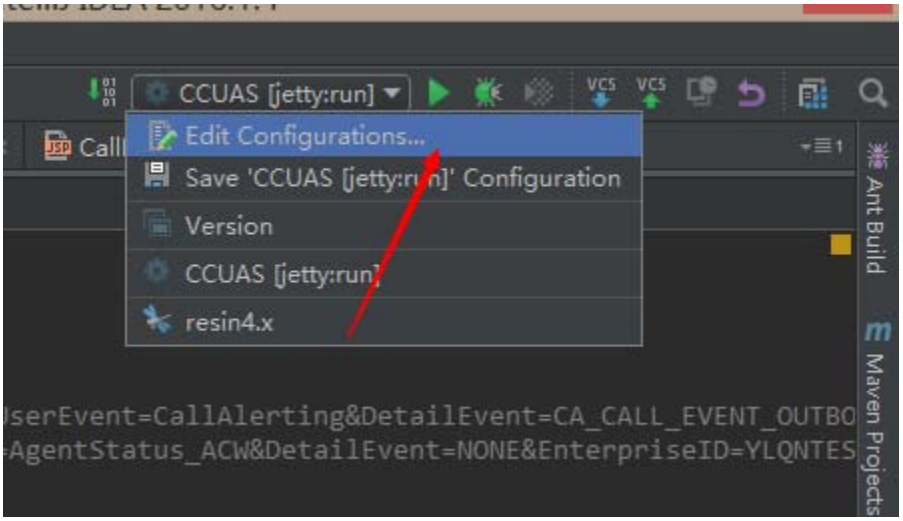
Maven 组件管理如下图所示。



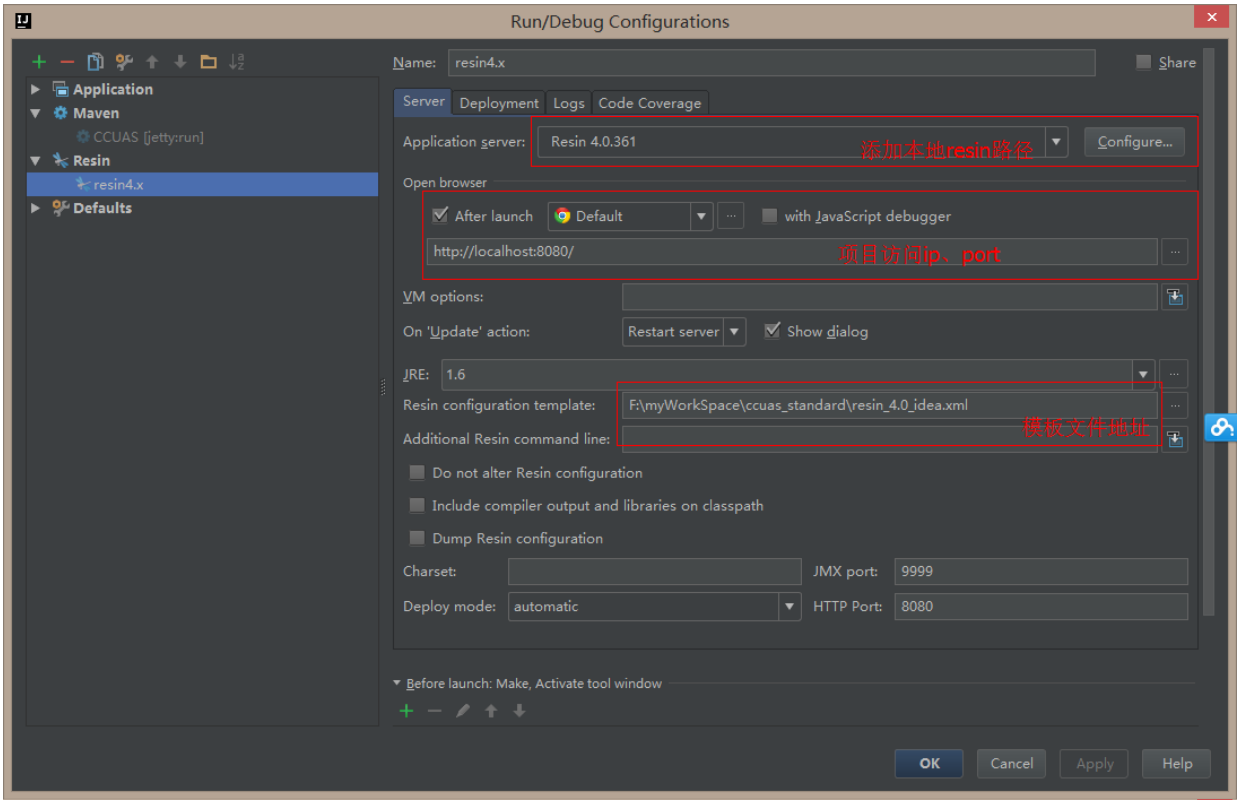
图中 1 处表示 Maven 常用的工具栏，图 2 处展示了 Maven 的生命周期，通过双击对应的命令来执行项目编译、打包、部署等操作，图 3 处为在 pom.xml 中配置的插件列表，方便调用插件，图 4 处为在 pom.xml 中配置的依赖包列表。

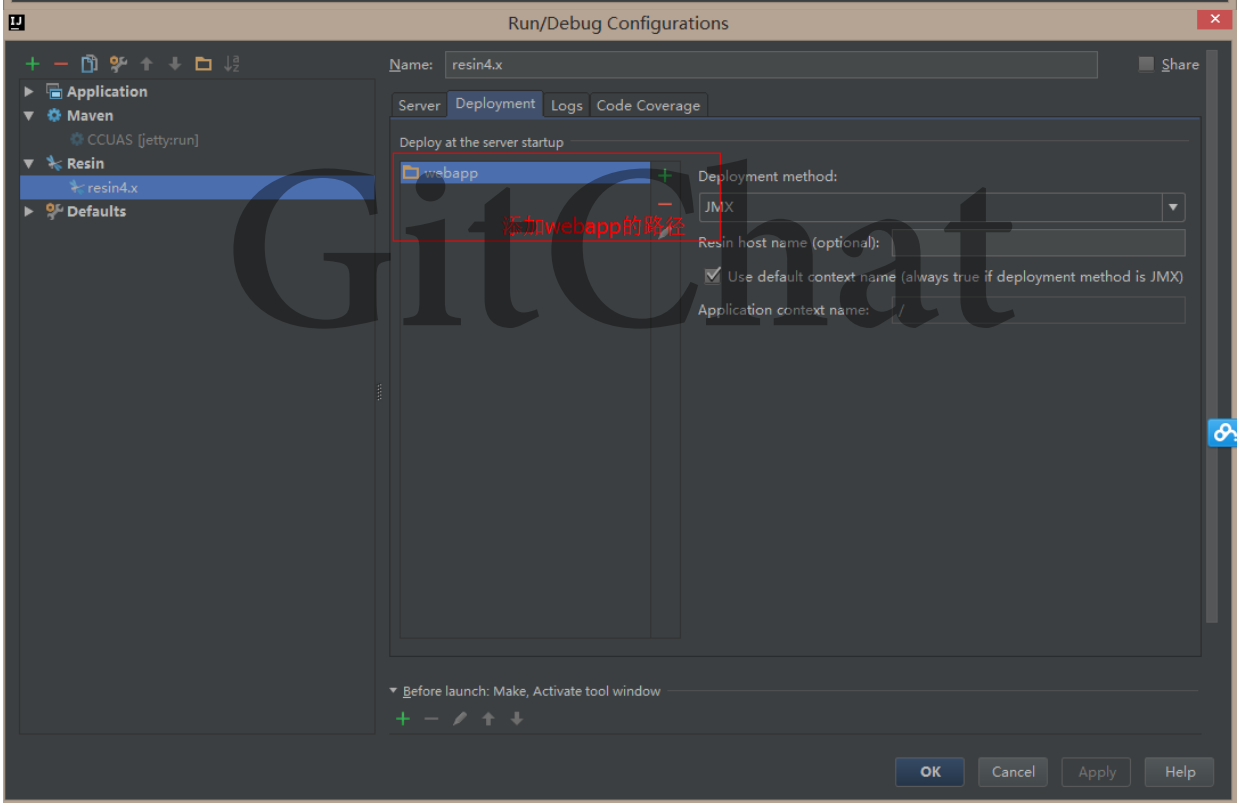
容器配置

接下来，我们说说开发常用容器。本次主要使用 resin。



如下图所示，在配置完项目结构，选择 Edit Configurations 来添加容器。



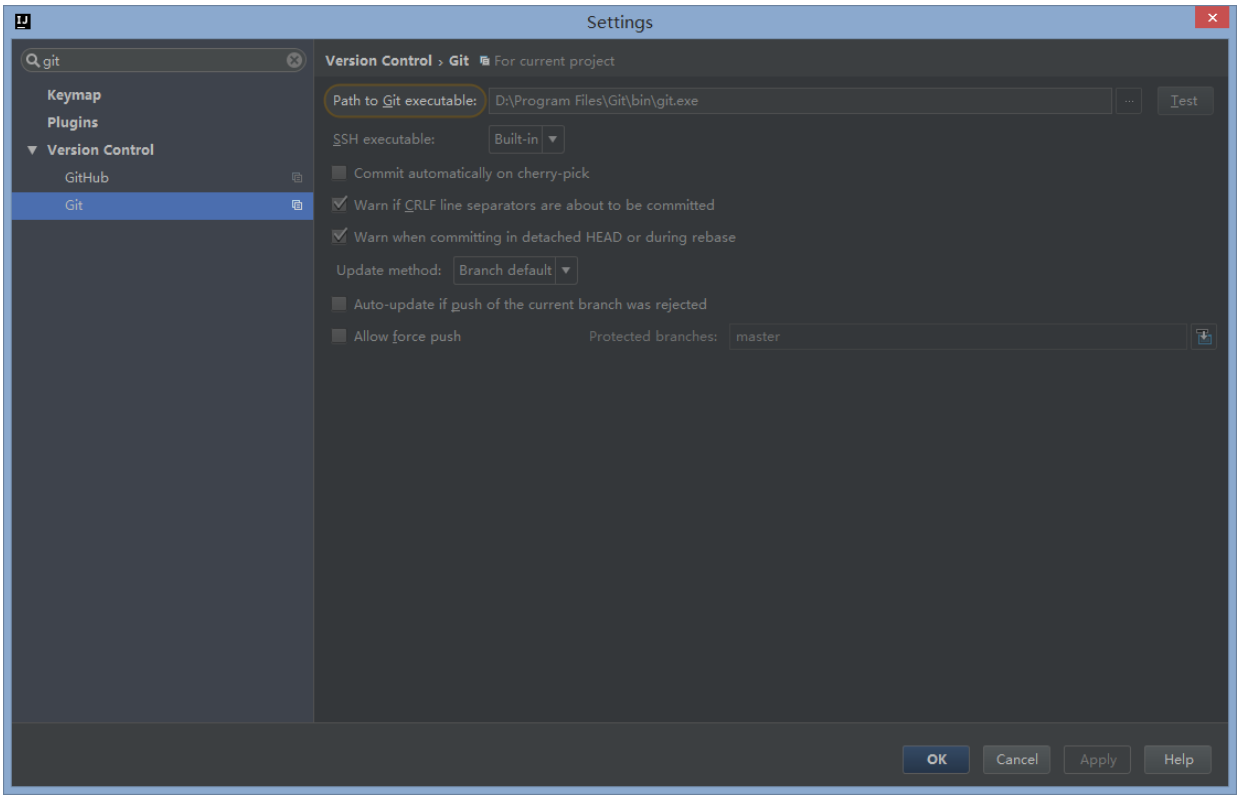


集成 Git

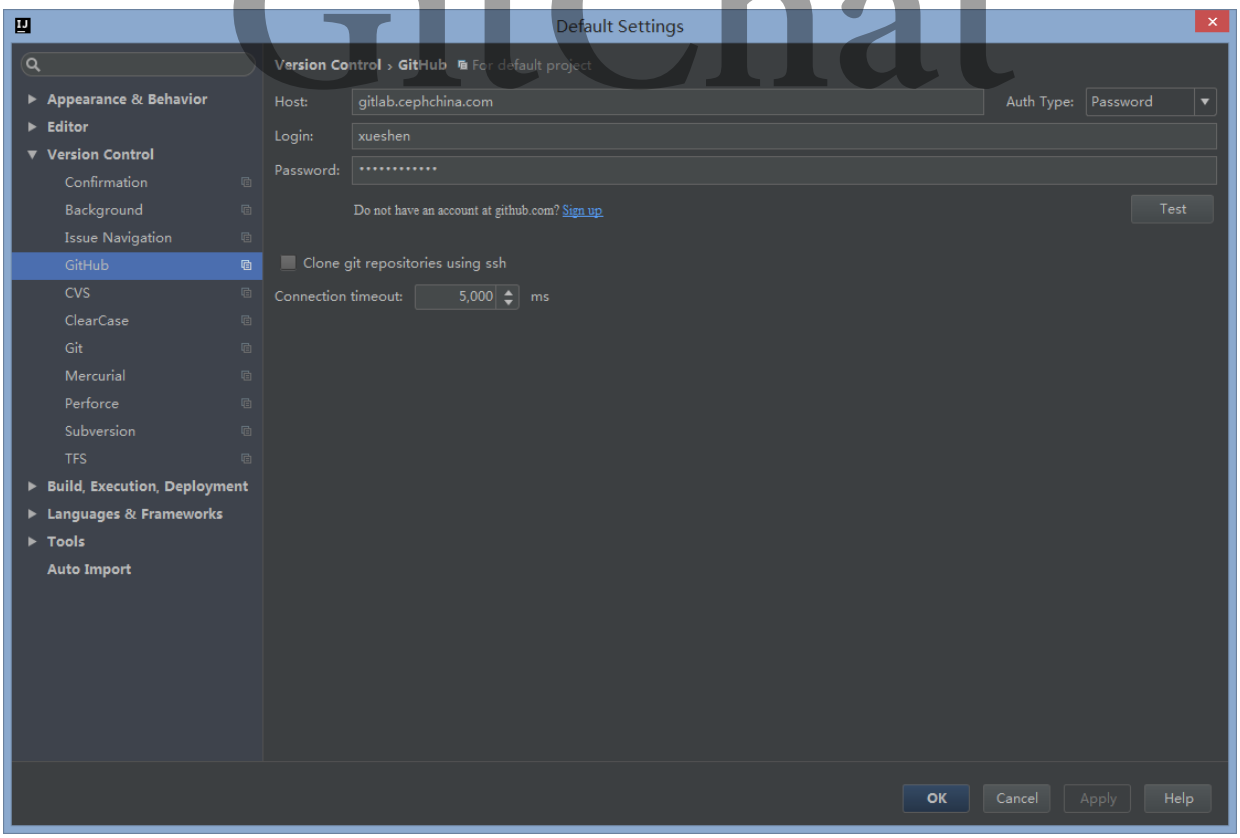
配置 Git 的过程如下。

1. 在本机安装 Git 客户端。
2. 进入 File –> Other settings –> Version Control –> Git。
3. 在“Path to Git executable”关联客户端 git.exe。

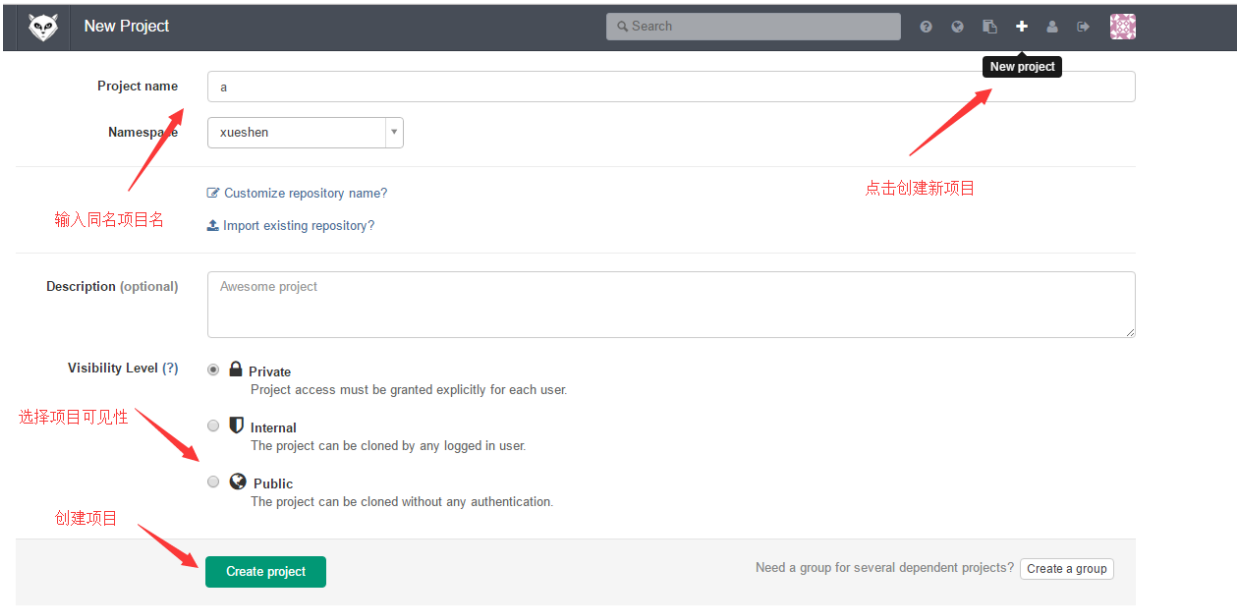
操作界面如下图所示。

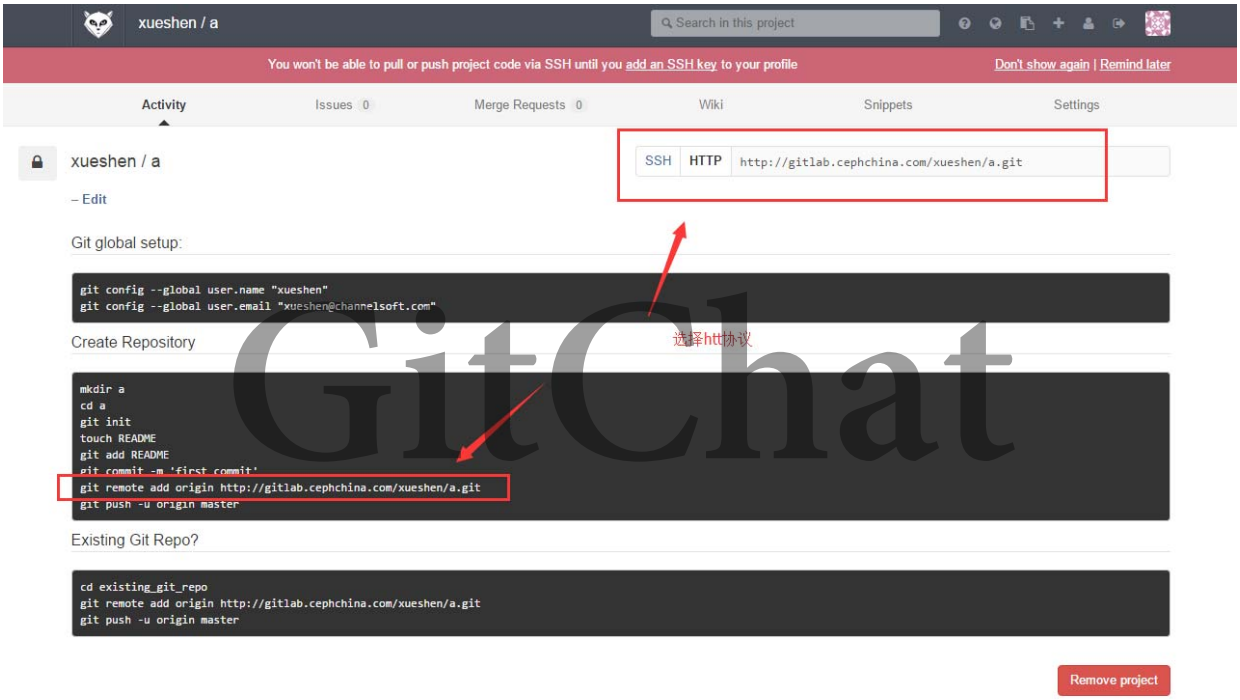


接下来配置 GitLab 账号，操作过程为：进入File –> Other settings –> Version Control –> gitLab，如下图所示。



通过 GitLab 创建项目，操作过程如下图所示。

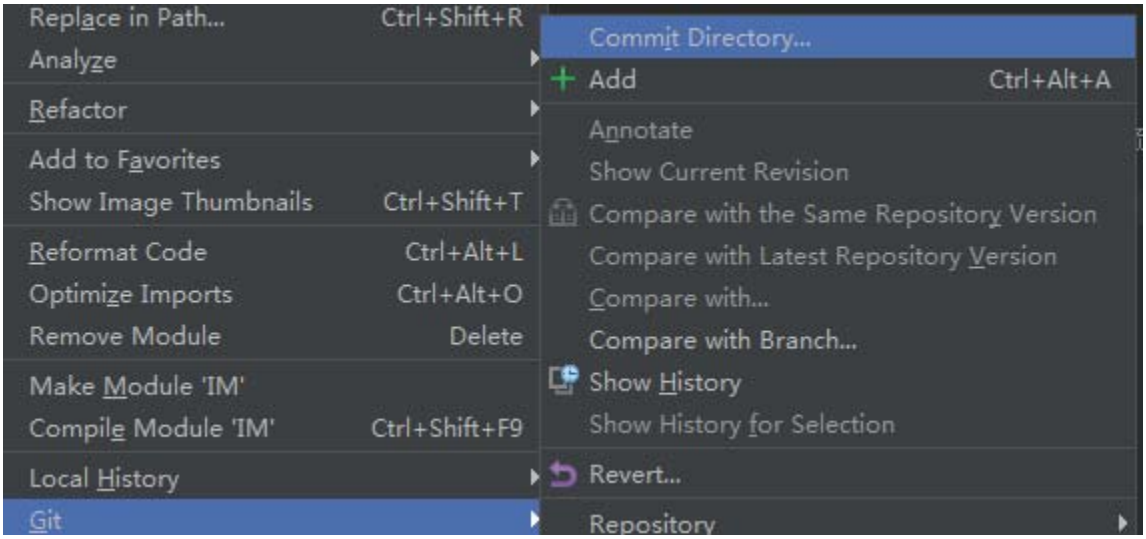




本地关联 GitLab，操作步骤如下。

1. 建立相关项目。
2. 创建 Git 本地：VCS -> Import into Version Control -> Create Git Repository。
3. 右键工程 -> add。
4. 选择 Commit Directory（不 push）。
5. 打开 terminal 控制台按照提示添加服务端地址（首先要安装git 并且配置环境变量）
6. push 整个工程

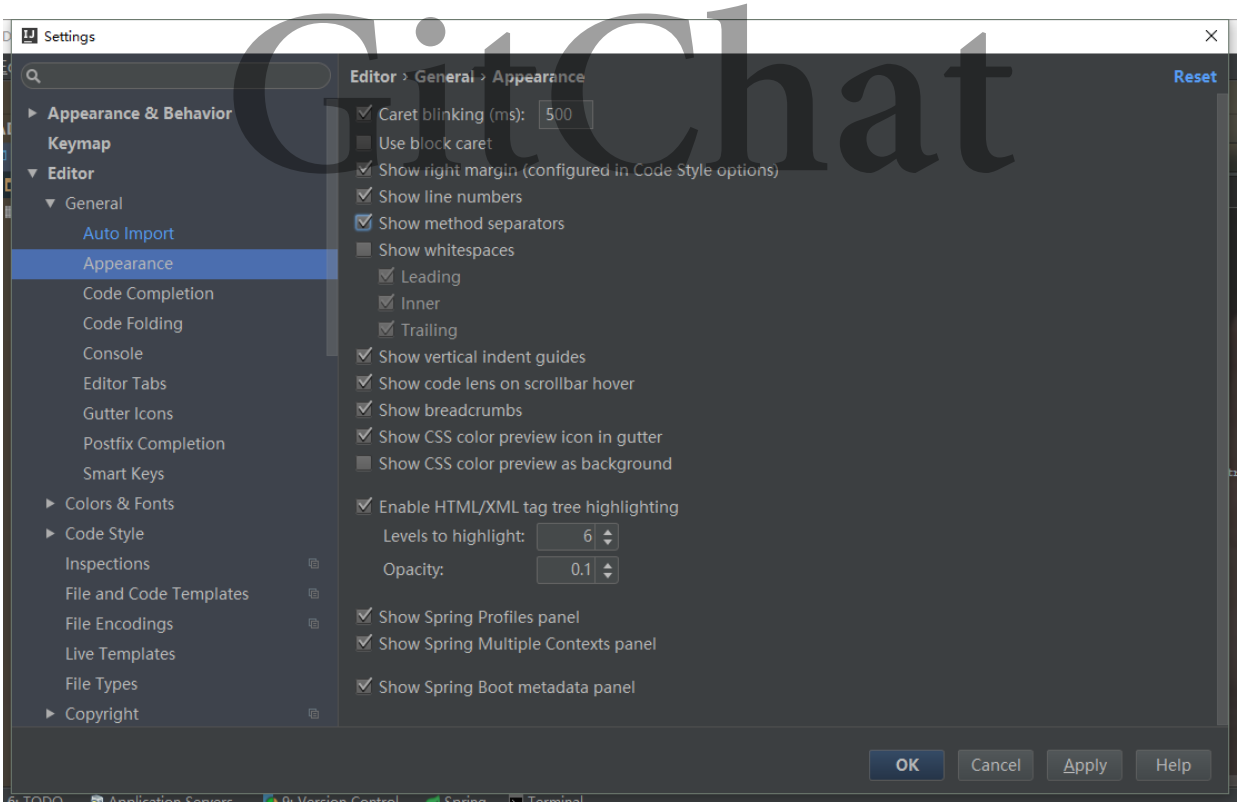
操作过程如下图所示。



如果提示“Push failed: Failed with error: fatal: Could not read from remote repository.”，先输入 $ git remote rm origin，再重新按照提示输入关联命令就不会报错了！

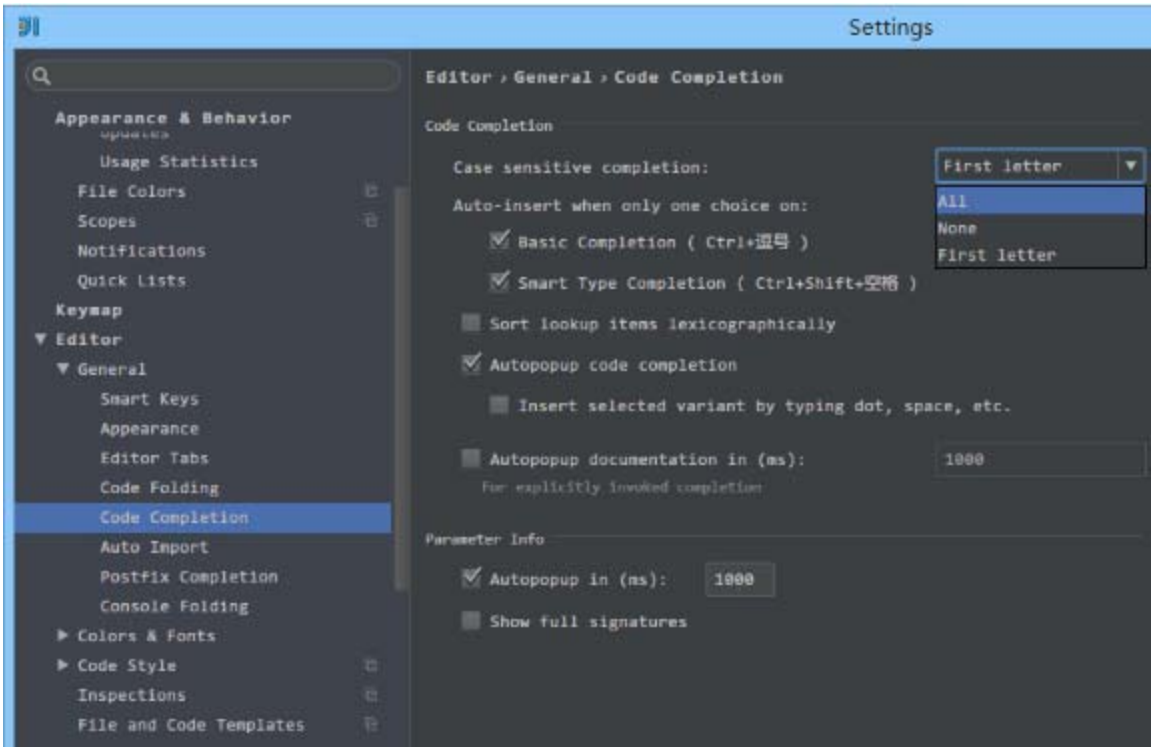
IDEA 常用设置

IDEA 常用设置如下图所示。



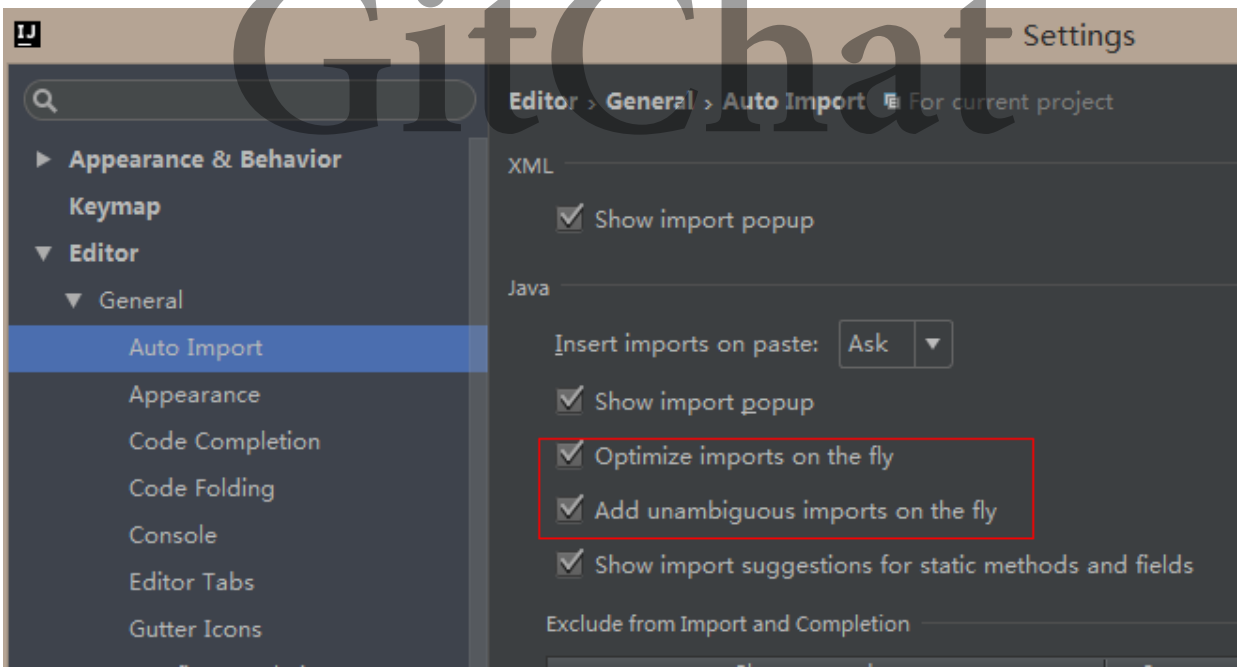
上图中，Show line numbers，表示显示行号，建议勾选。Show method separators，表示显示方法线，建议勾选。

IntelliJ IDEA 的代码提示和补充功能有一个特性：区分大小写。如上图标注所示，默认是 First letter，区分大小写。如果不想区分大小，改为 None 选项即可。

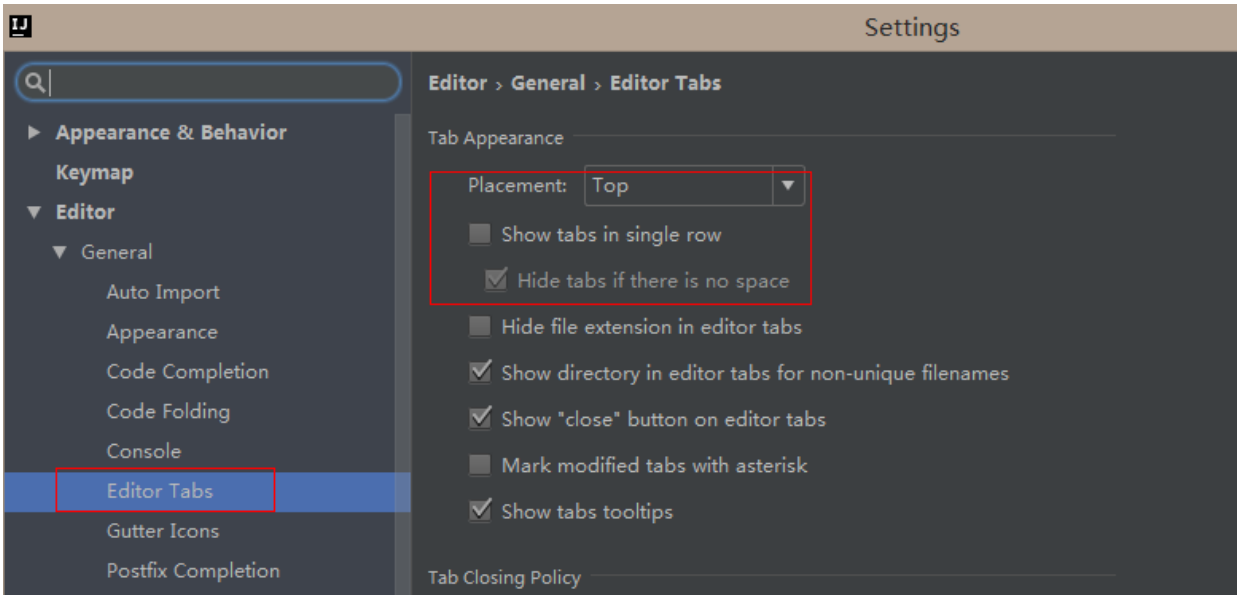


默认 IntelliJ IDEA 是没有开启自动 import 包的功能。勾选下图第一项，IntelliJ IDEA 将在我们书写代码的时候自动帮我们优化导入的包，比如自动去掉一些没有用到的包。

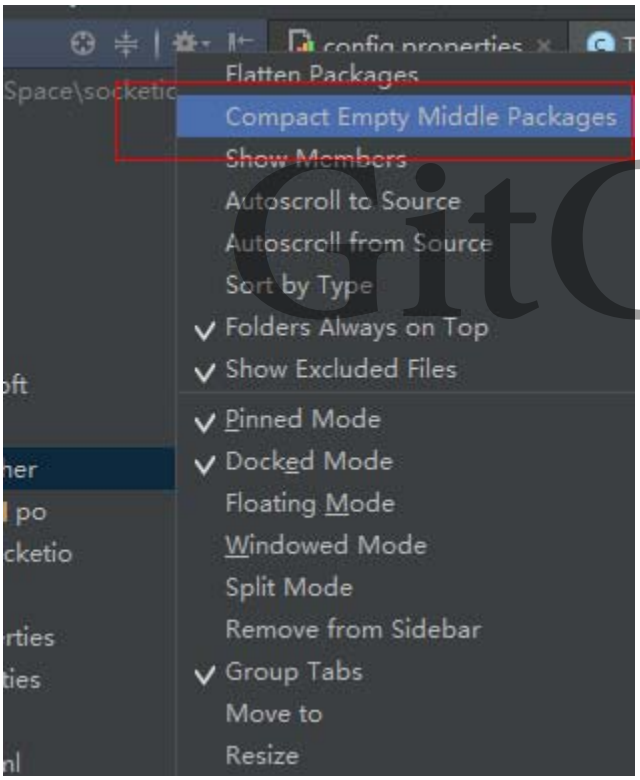
勾选第二项，IntelliJ IDEA 将在我们书写代码的时候自动帮我们导入需要用到的包。



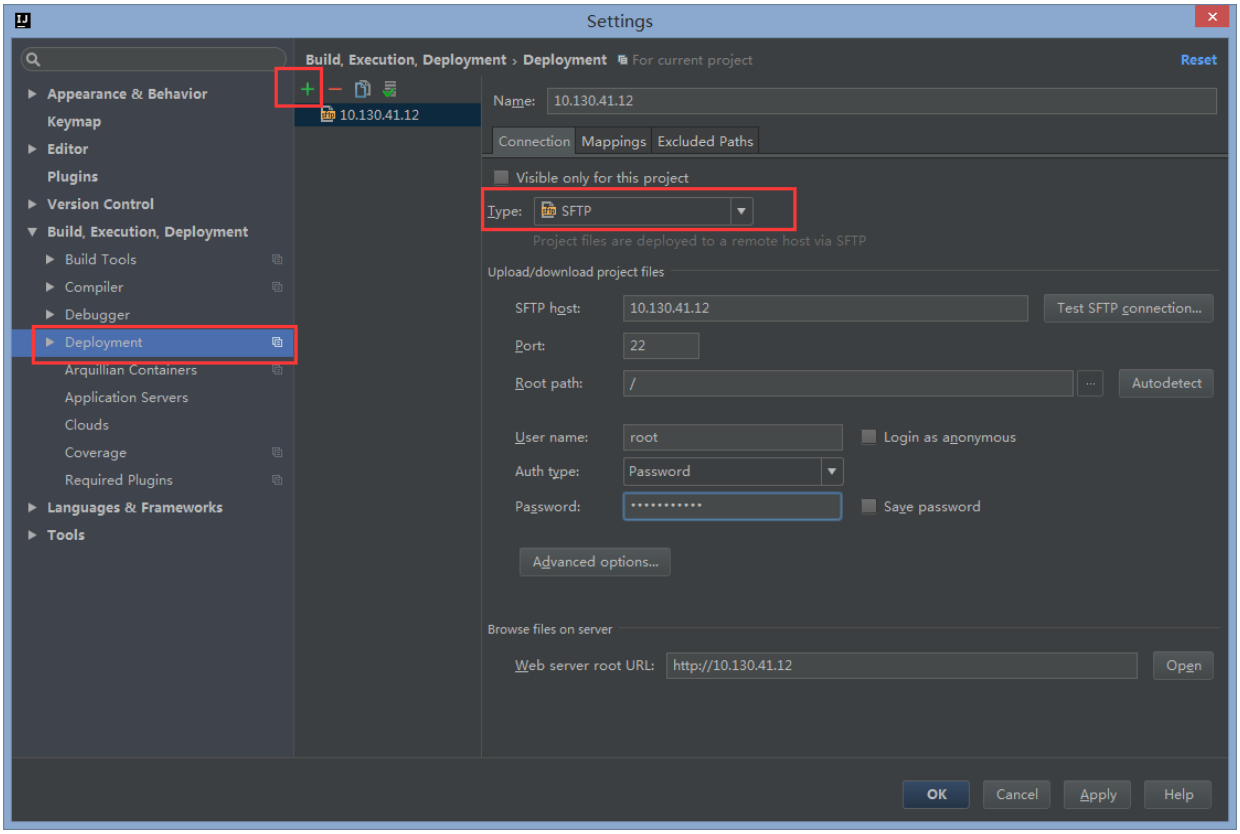
IntelliJ IDEA 默认是把所有打开的文件名 Tab 单行显示的，如下图所示。我们可以修改设置使其多行显示。



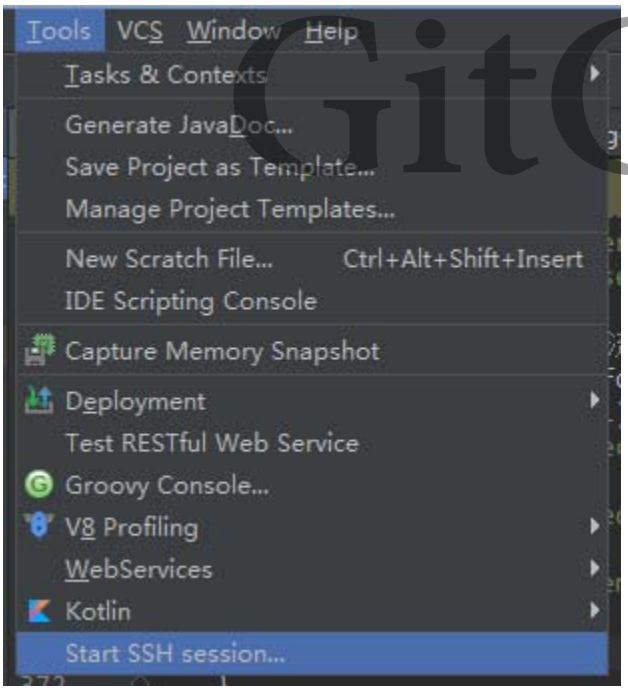
IntelliJ IDEA 默认是会折叠空包的，这样就会出现包名连在一起的情况。但是有些人不喜欢这种结构，喜欢整个结构都是完整树状的，所以我们可以去掉演示中的勾选框即可，如下图所示。

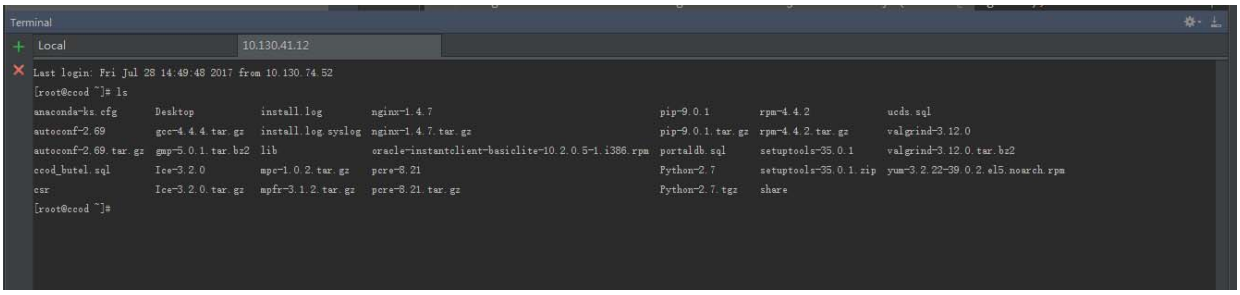


浏览远端服务器，操作为 File –> Settings –> Deployment，添加会话终端如下图所示。

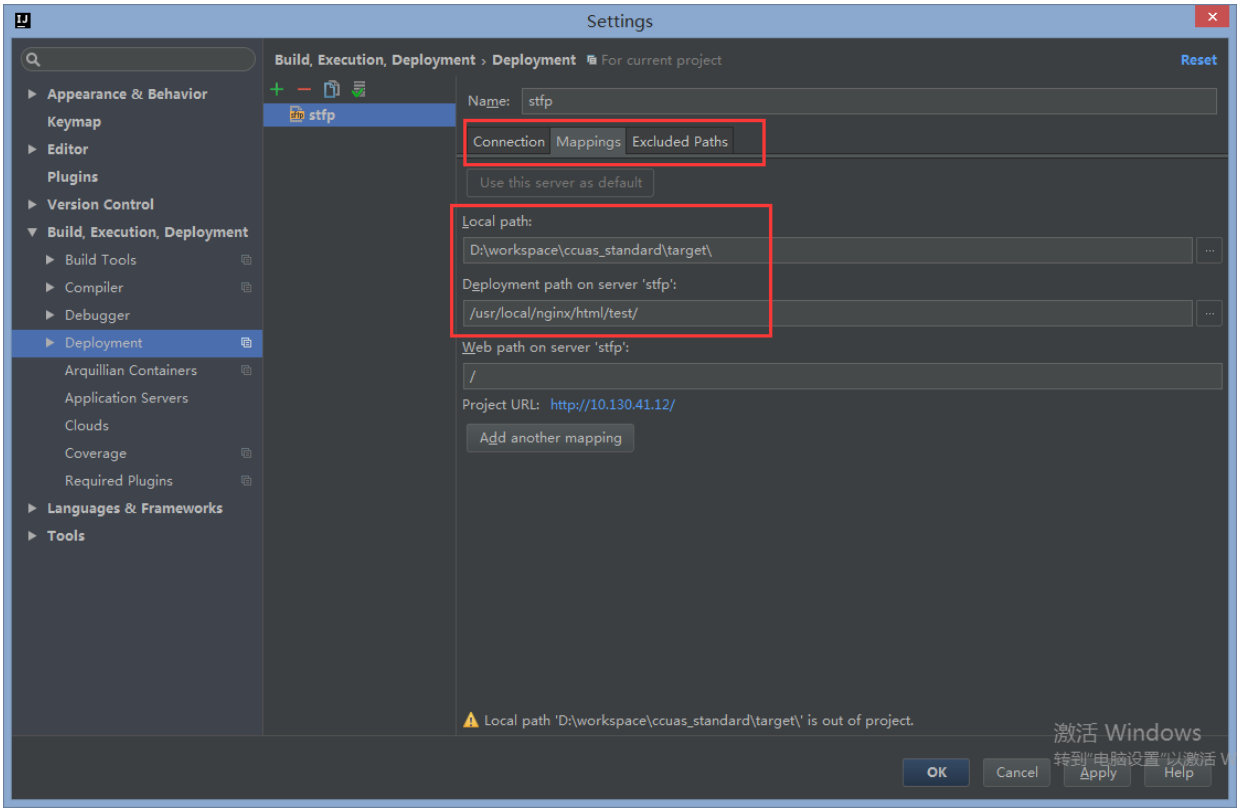


接着，点击 Tools，选择 Start SSH session，启动终端，如下图所示。

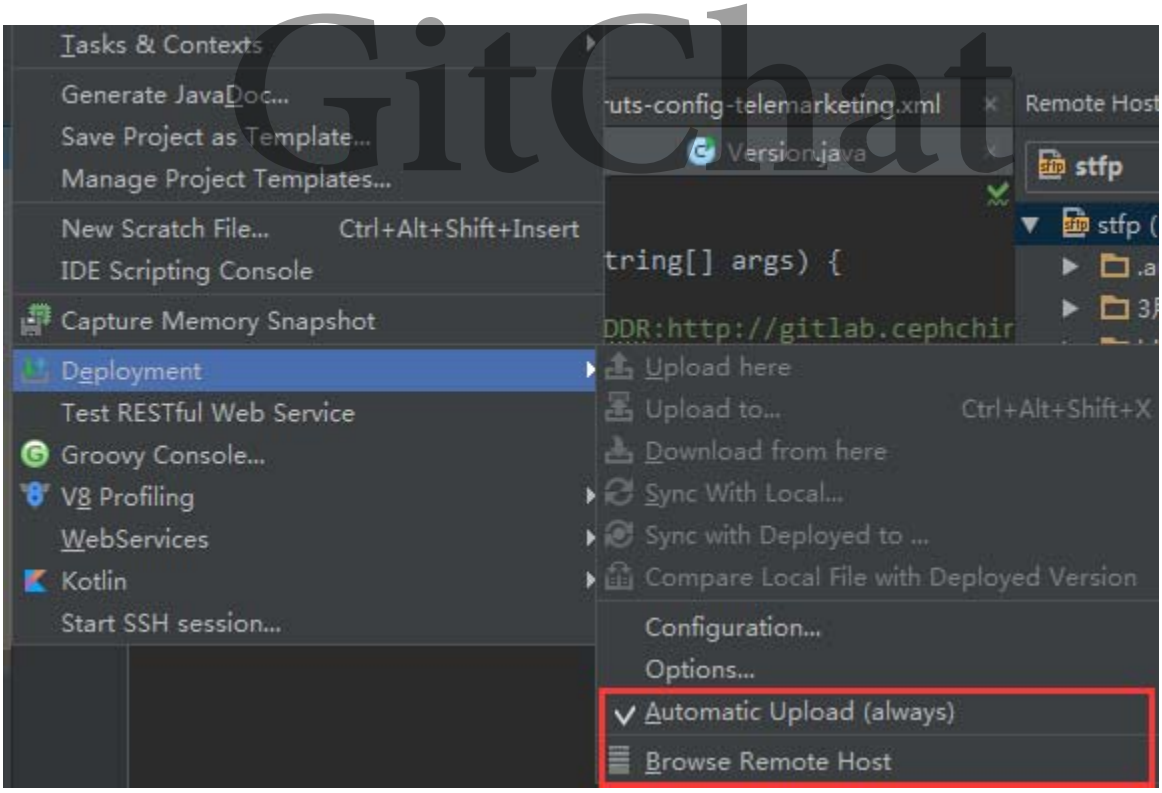




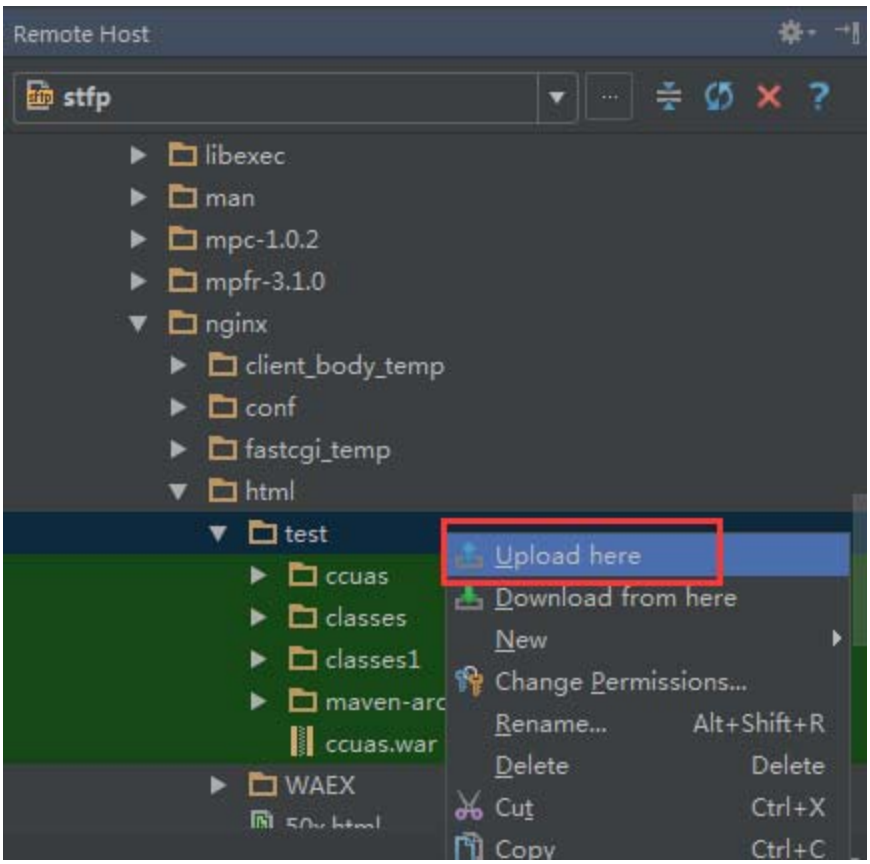
自动上传，操作过程为 File –> Settings –> Deployment –> Mappings，设置本地、远端同步文件夹。如下图所示。



点击 Tools -> Browse Remote Host，开启远端服务器浏览框，并勾选 Automaic UpLoad 自动上传功能。如下图所示。



在 Remote Host 中，选中目标地址右键 UpLoad here，即可实时同步代码，如下图所示。


Remote Host
stfp
libexec
man
mpc-1.0.2
mpfr-3.1.0
nginx
client_body_temp
conf
fastcgi_temp
html
test
ccuas
classes
classes1
maven-arc
ccuas.war
WAEX
Upload here
Download from here
New
Change Permissions...
Rename... Alt+Shift+R
Delete Delete
Cut Ctrl+X
Copy Ctrl+C

GitChat