

# [IntelliJ Idea 教程]

[涵盖安装、配置、常见问题&技巧、Maven、Git、Tomcat、快捷键、项目配置等]

[作者: Ricky]  
[交流群: 244930845]  
2017 年 5 月 20 日

# 目录

入门.....	3
安装.....	3
目录说明.....	7
启动配置.....	7
配置空间.....	8
首次启动.....	9
首次配置.....	13
Appearance&Behavier.....	14
KeyMap.....	18
Editor.....	18
Plugins.....	46
Version Control.....	46
Build Execution Deployment.....	47
Languages&Frameworks.....	52
Tools.....	53
第一次启动后.....	53
调出面板和按钮组.....	54
面板说明.....	54
项目配置.....	55
Project.....	55
Modules.....	55
Libraries.....	56
Facets.....	57
ArtFacts.....	58
SDK.....	58
Global libraries.....	59
Problems.....	59
Maven 专题.....	60
配置.....	60
主配置.....	61
Import 配置.....	61
Ignore Files 配置.....	62
Runner 配置.....	62
RunnerTest.....	63
Repositories 配置.....	64
使用.....	64
面板说明.....	65
命令模式.....	67
Tomcat 专题.....	69
安装配置.....	70
启动.....	71
面板说明.....	71

Run with coverage.....	73
GIT 专题.....	74
安装.....	75
使用.....	75
荡项目.....	76
更新项目.....	76
面板说明.....	76
Local Changes.....	77
Shelf 面板.....	77
Log 面板.....	78
Console 面板.....	78
项目 git 面板.....	79
仓库选项.....	79
SVN 专题.....	80
配置.....	80
1.1 下载&安装 svn.....	80
1.2 配置.....	80
检出项目.....	81
面板说明.....	87
头部面板.....	87
VersionControl.....	87
常用操作.....	90
加入版本控制.....	90
提交远程.....	91
更新项目.....	93
冲突解决.....	94
常用技巧&问题.....	95
JSP 实时编译问题.....	96
搜索功能失效.....	96
自动导入依赖.....	96
提示不区分大小写.....	96
剪贴板数量设置.....	97
生成 JAVADOC.....	98
设置 Spring 支持.....	99
启动时不自动打开项目.....	99
全屏设置.....	100
本地历史.....	100
搜索.....	101
所有文件.....	101
项目文件.....	102
取消重复代码提示.....	102
设置字符集.....	102
项目字符集.....	103
单个文件.....	103

快捷键.....	103
Ctrl.....	104
Alt.....	105
Shift.....	106
Ctrl + Alt.....	106
Ctrl + Shift.....	107
Alt + Shift.....	108
Ctrl + Shift + Alt.....	108
其他.....	109



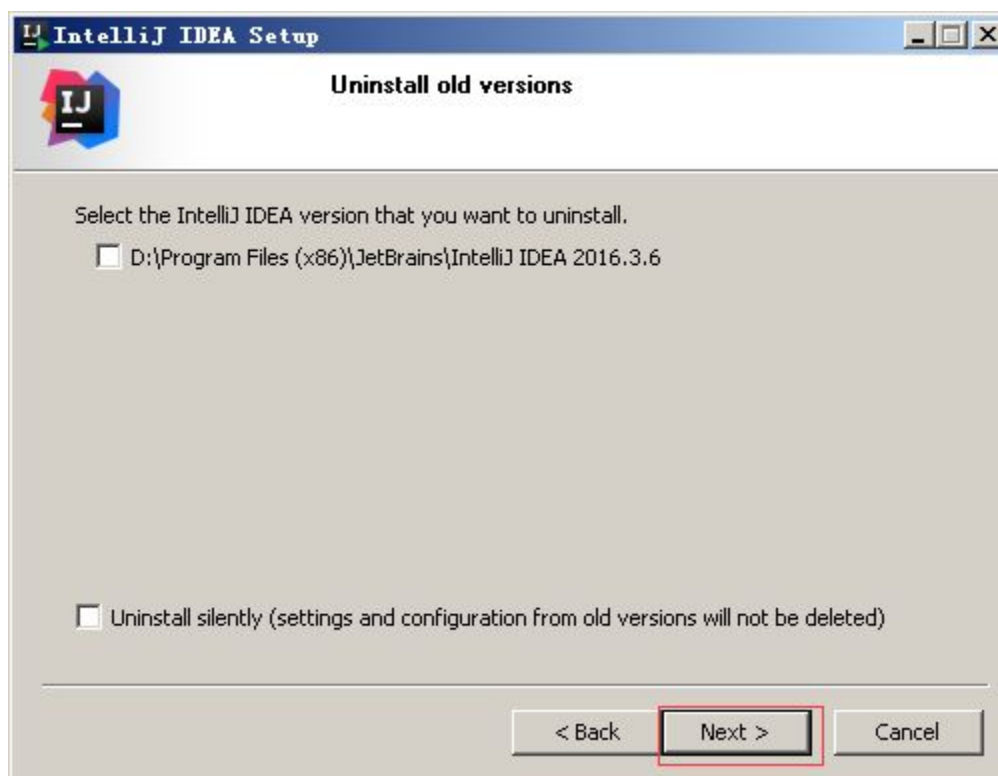
# 入门

## 安装

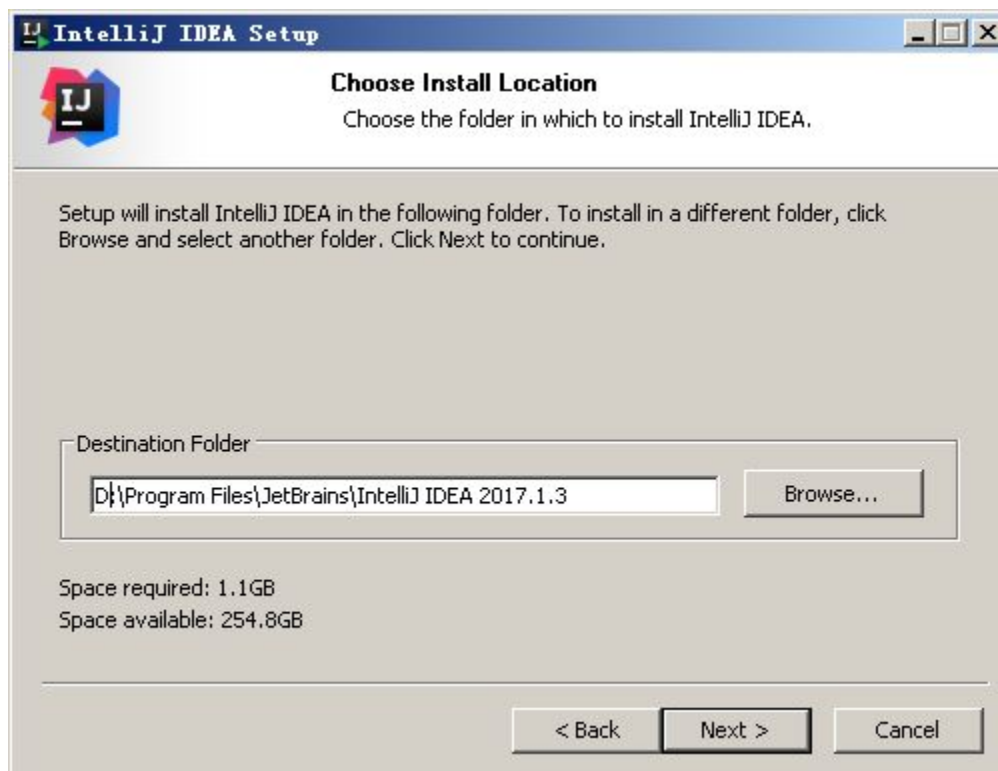
### 运行安装包



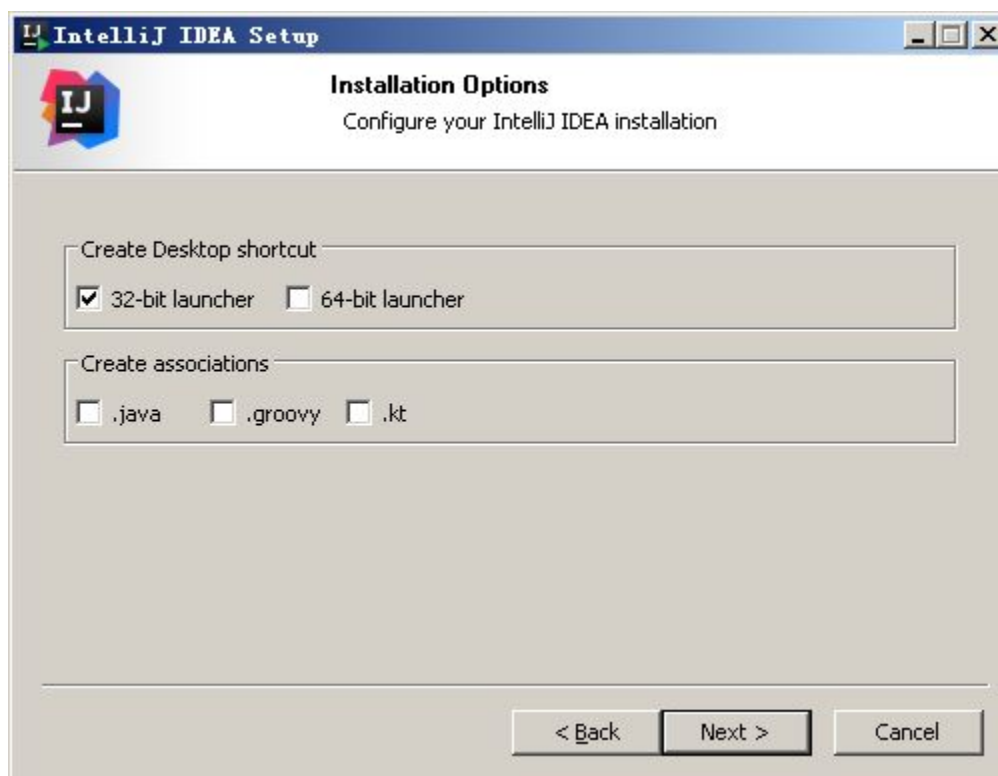
下一步，不卸载旧版本



选择安装目录



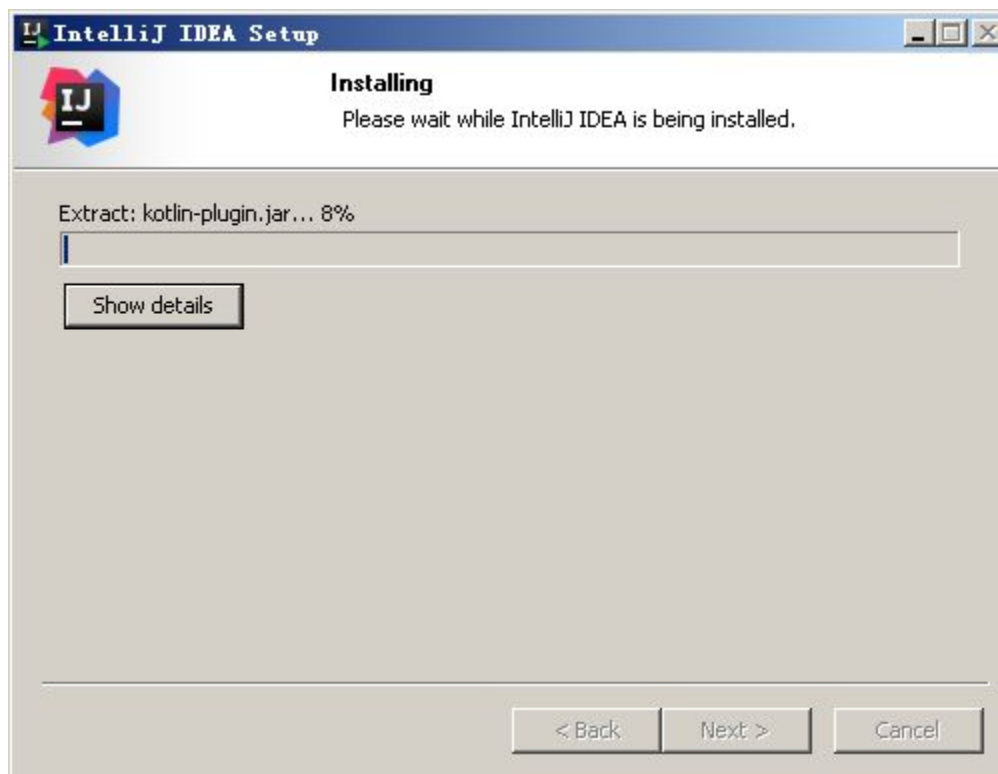
选择桌面快捷和关联文件打开方式



开始菜单



开始安装



完成



## 目录说明

名称	日期	类型	大小
bin	2017/5/18 13:37	文件夹	
help	2017/5/18 13:40	文件夹	
jre64	2017/5/18 13:37	文件夹	
lib	2017/5/18 13:36	文件夹	
license	2017/5/18 13:36	文件夹	
plugins	2017/5/18 13:36	文件夹	
redist	2017/5/18 13:36	文件夹	
build.txt	2017/5/13 0:49	TXT 文件	1 KB
Install-Windows-zip.txt	2017/5/13 0:49	TXT 文件	3 KB
ipr.reg	2017/5/13 0:49	注册表项	1 KB

**Bin:** 容器，执行文件和启动参数等。

**Help:** 快捷键文档和其他帮助文档

**Jre64:** 64 位 java 运行环境

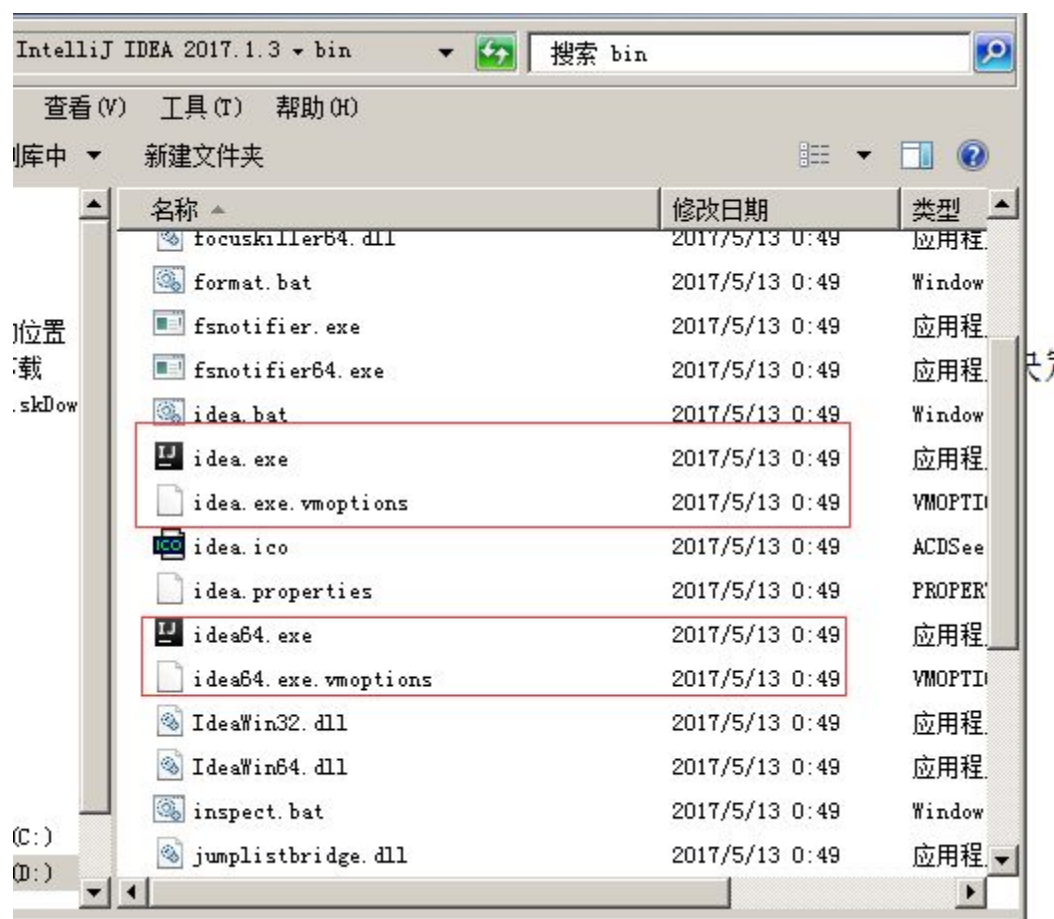
**Lib:** idea 依赖的类库

**License:** 各插件许可

**Plugin:** 插件

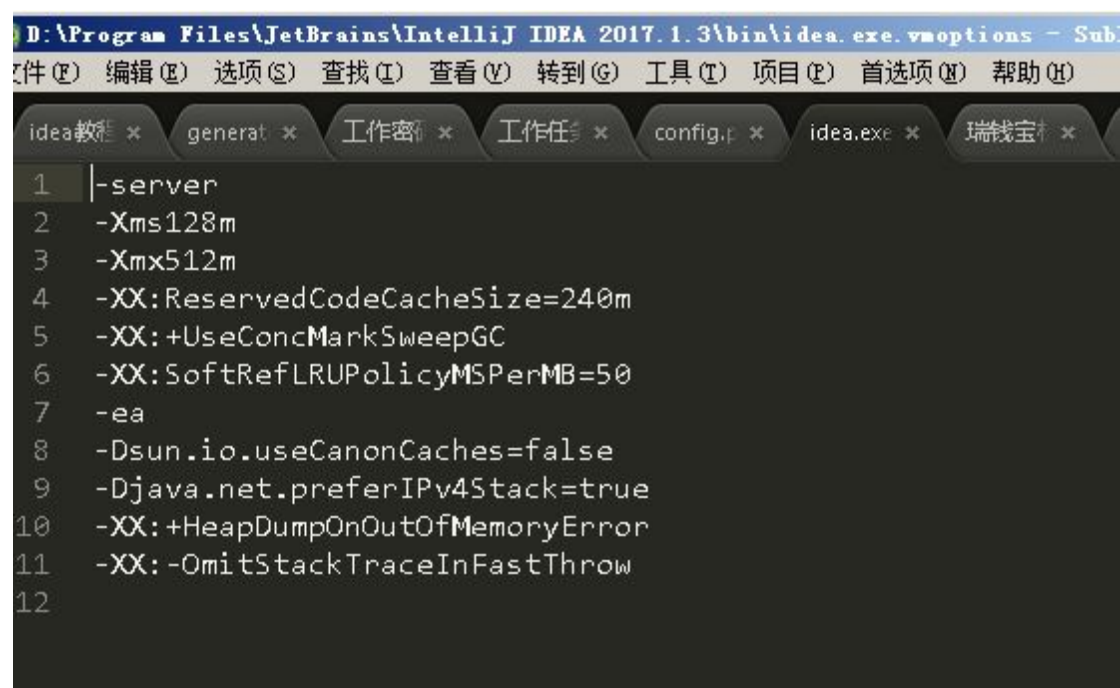
## 启动配置

Idea 启动有 32 位和 64 位之分，具体根据运行环境决定



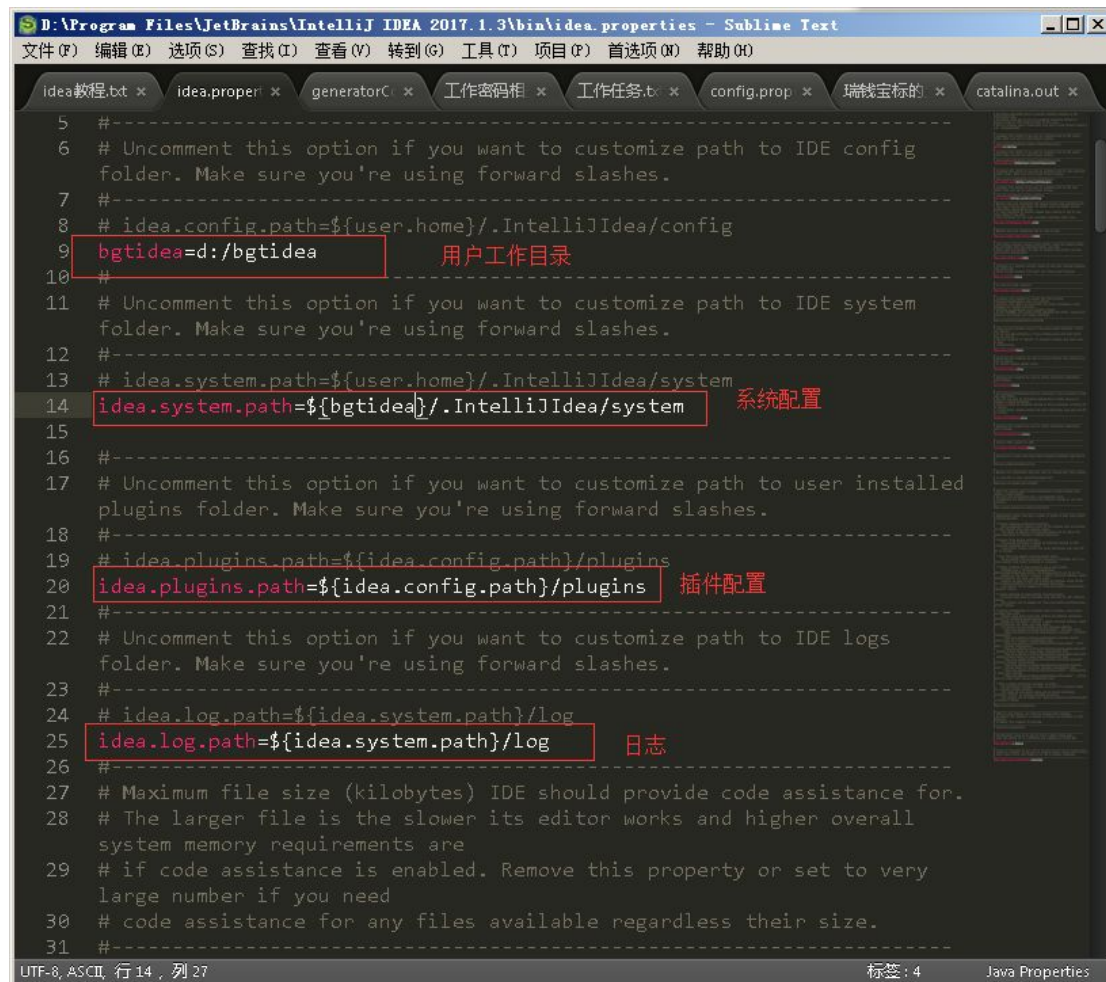
这里以 idea64 为例进行说明

可以根据机器情况配置 vm 参数



## 配置空间

配置这个的目的是方便进行迁移,即在新的环境中不用在手动配置相关配置(比如主题, maven, jdk 等)



```
D:\Program Files\JetBrains\IntelliJ IDEA 2017.1.3\bin\idea.properties - Sublime Text
文件(F) 编辑(E) 选项(O) 查找(F) 查看(V) 转到(G) 工具(T) 项目(P) 首选项(O) 帮助(H)

idea教程.txt x idea.proper x generatorC x 工作密码相 x 工作任务.b x config.prop x 瑞钱宝标的 x catalina.out x

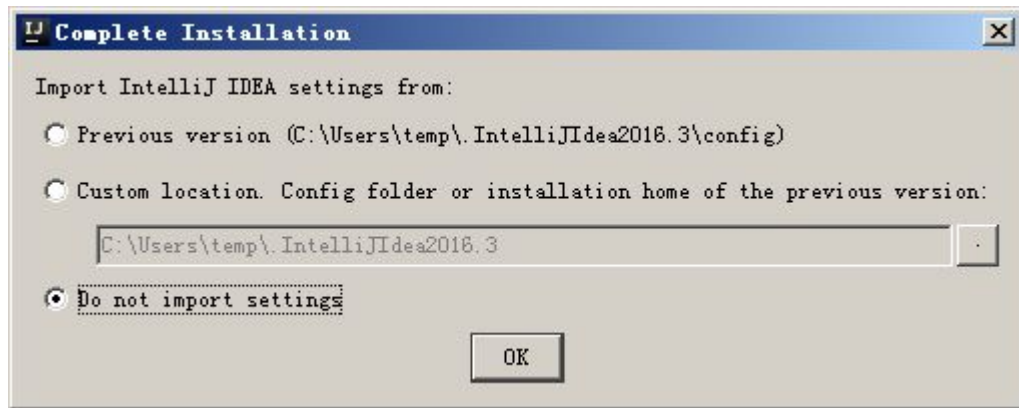
5 #-----
6 # Uncomment this option if you want to customize path to IDE config
  folder. Make sure you're using forward slashes.
7 #-----
8 # idea.config.path=${user.home}/.IntelliJ IDEA/config
9 bgidea=d:/bgidea          用户工作目录
10 #-----
11 # Uncomment this option if you want to customize path to IDE system
  folder. Make sure you're using forward slashes.
12 #-----
13 # idea.system.path=${user.home}/.IntelliJ IDEA/system
14 idea.system.path=${bgidea}/.IntelliJ IDEA/system  系统配置
15 #-----
16 # Uncomment this option if you want to customize path to user installed
  plugins folder. Make sure you're using forward slashes.
17 #-----
18 # idea.plugins.path=${idea.config.path}/plugins
19 # idea.plugins.path=${idea.config.path}/plugins  插件配置
20 #-----
21 # Uncomment this option if you want to customize path to IDE logs
  folder. Make sure you're using forward slashes.
22 #-----
23 # idea.log.path=${idea.system.path}/log
24 # idea.log.path=${idea.system.path}/log  日志
25 #-----
26 #-----
27 # Maximum file size (kilobytes) IDE should provide code assistance for.
28 # The larger file is the slower its editor works and higher overall
  system memory requirements are
29 # if code assistance is enabled. Remove this property or set to very
  large number if you need
30 # code assistance for any files available regardless their size.
31 #-----

UTF-8, ASCII 行 14, 列 27 标签: 4 Java Properties
```

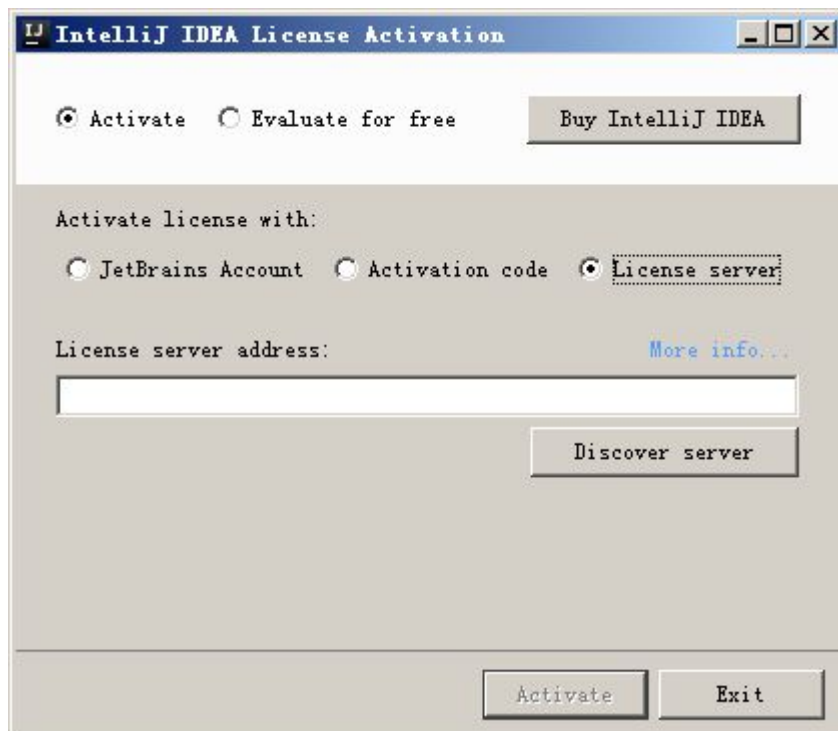
## 首次启动

配置 settings





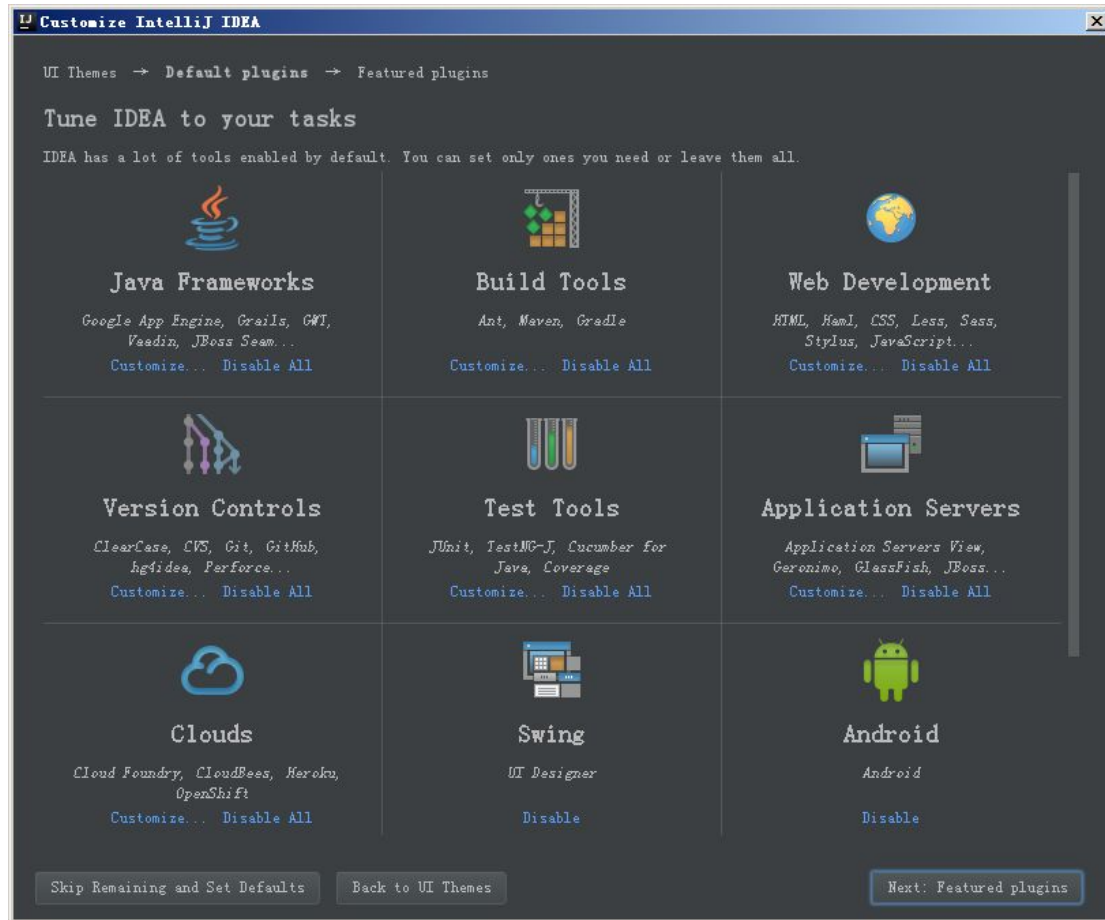
注册软件

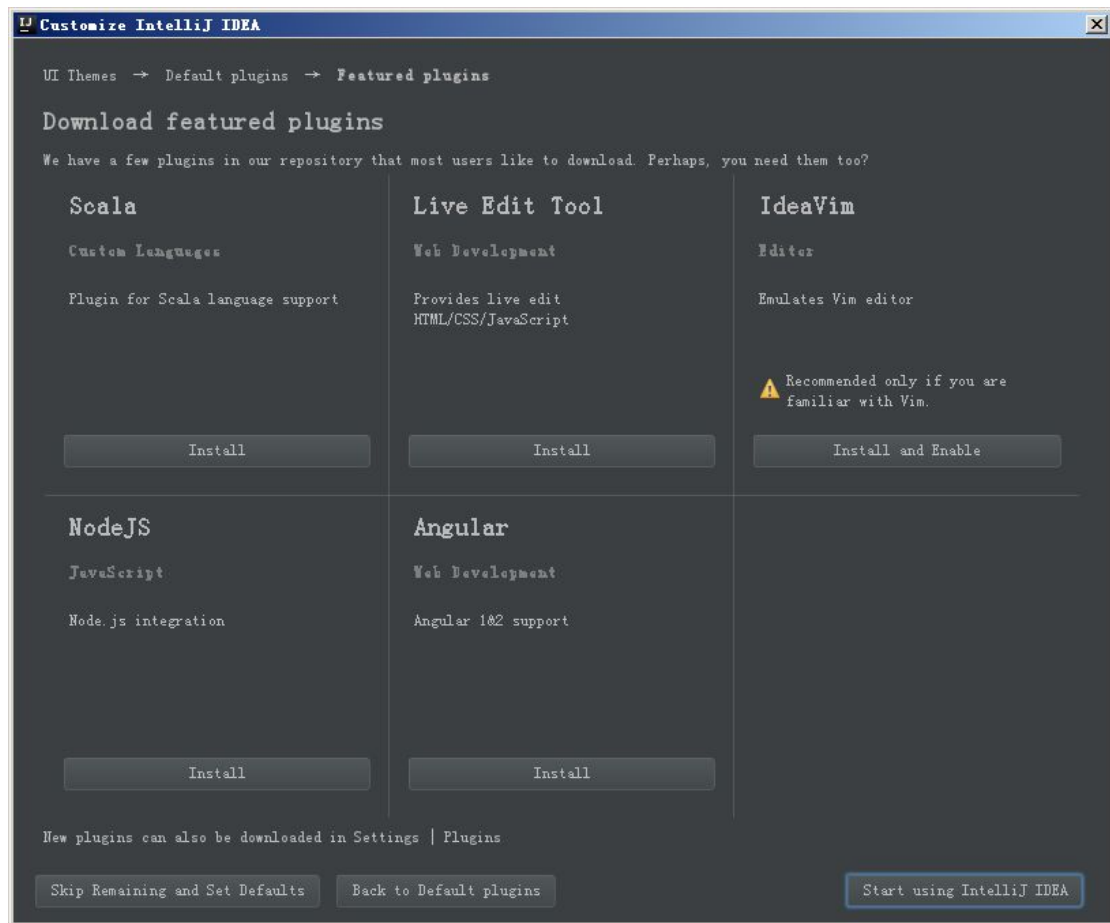


选择主题







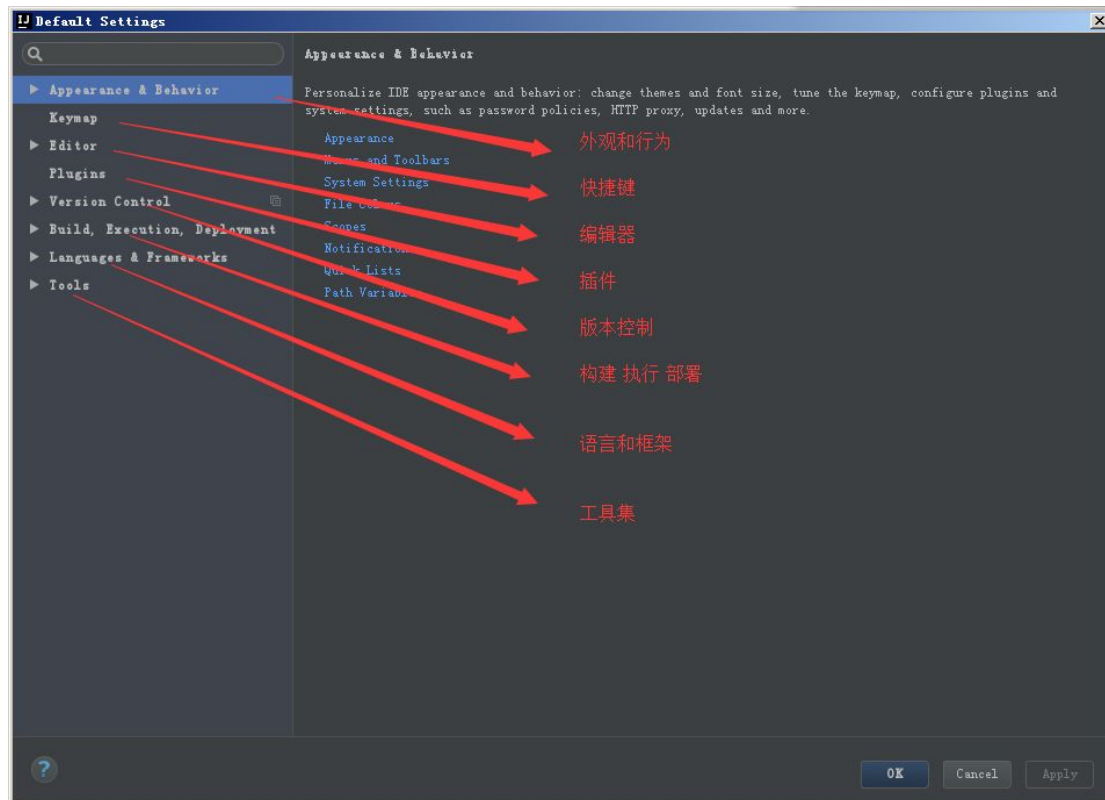


## 启动



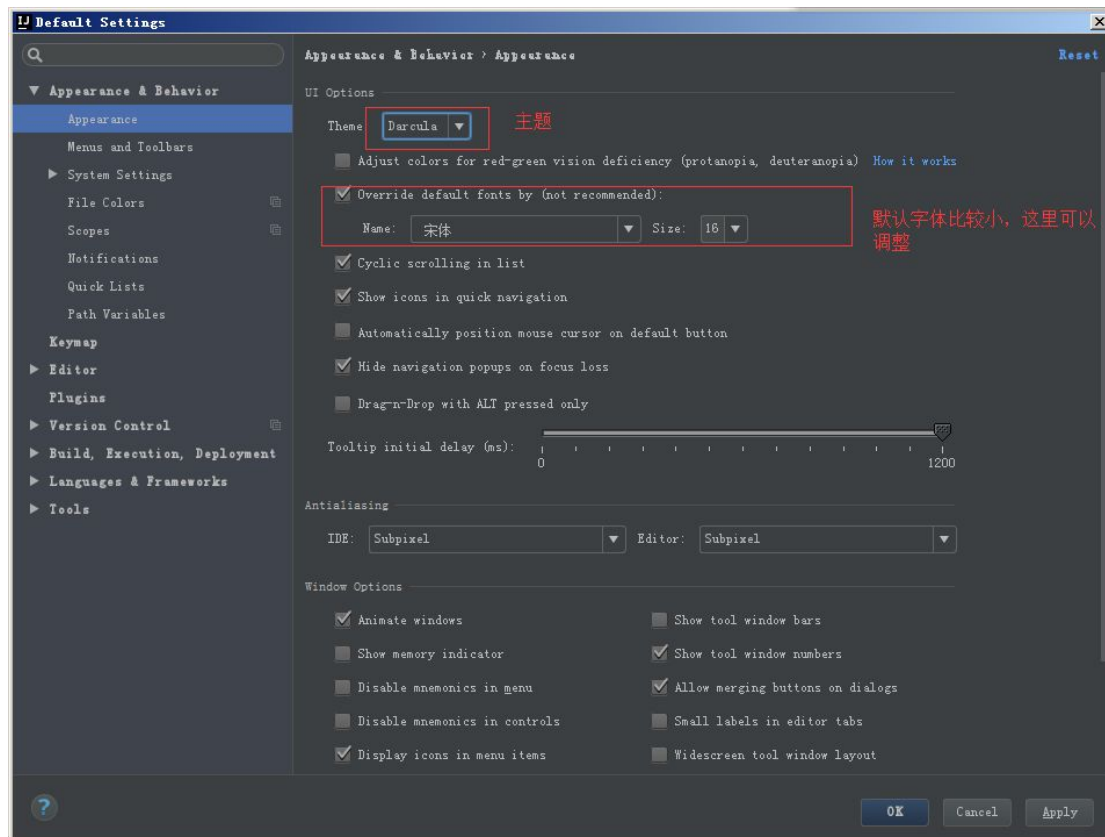
## 首次配置

Configuration—>settings

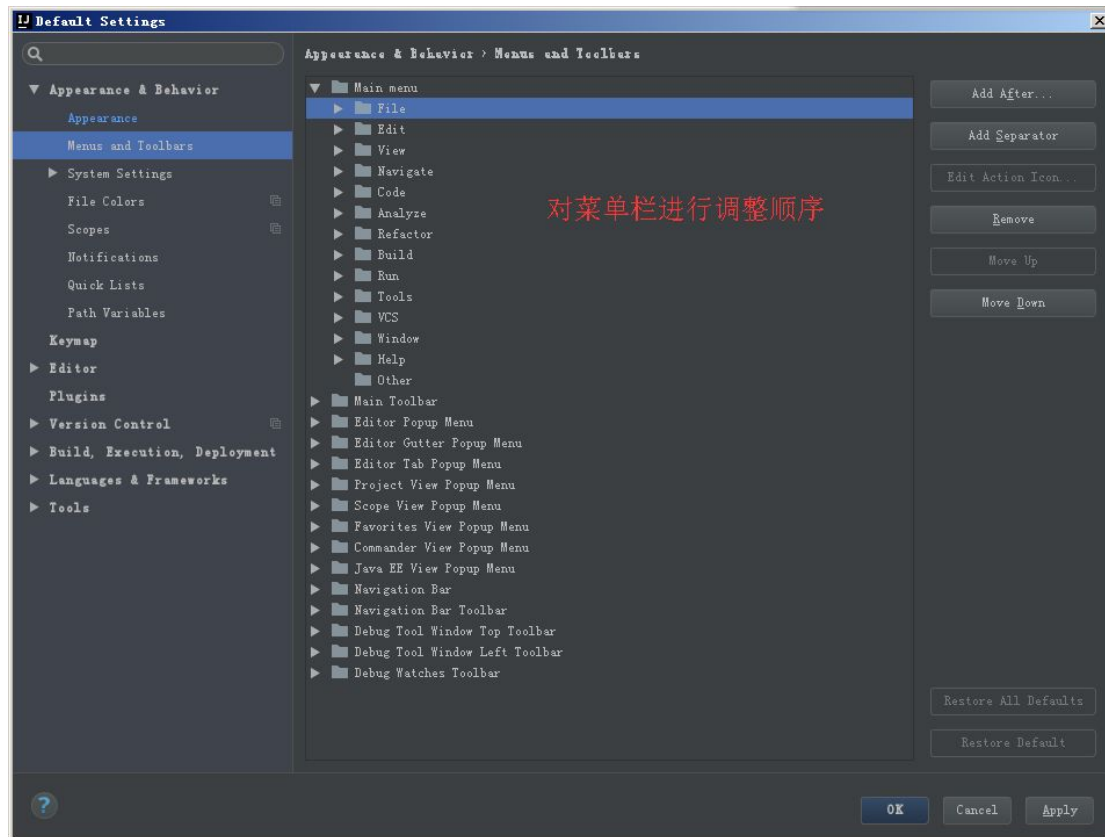


## Appearance&Behavier

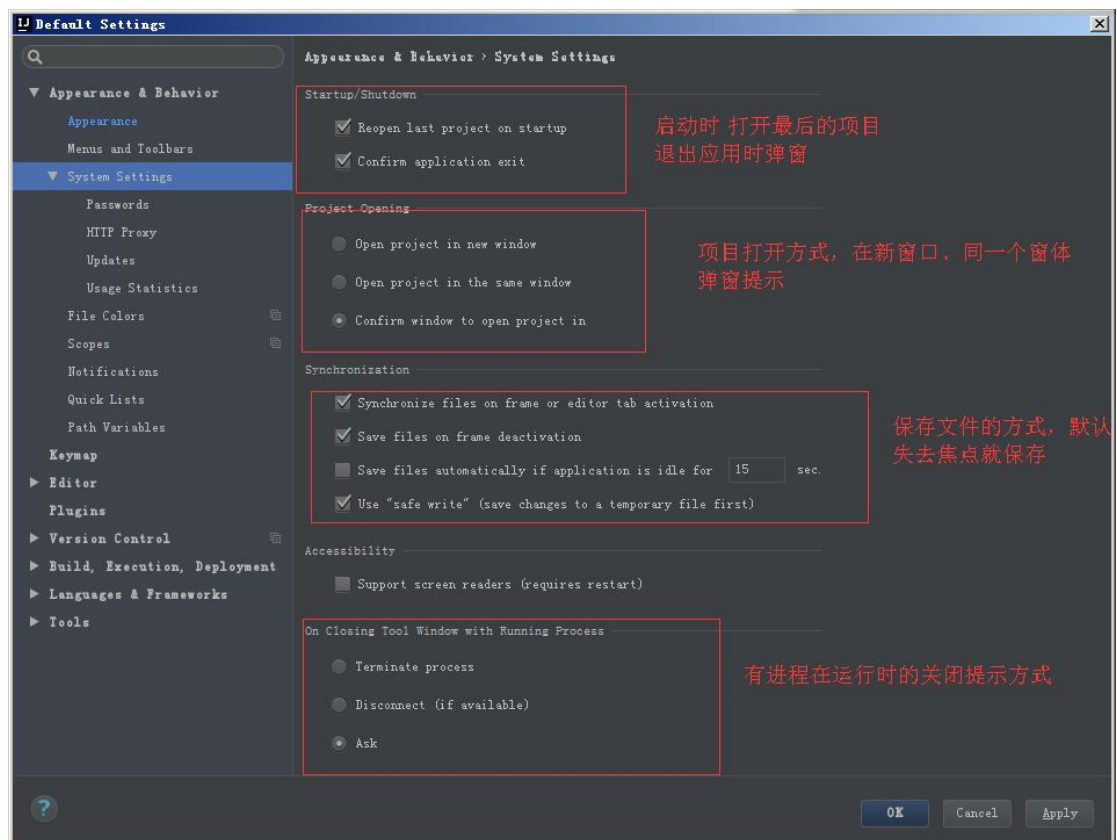
### Appearance



## Menus and Toolbars



## System Setting



## File Colors

文件颜色, 保存默认即可

## Scope

范围, 保存默认即可

## Notificactions

通知, 保存默认即可。

## Quick Lists

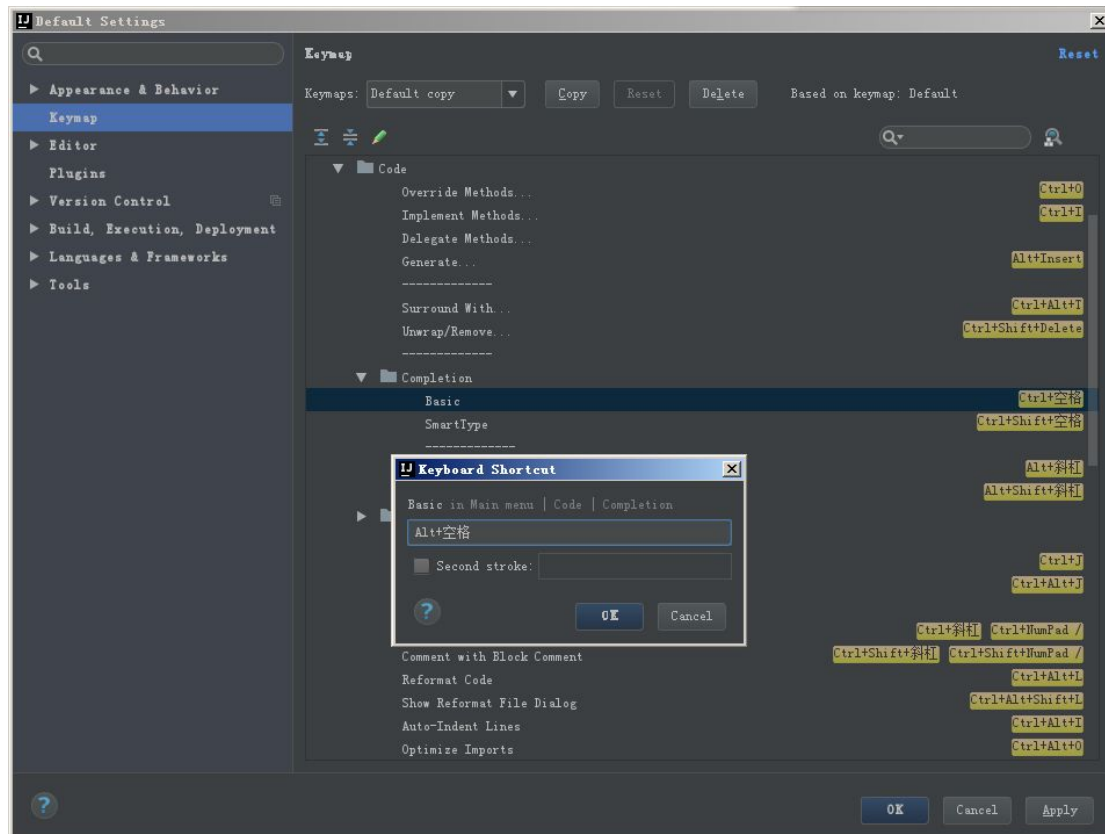
快捷列表, 保存默认即可。

## Path variable

可用路径配置，保存默认即可。

## KeyMap

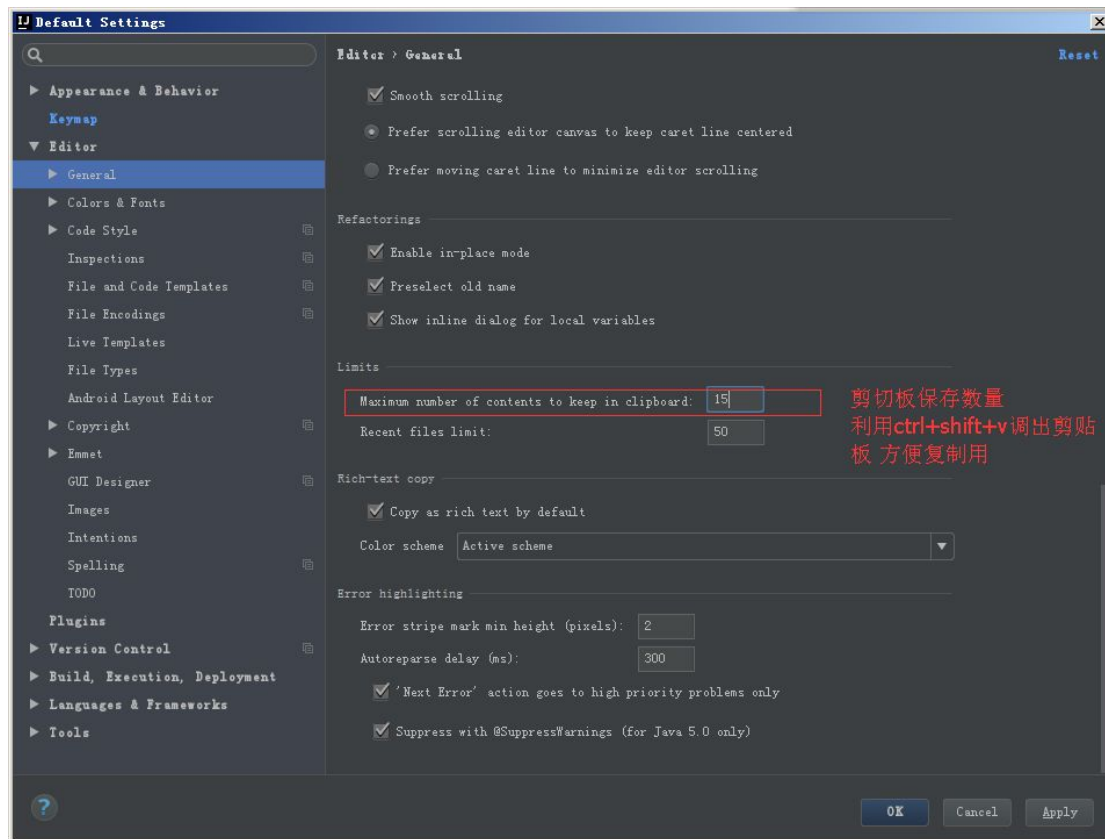
配置快捷键



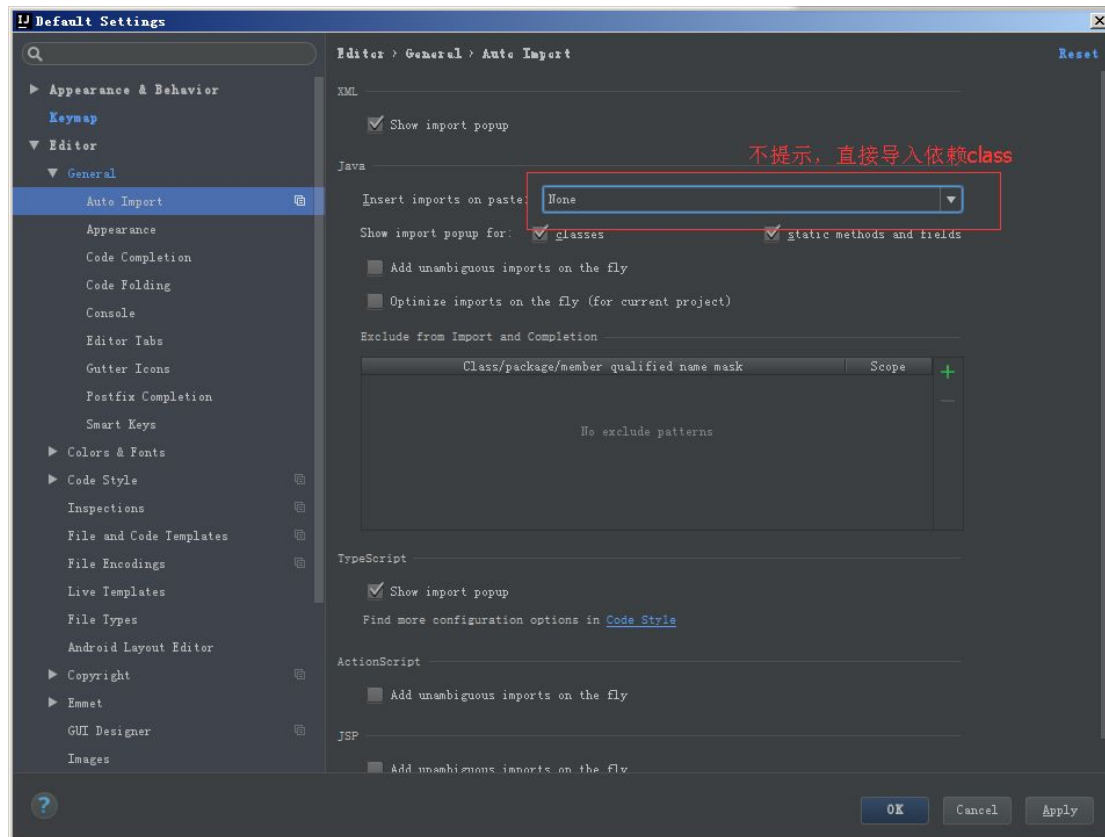


## Editor

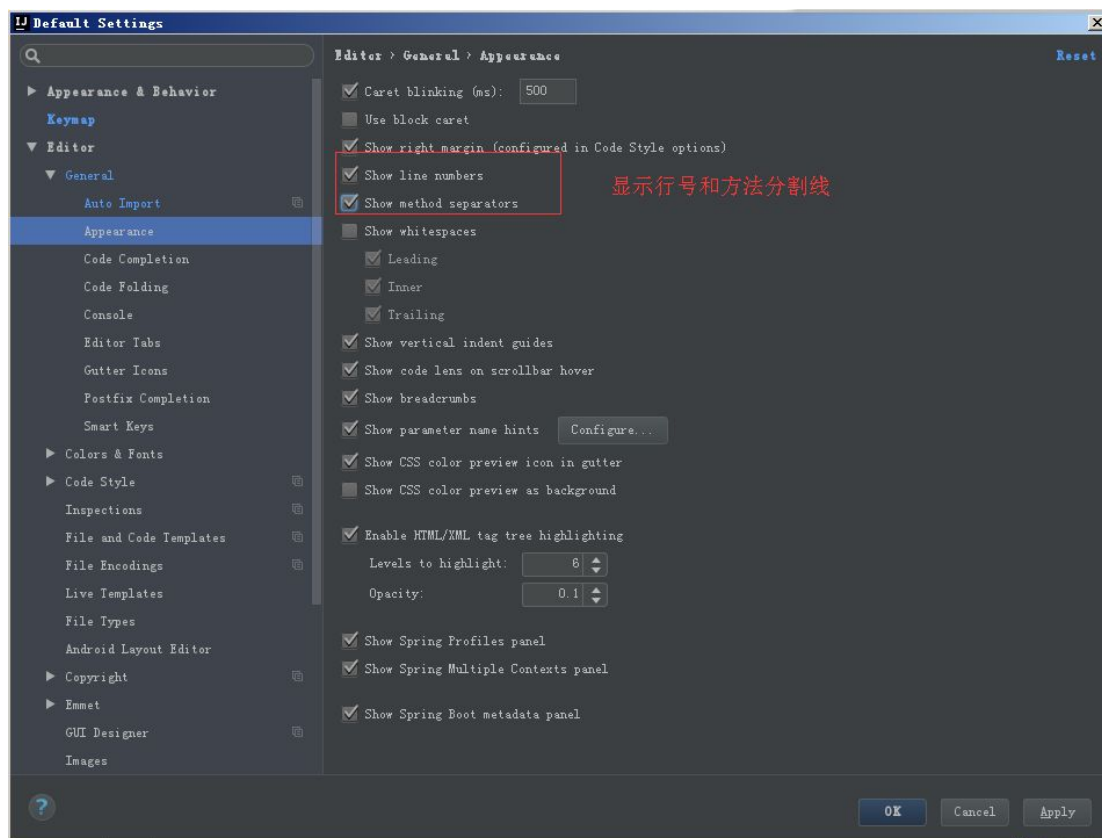
### General



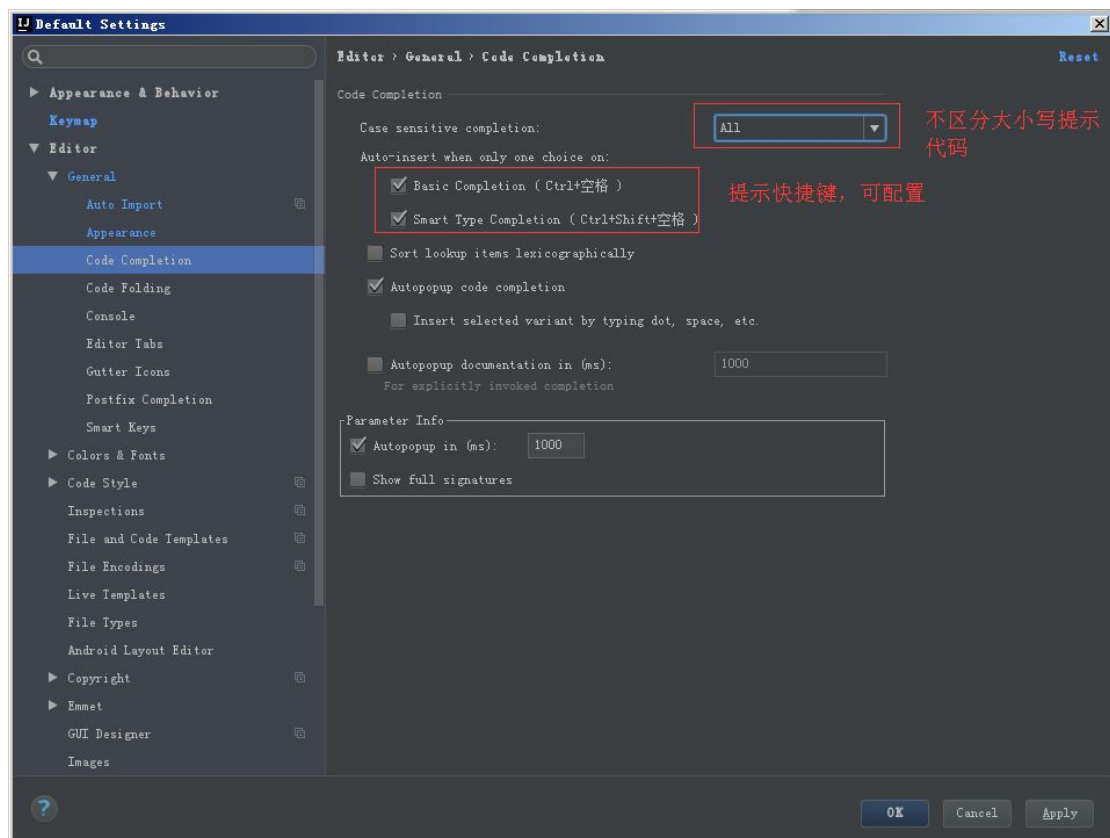
## Autoimporting



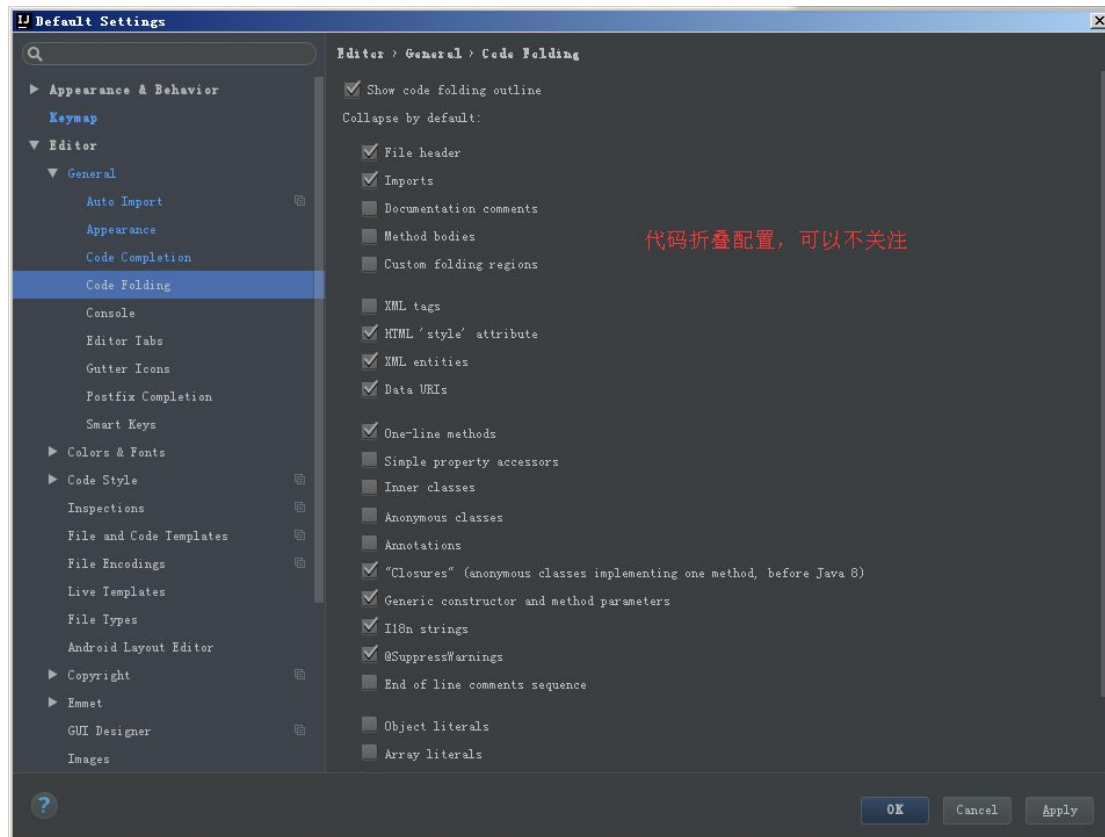
## Appearance



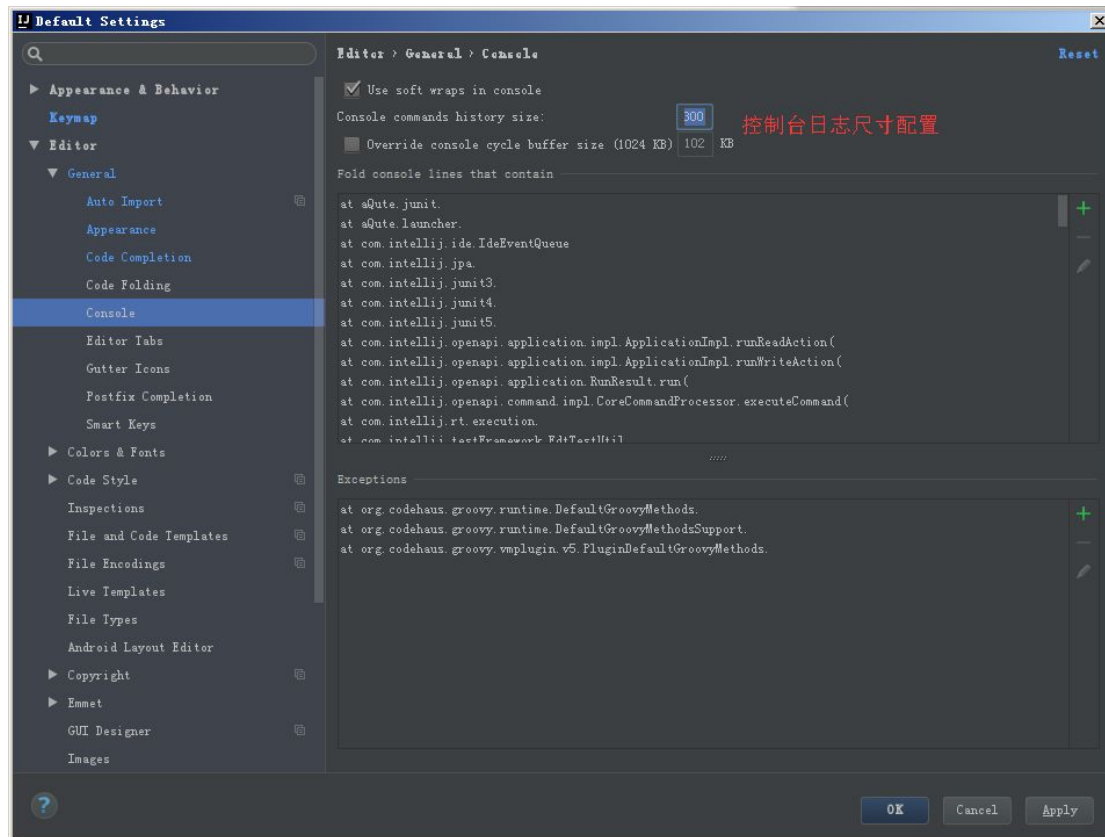
## Code Complete



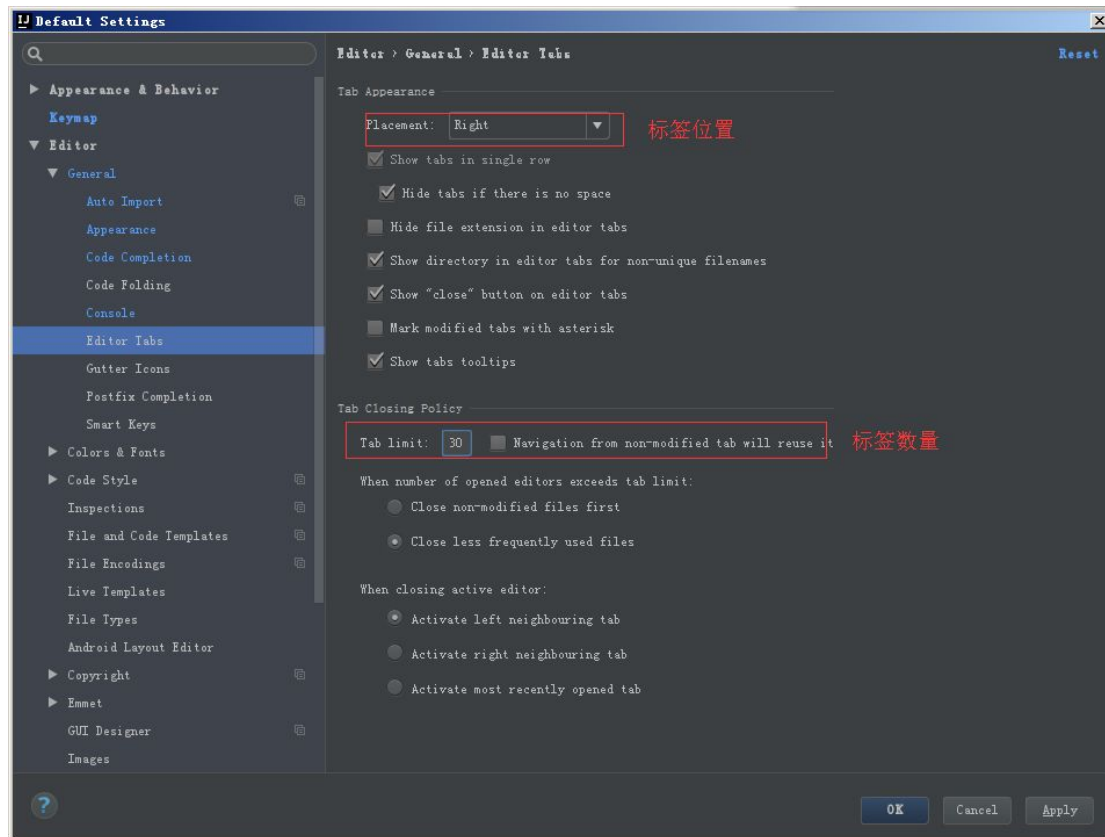
## Code Folding



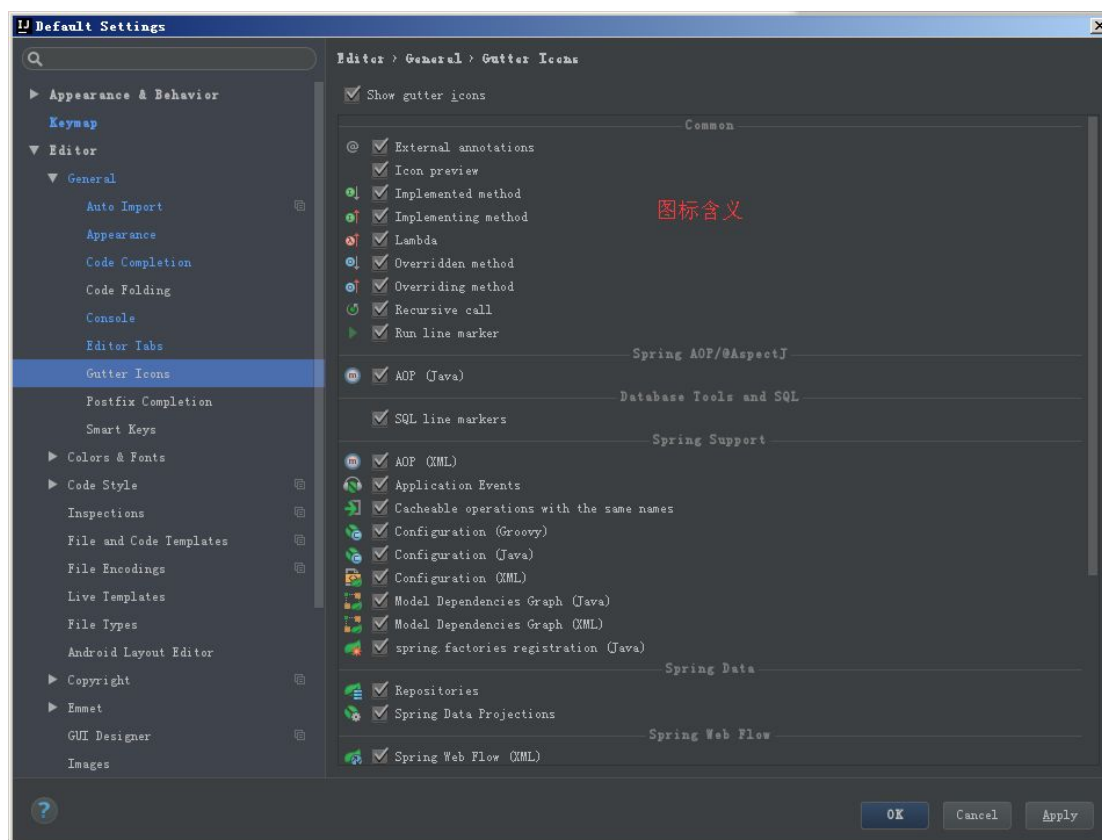
## Console



## Editor tab



## Gutter icons



## Postfix completion

保存默认即可。

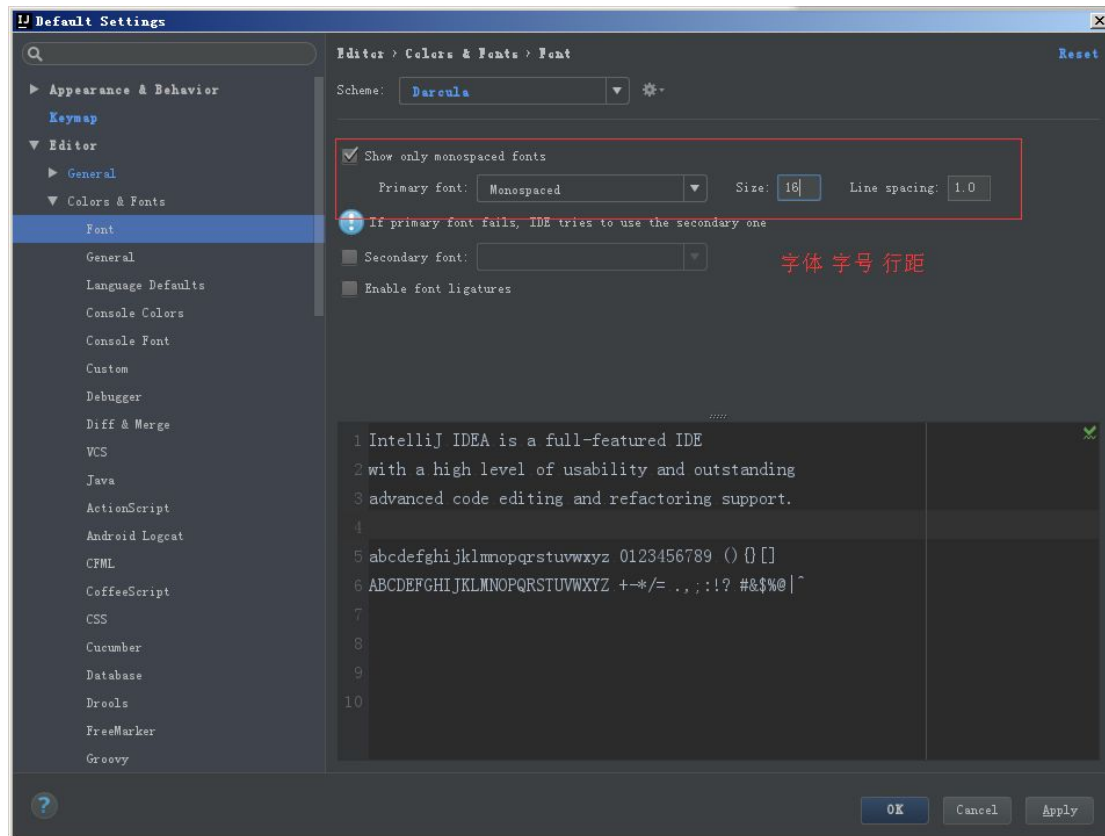
## Smart keys

保存默认即可。

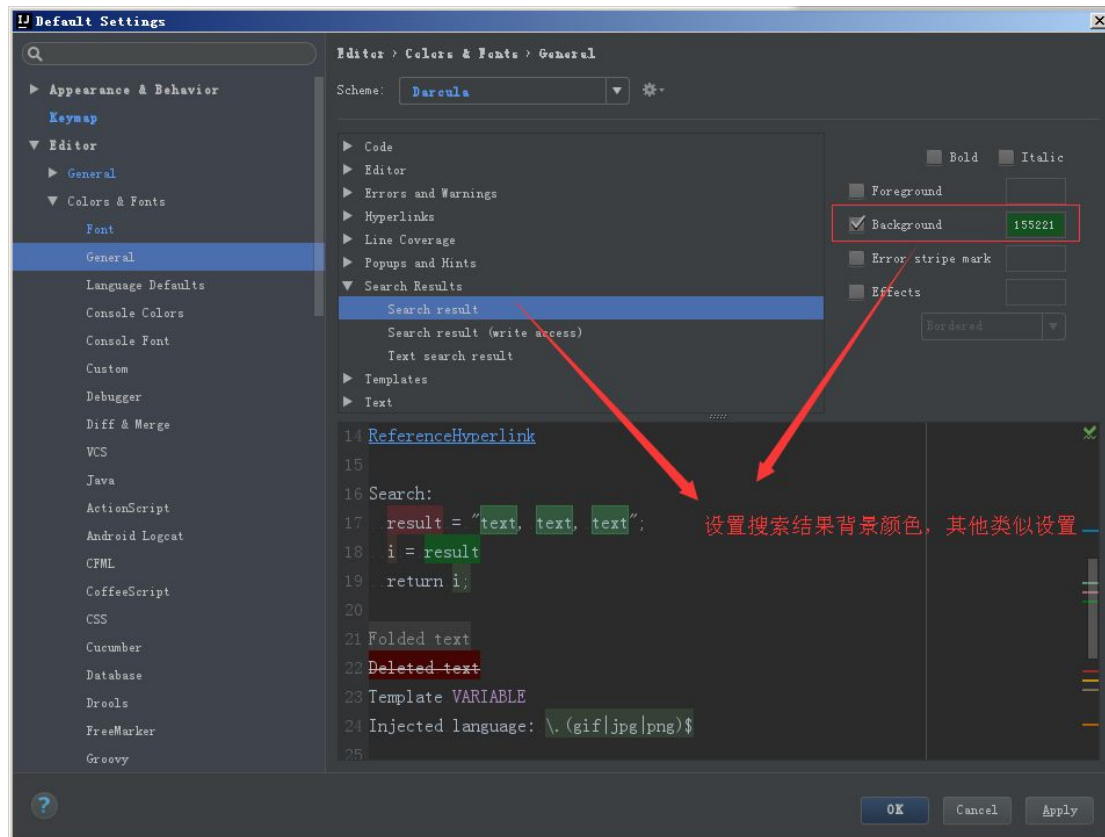


## Color&Fonts

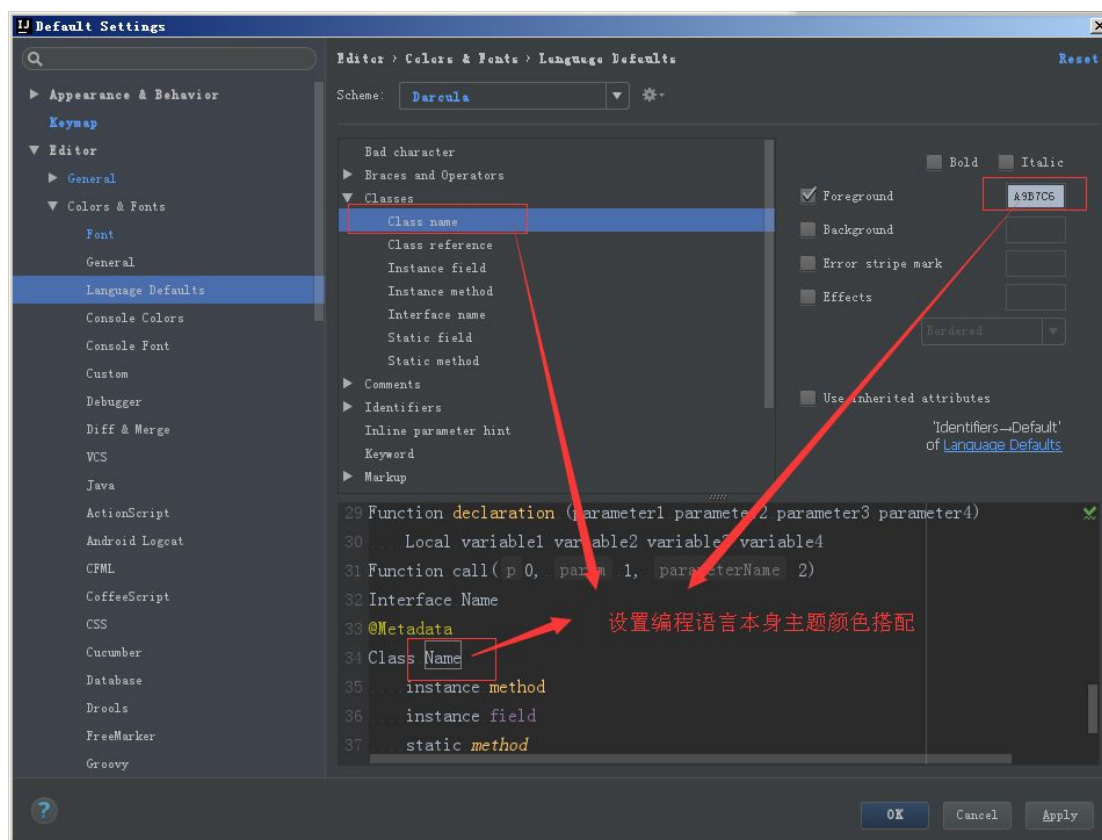
### Font



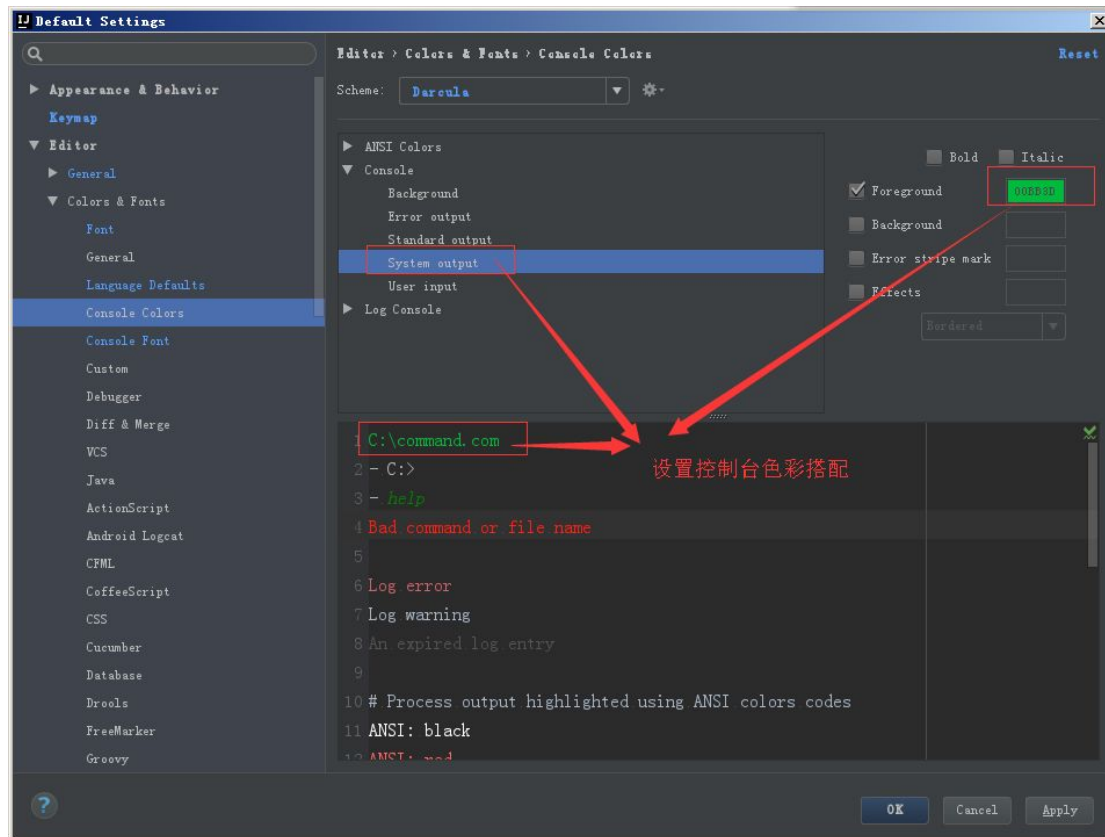
## General



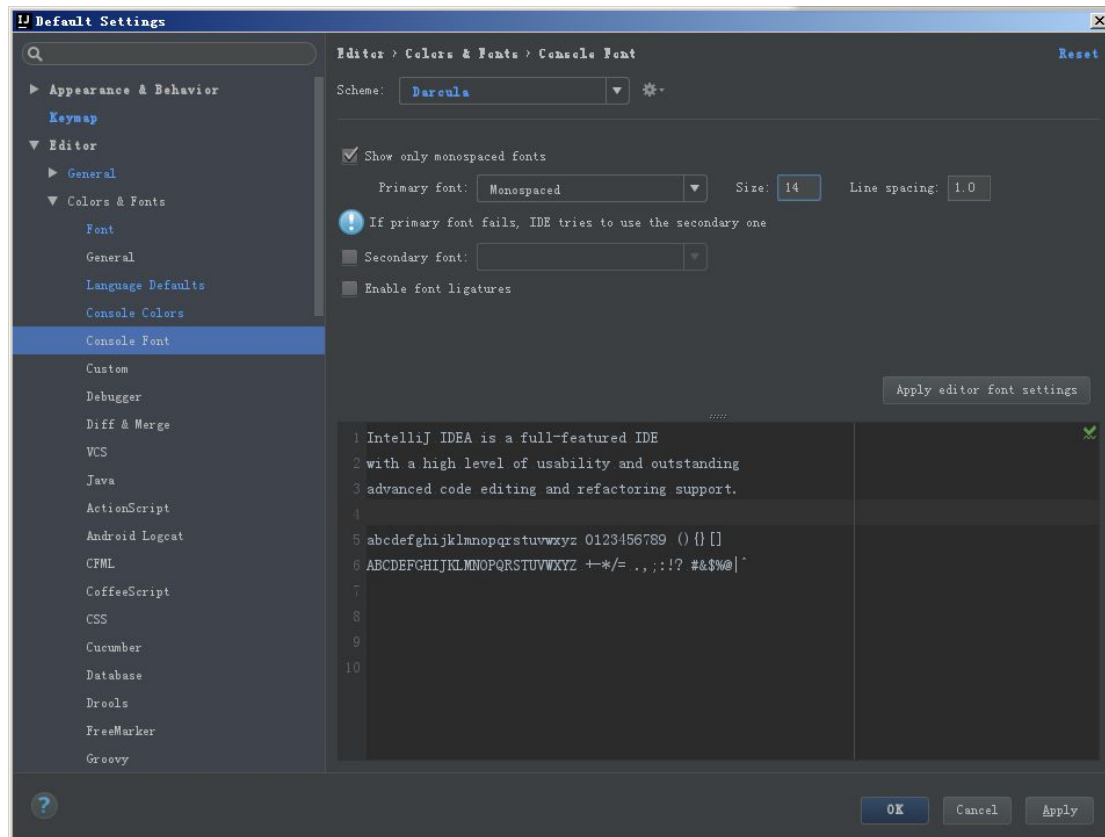
## Language Defaults



## Console Colors



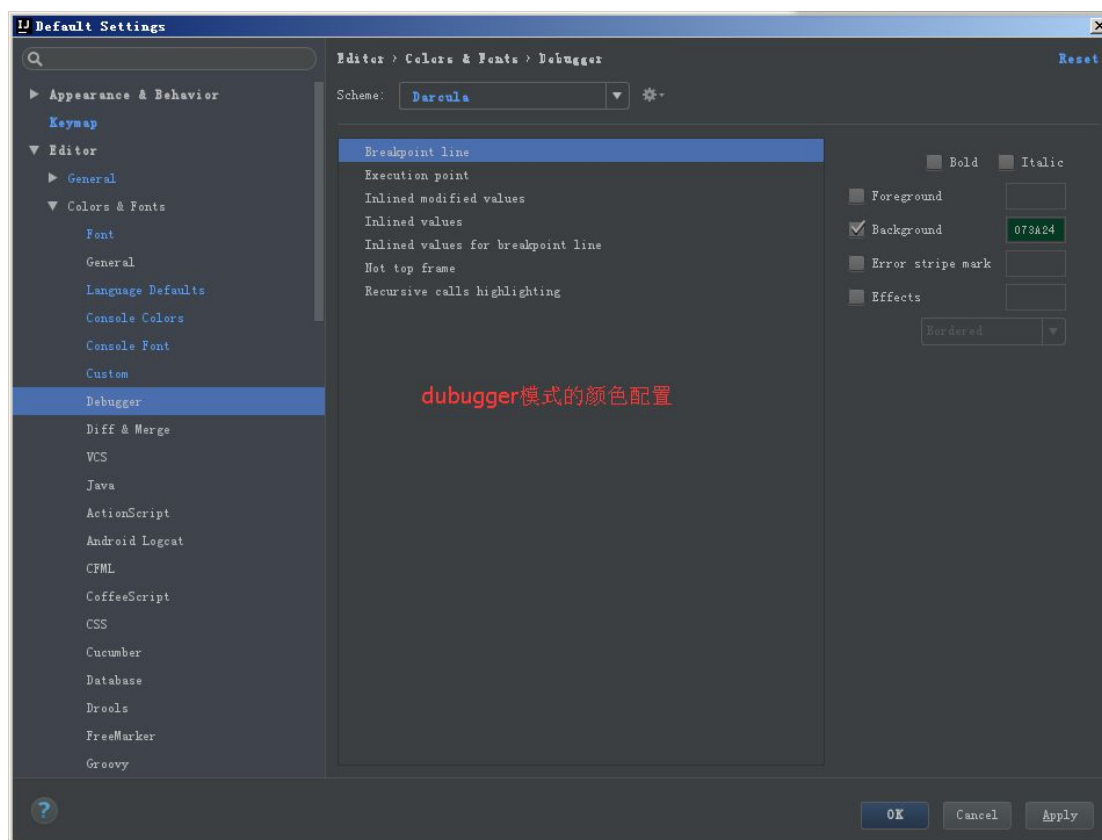
## ConsoleFont



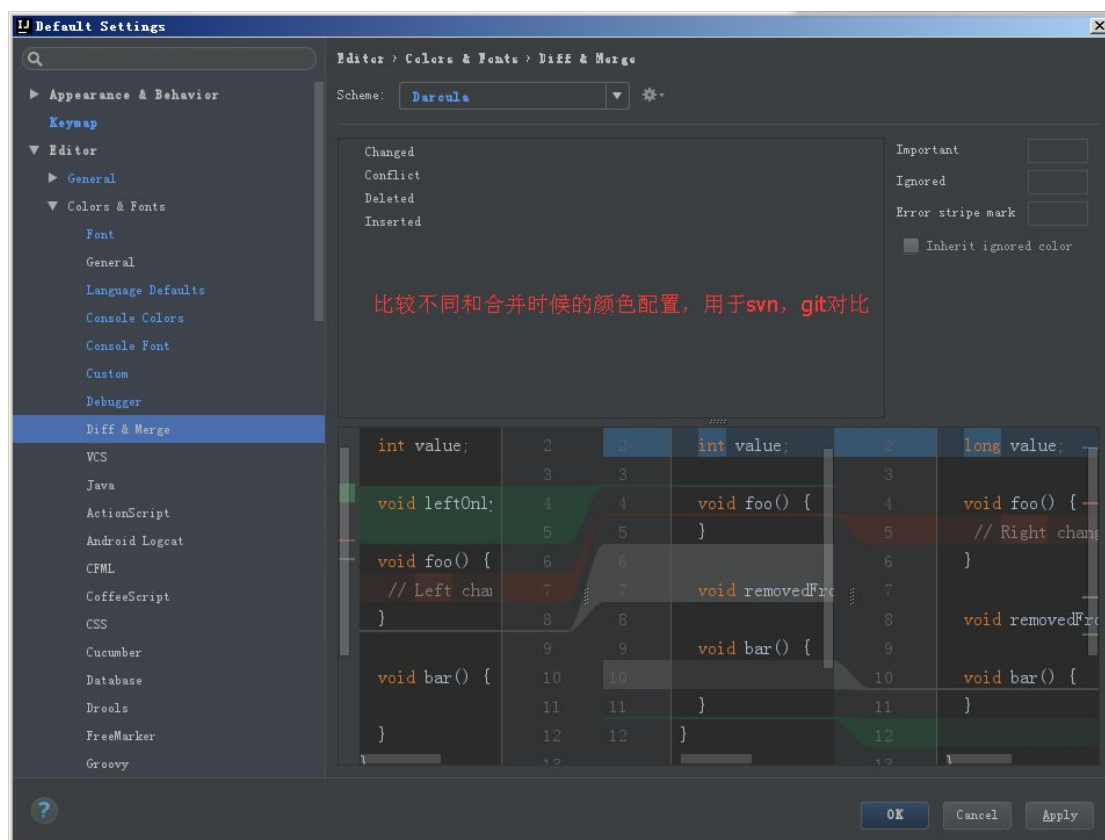
## Custom

定制的习惯配置。

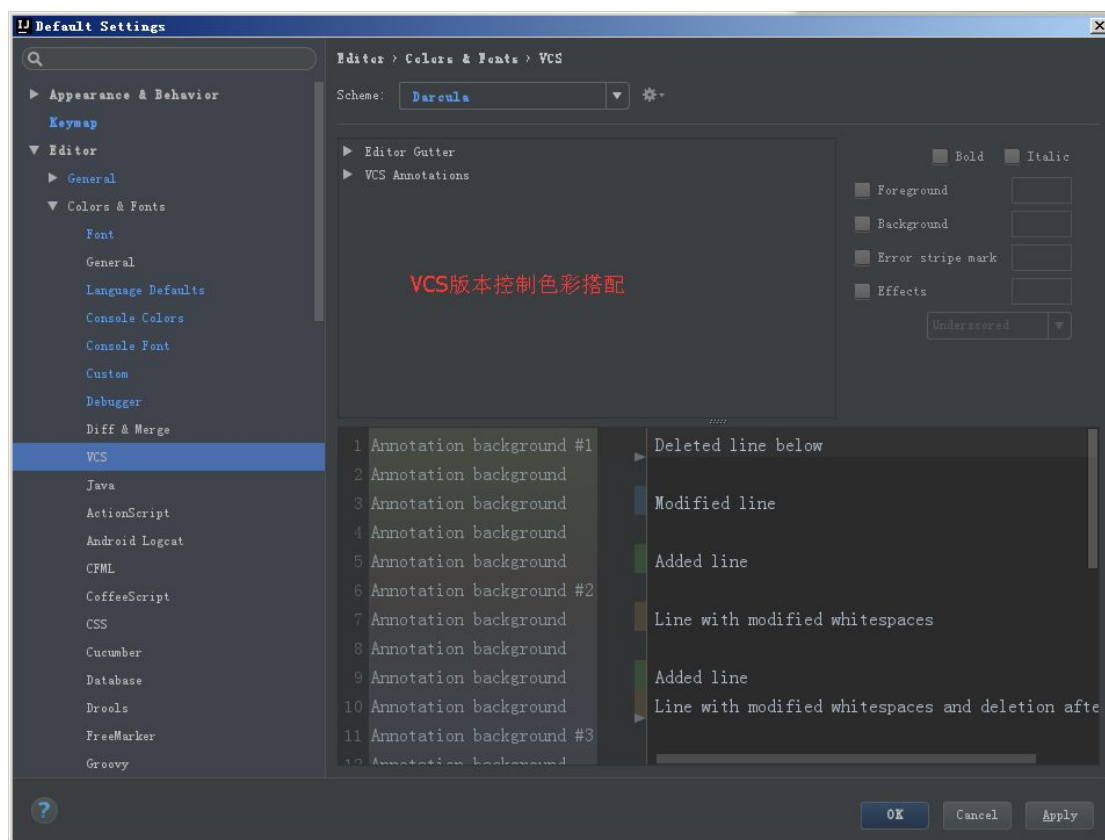
## Dubgger



## Diff&merge

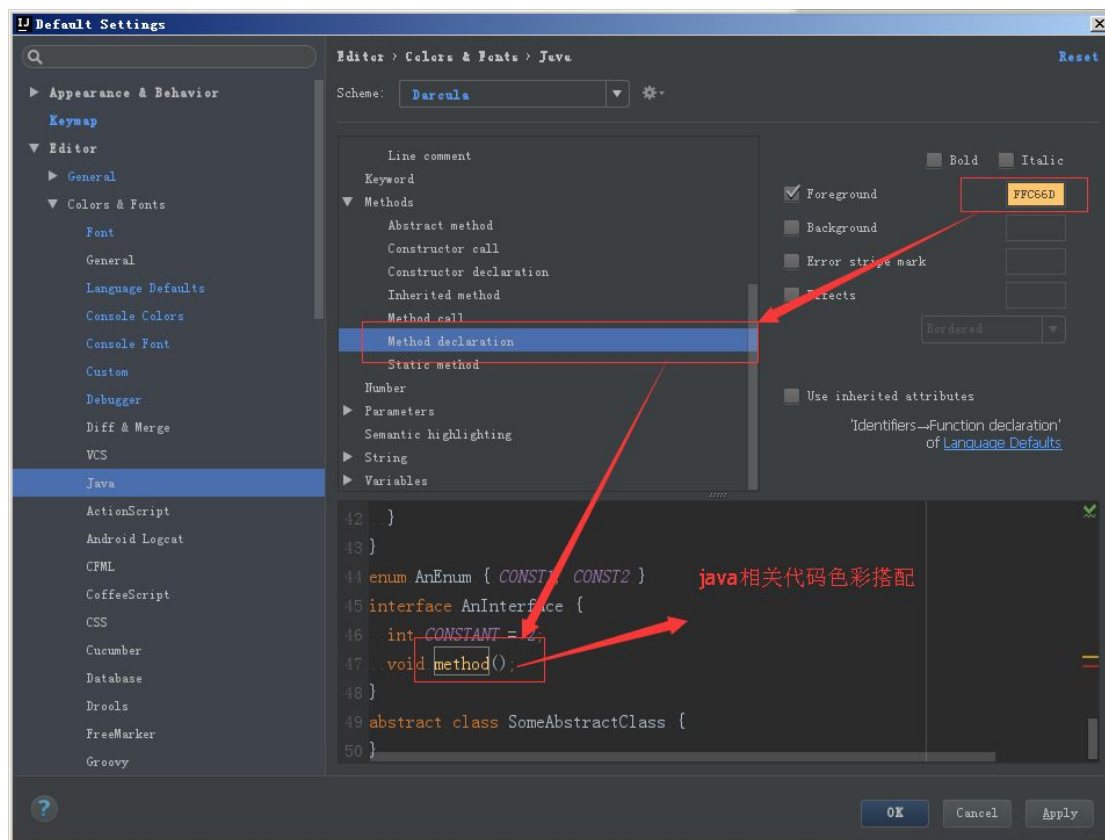


## Vcs





## Java



## ActionScript

As 脚本配置，默认即可

## Android Logcat

安卓日志配置，用的话可以配置

## CFML

ColdFusion Markup Language，默认配置即可

## CoffeeScript

支持的一种脚本语言，默认配置即可

## Css

Css 配置，默认即可

## Cucumber

**Cucumber** 是一个能够理解用普通语言 描述的测试用例的支持行为驱动开发（BDD）的自动化测试工具，用 Ruby 编写，支持 Java 和 .Net 等多种开发语言。

## Database

数据库色彩配置，默认即可

## Drools

Java 规则引擎色彩搭配，默认即可

## FreeMaker

模板语言色彩搭配，默认即可

## Groovy

默认即可

## Haml

默认即可

## Html

默认即可

## JavaScript

默认即可

## Jpa/hibernate QL

默认即可

## JSON

默认即可

## JSP

默认即可

## Kotlin

默认即可

## Less

默认即可

## Markdown

默认即可

## OGNL

默认即可

## OSGI Manifest

默认即可

## Properties

默认即可

## RegExp

默认即可

## Sass/SCSS

默认即可

## Spring EL

默认即可

## SQL

默认即可

## Stylus

默认即可

## Table Diff

默认即可

## **Tapestry**

默认即可

## **Tapestry**

默认即可

## **TypeScript**

默认即可

## **Velocity**

默认即可

## **XML**

默认即可

## **Xpath**

默认即可

## **XSLT**

默认即可

## **YAML**

默认即可

## **Spy-JS**

默认即可

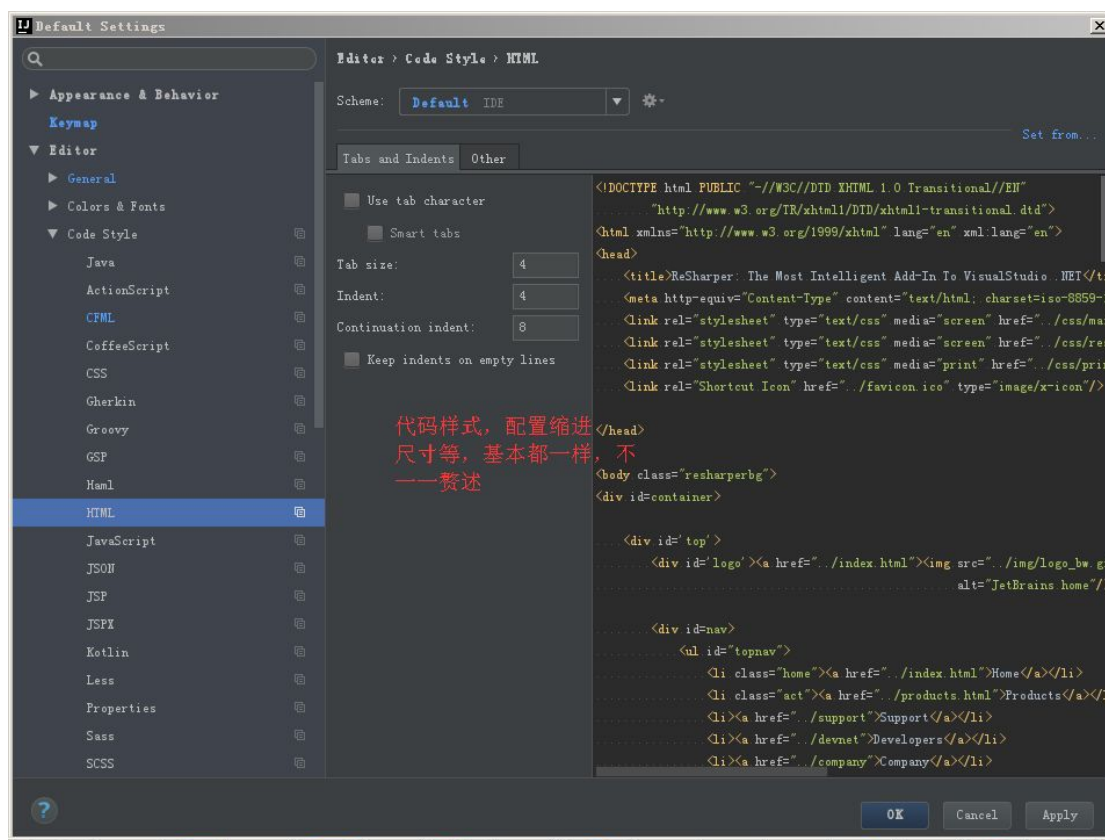
## File Status

默认即可

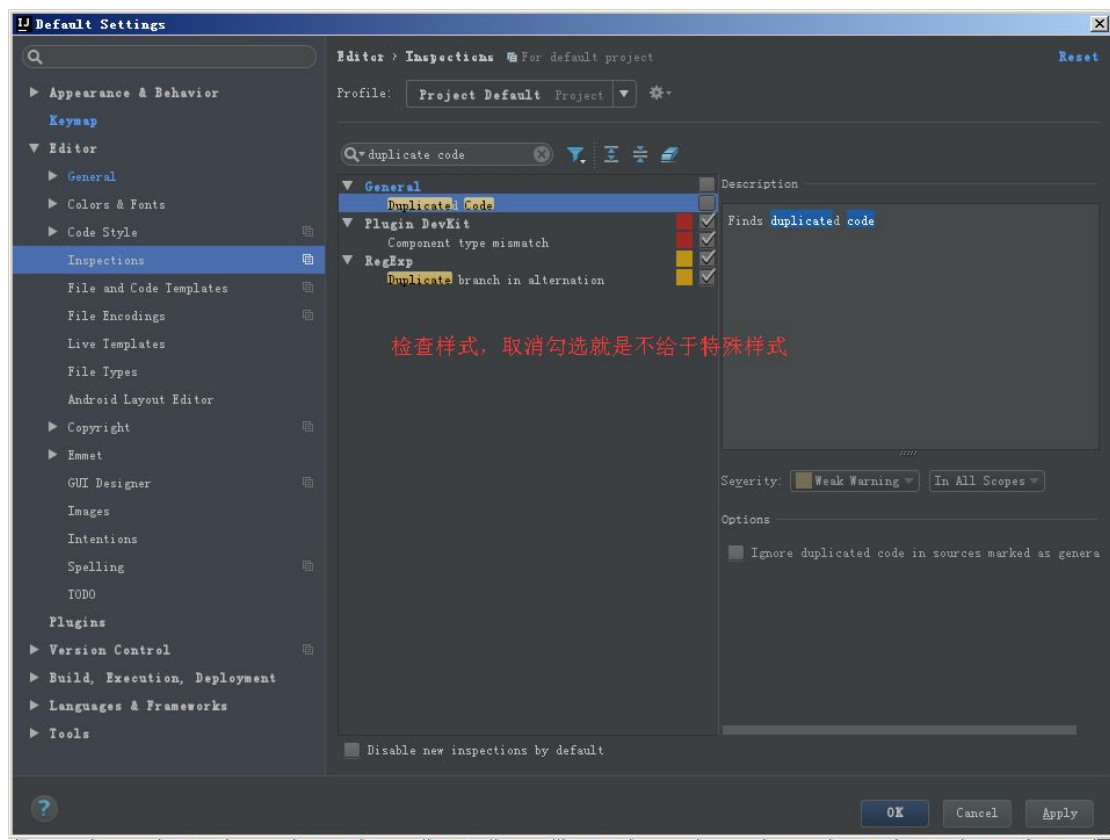
## By Scope

默认即可

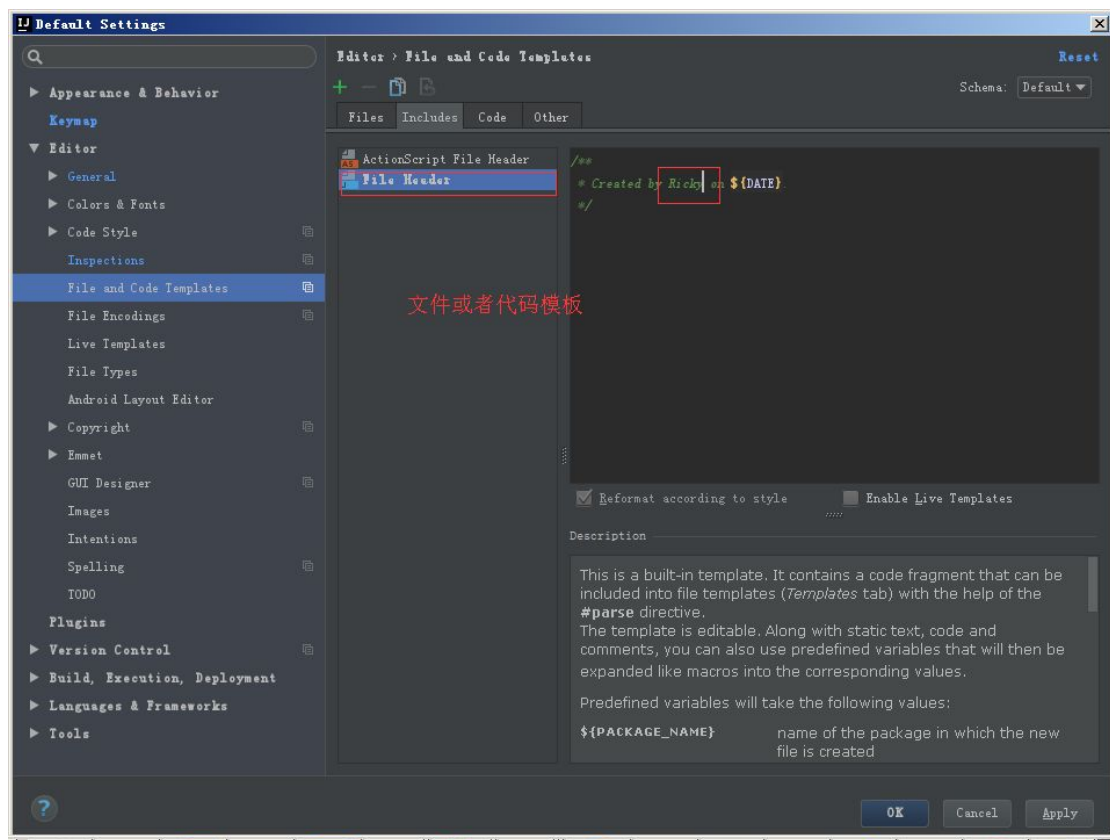
## CodeStyle



# Inspections

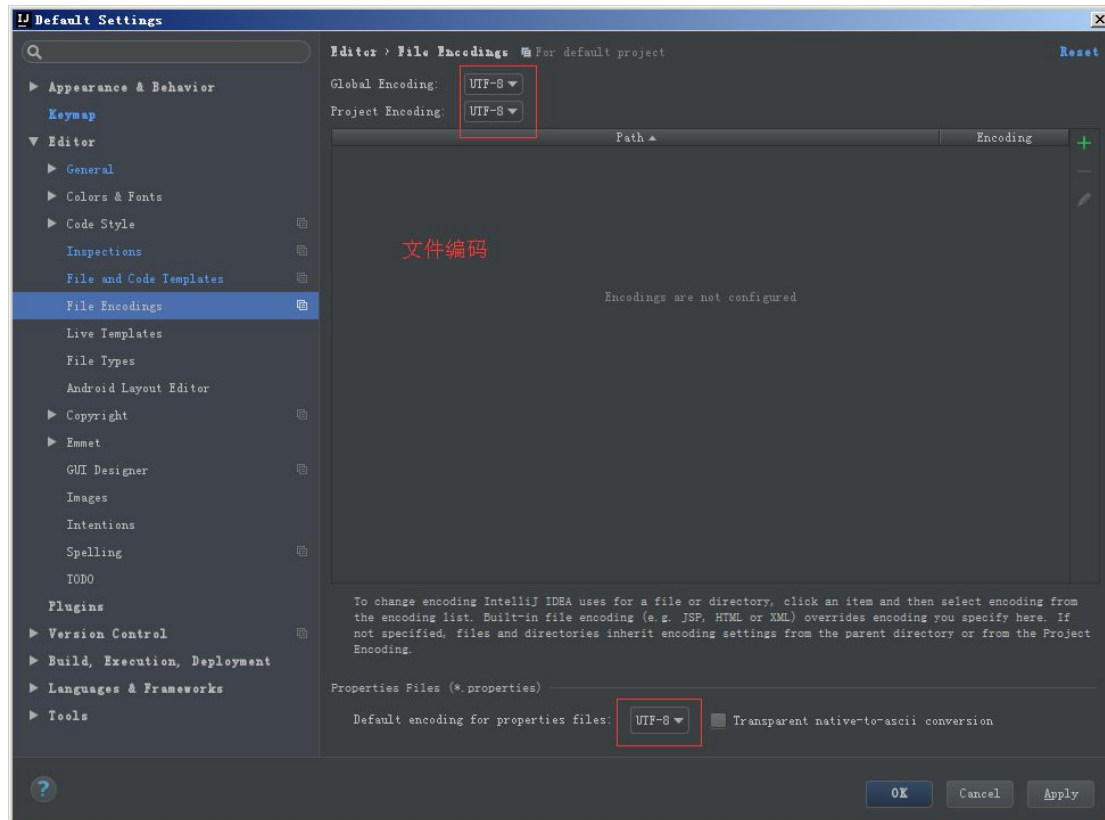


## File & Code Template

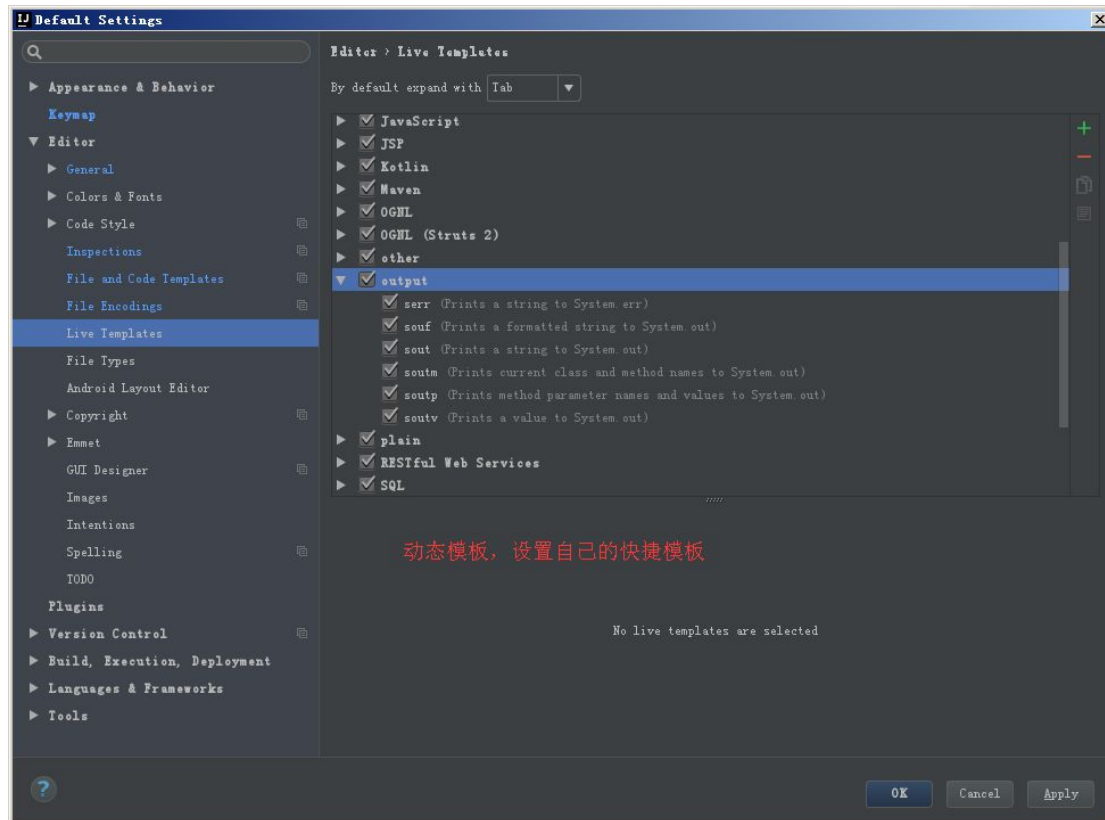




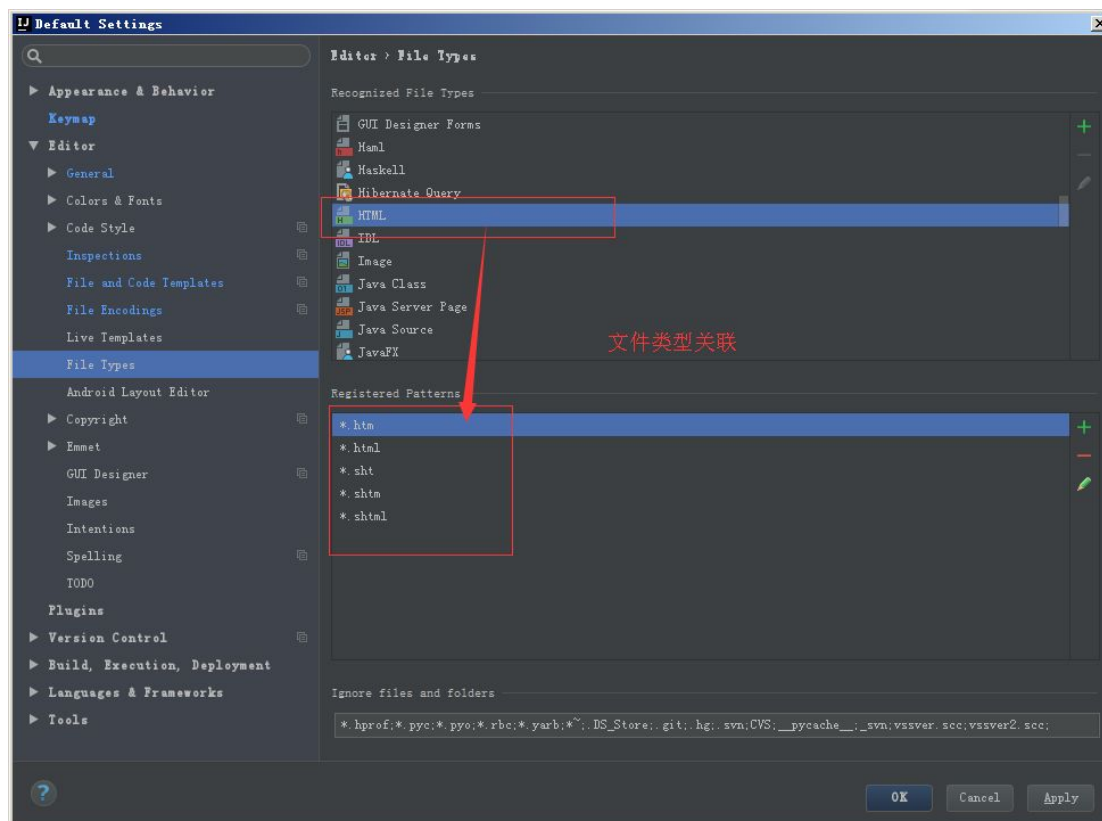
## File Encoding



## Live Template



## File Type



## Android Layout Editor

默认即可

## Copyright

默认即可

## Emmet

Emmet 语法

## GUI Designer

默认即可

## Images

图片配置，默认即可

## Intentions

意图，打算，默认即可

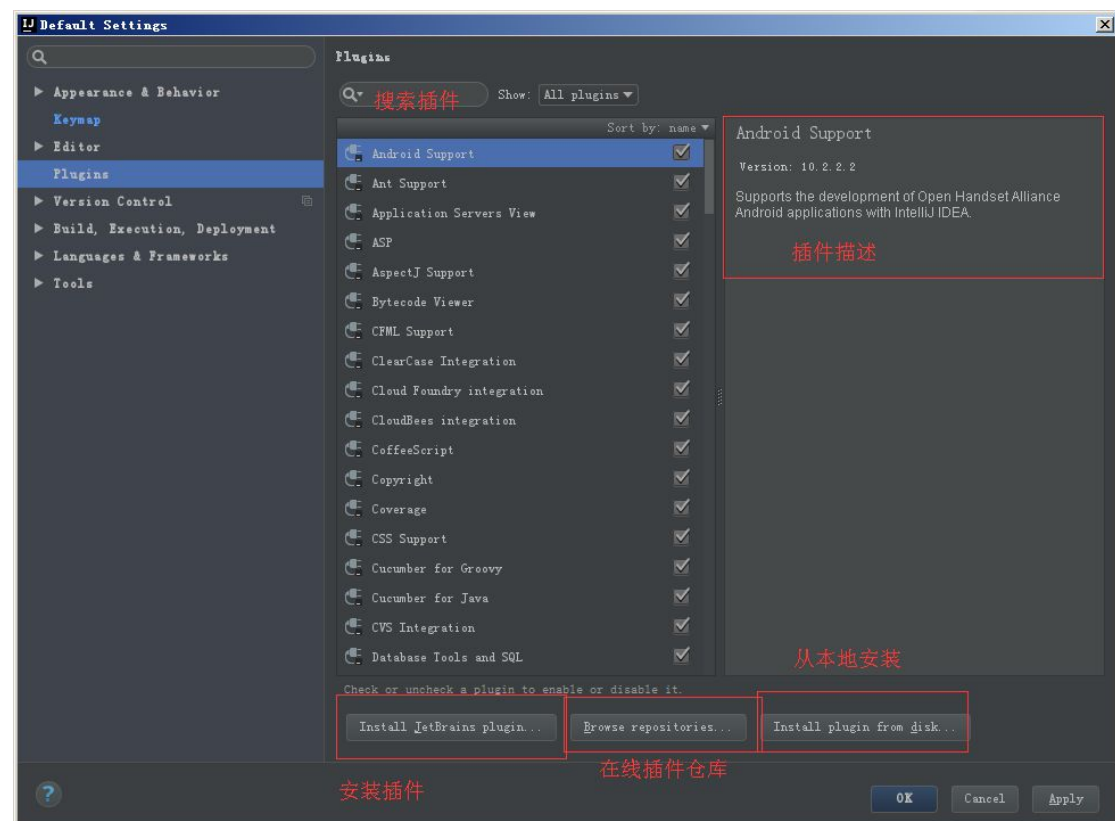
## Spelling

拼写，默认即可

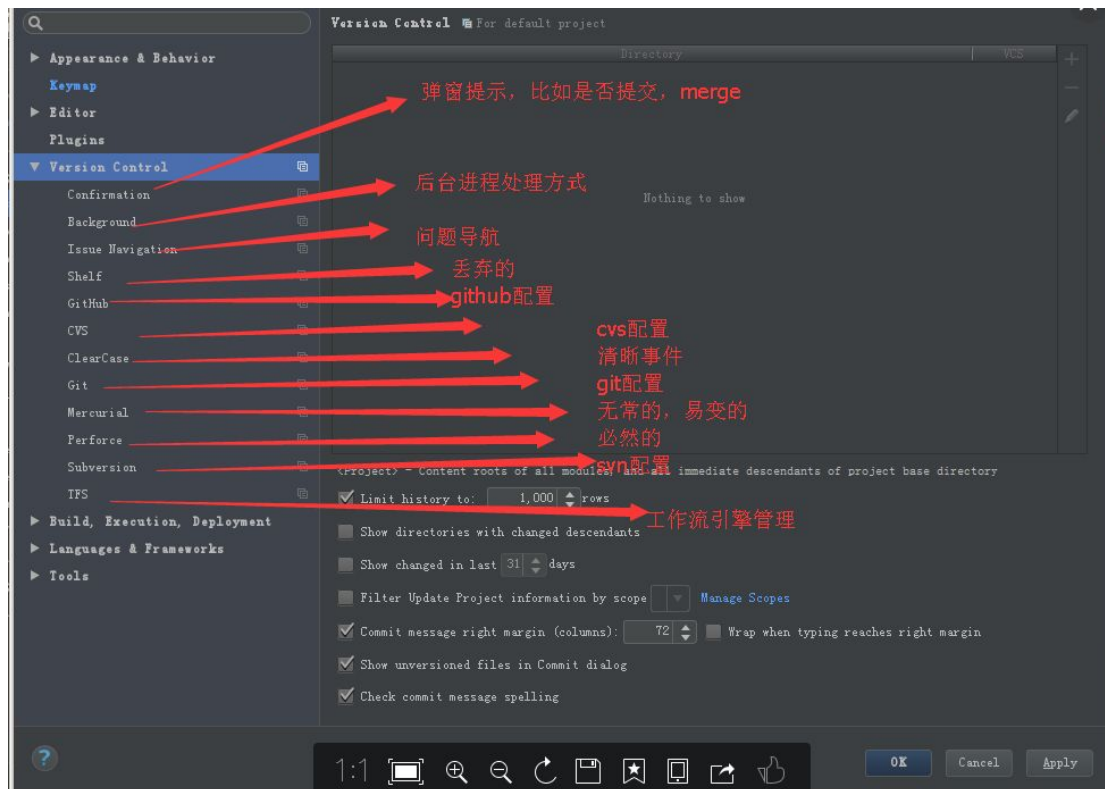
## TODO

待办事项，默认即可

## Plugins

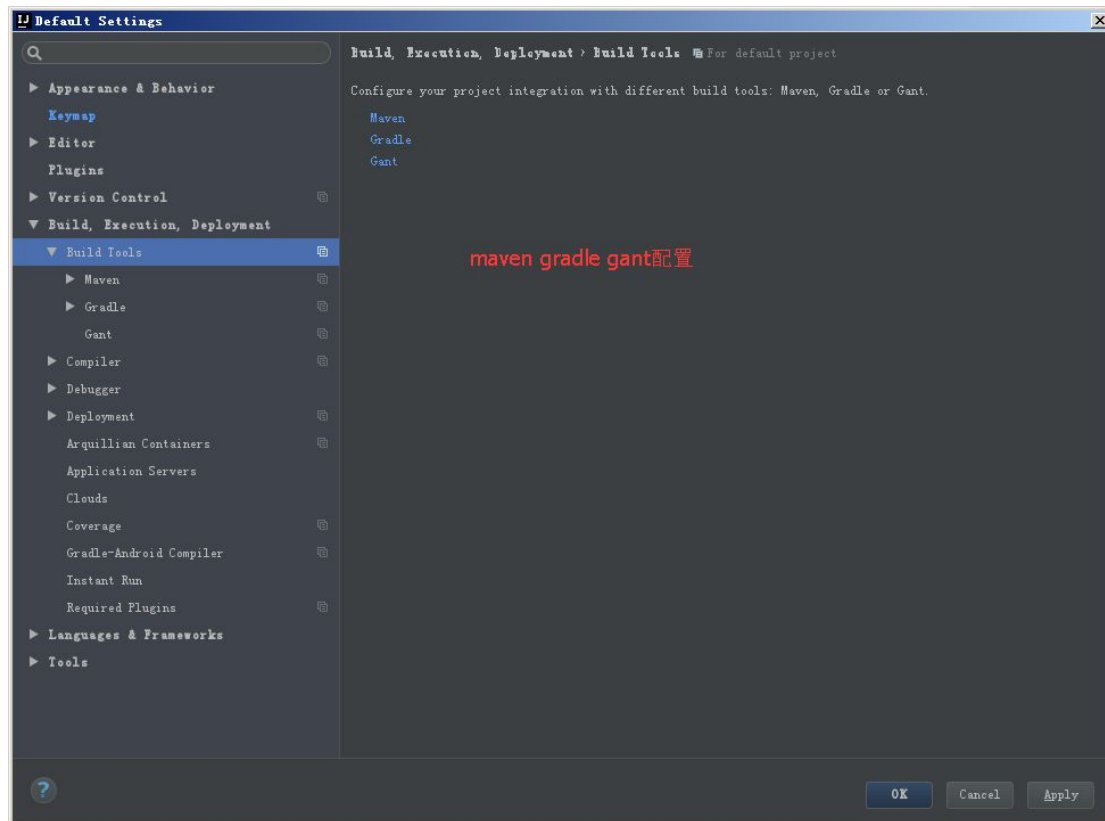


## Version Control

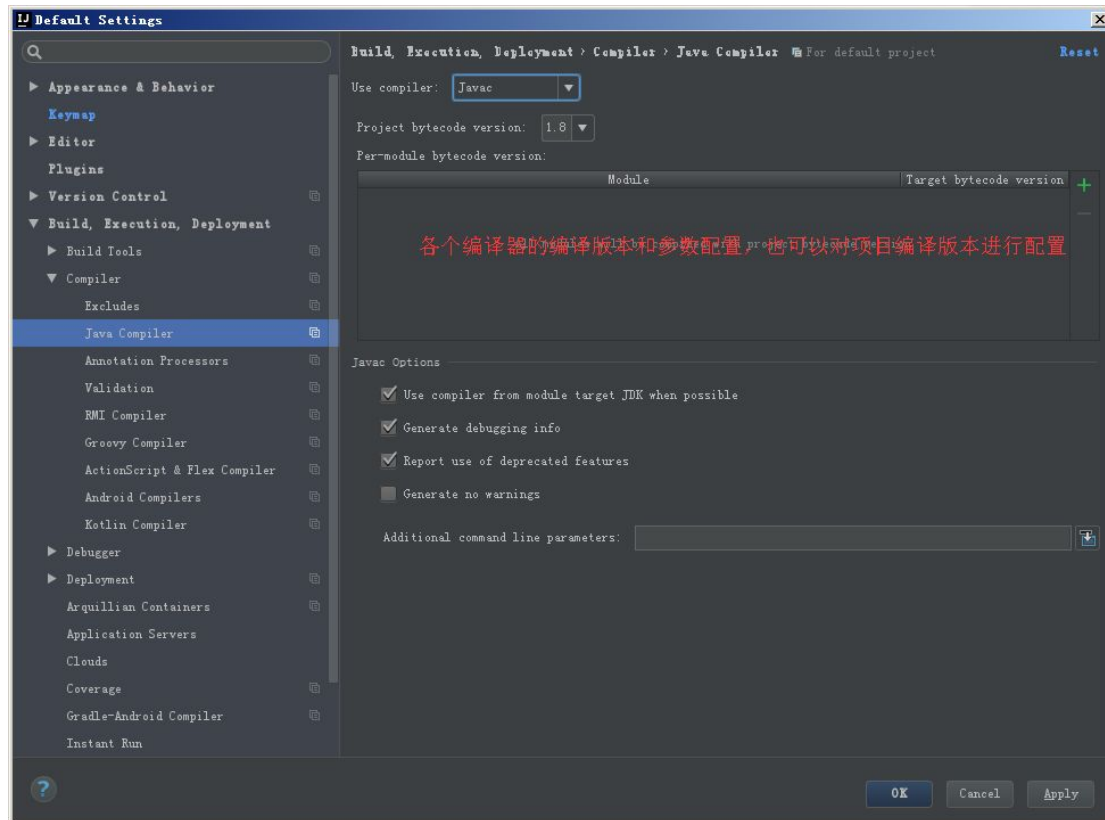


## Build Execution Deployment

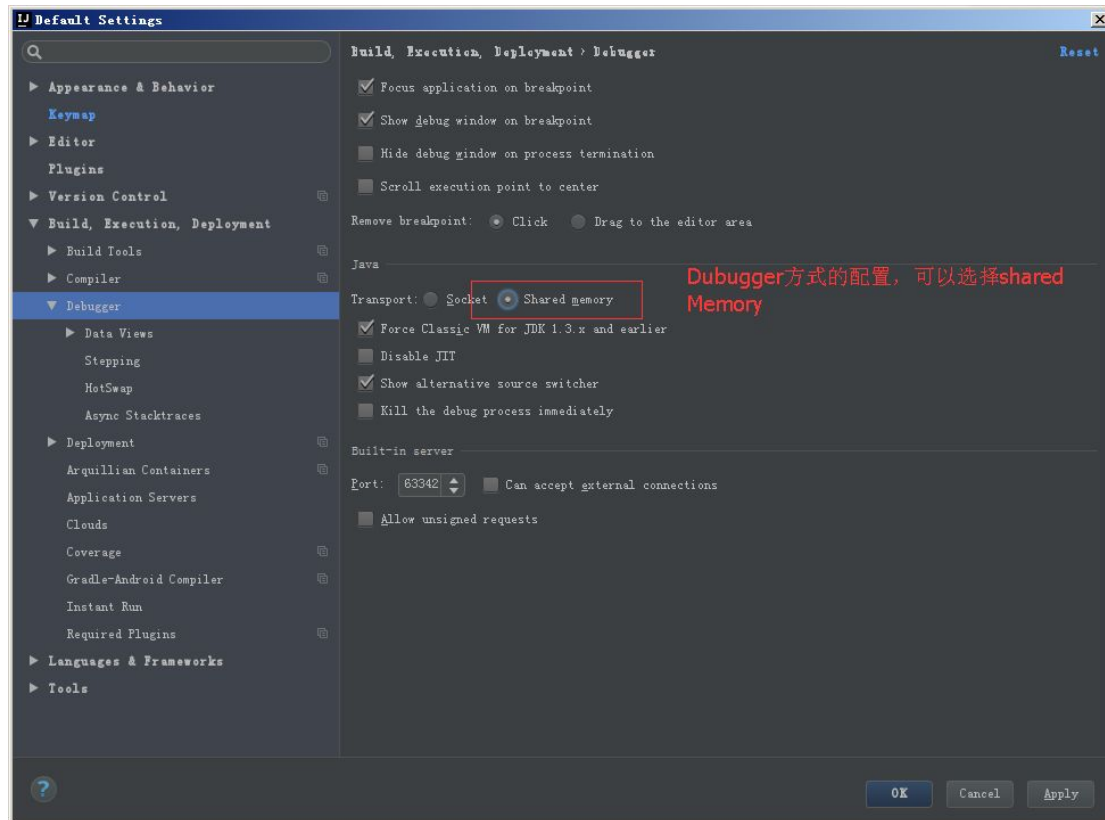
### Build Tools



## Compiler

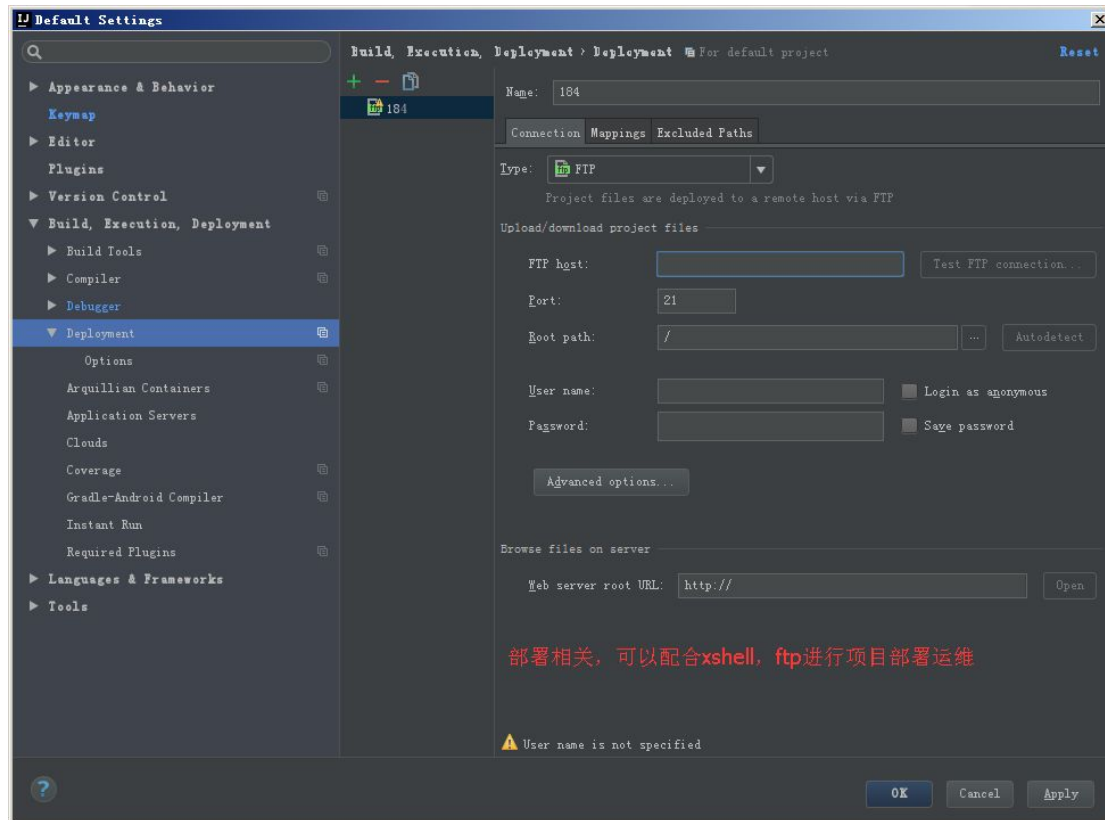


## Dubugger

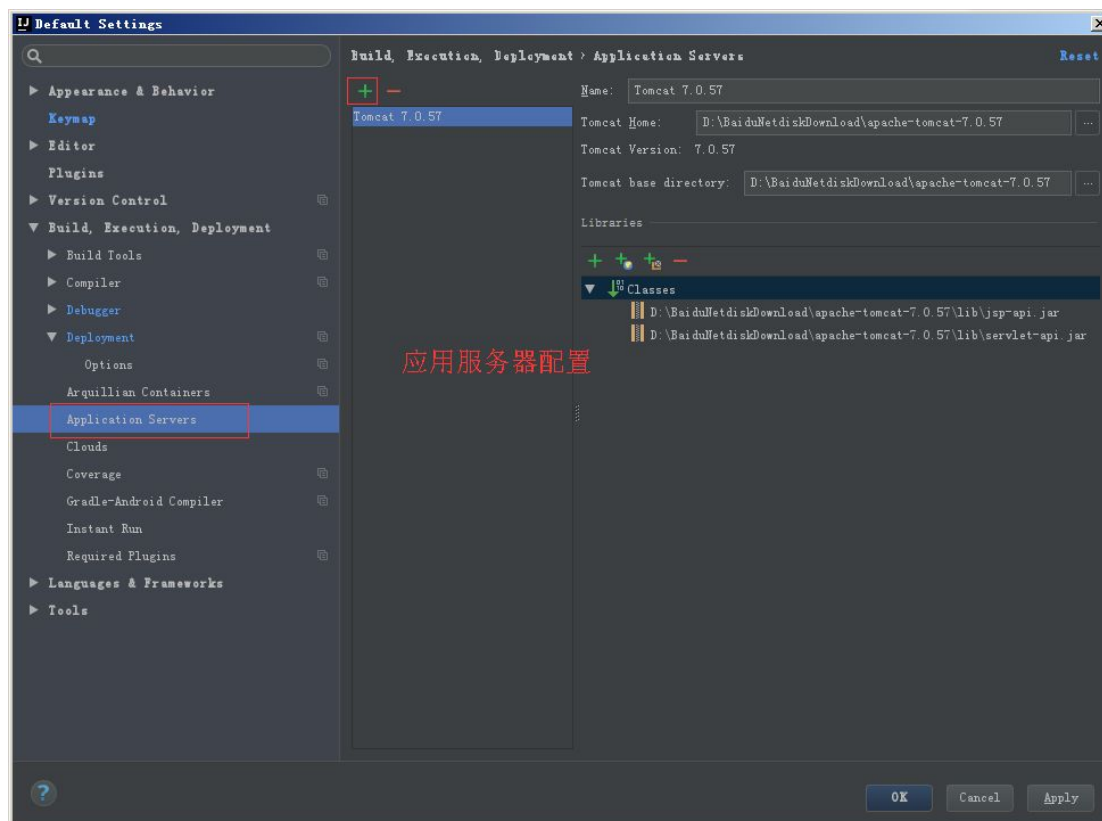




## DeployMent



## ApplicationServer

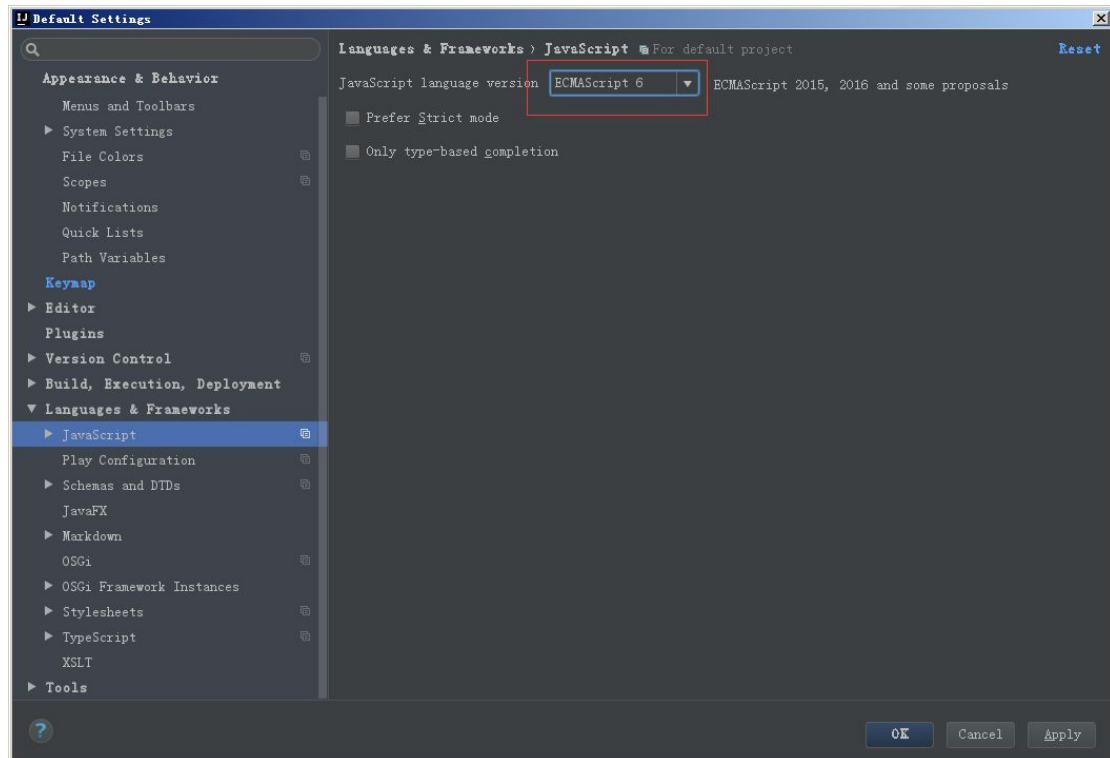


## 其他

其他默认即可

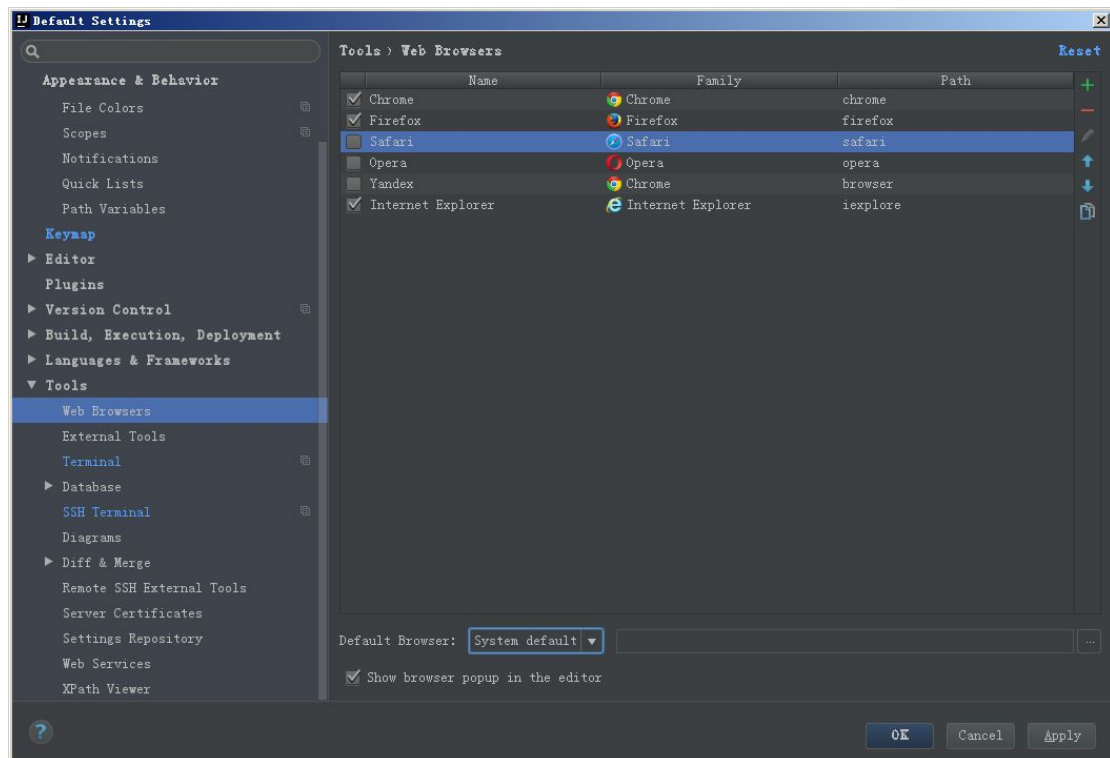
## Languages&Frameworks

此模块默认配置即可，功能是配置语言编译版本，比如下面

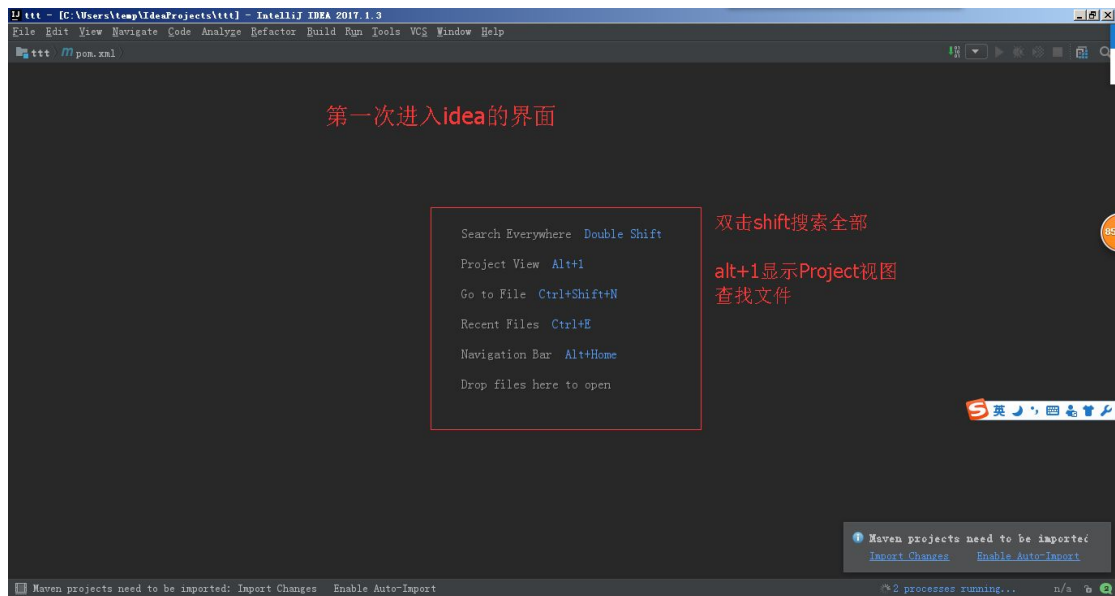


## Tools

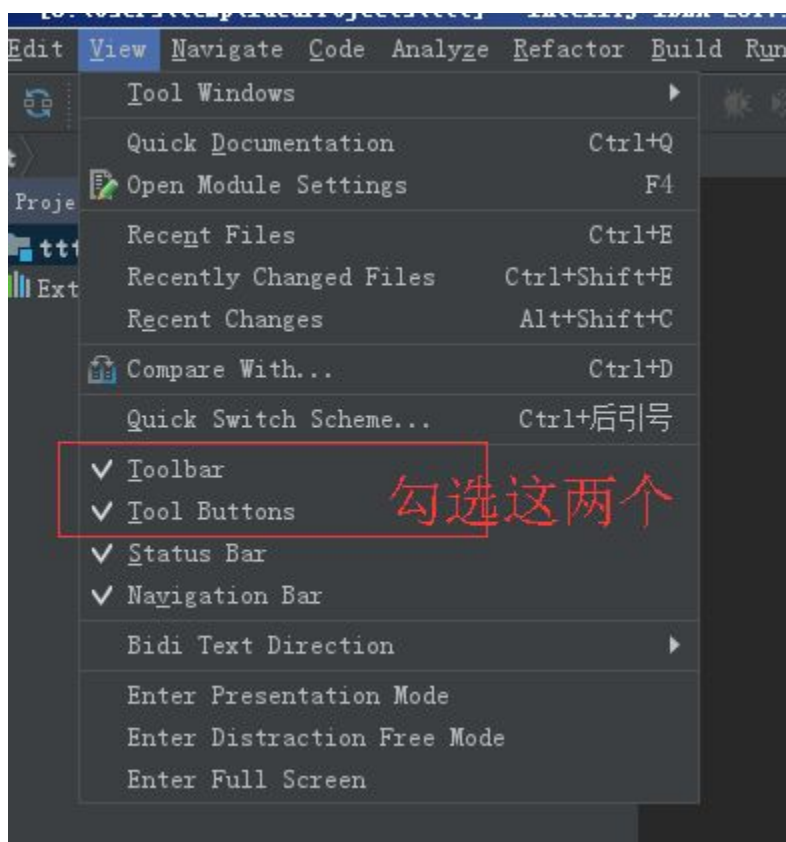
常用工具，默认即可



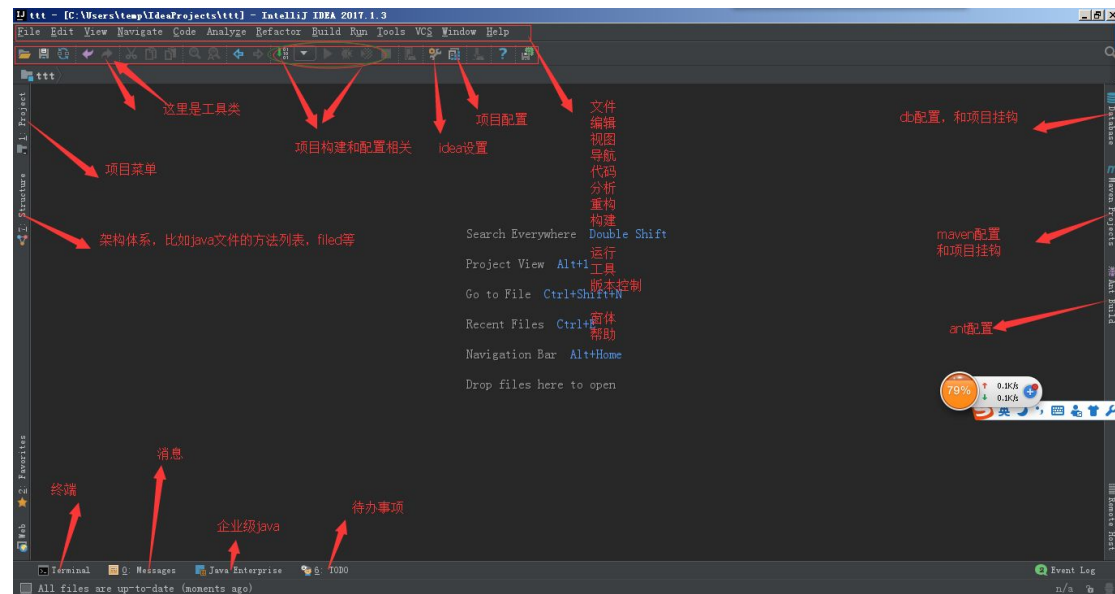
## 第一次启动后



## 调出面板和按钮组

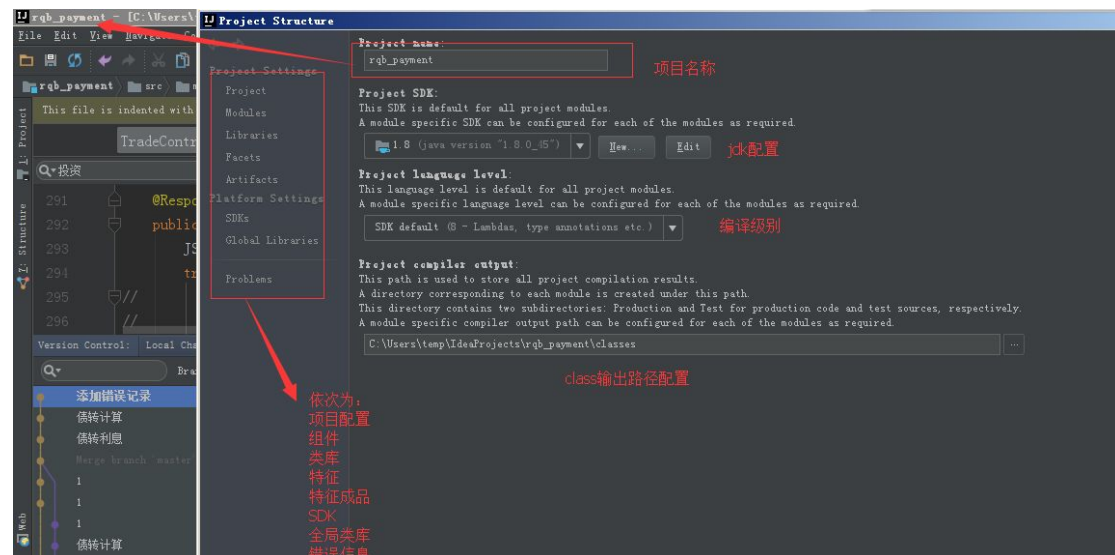


## 面板说明

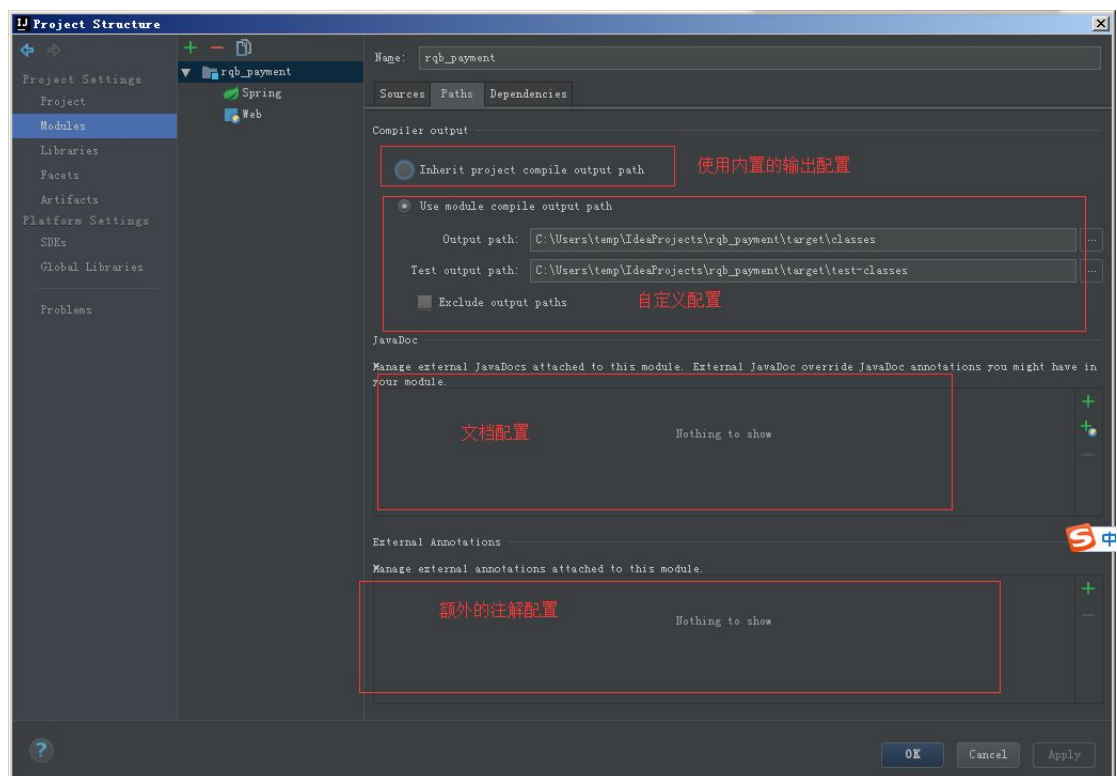
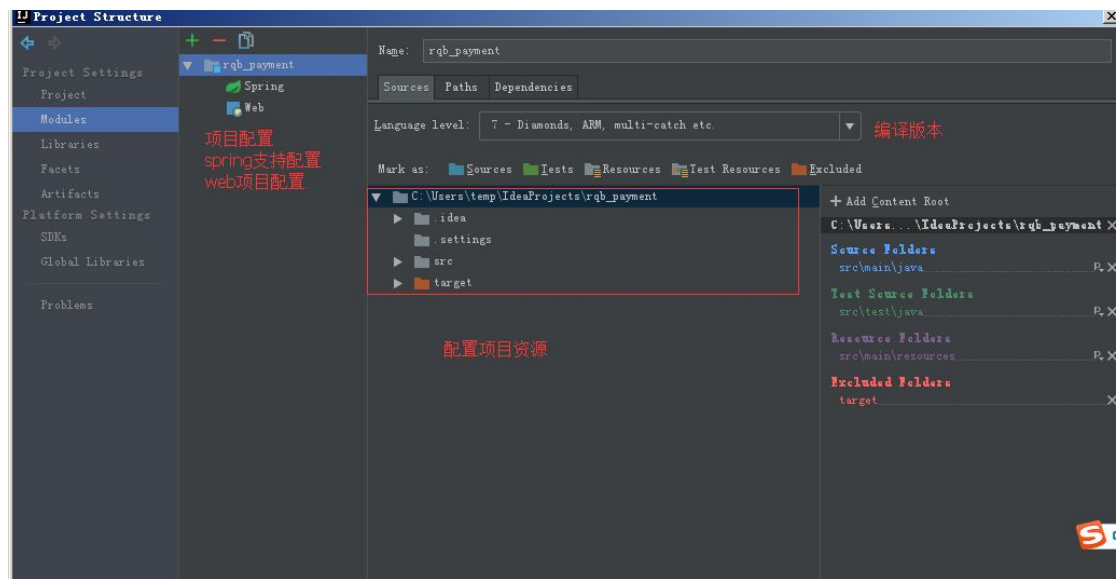


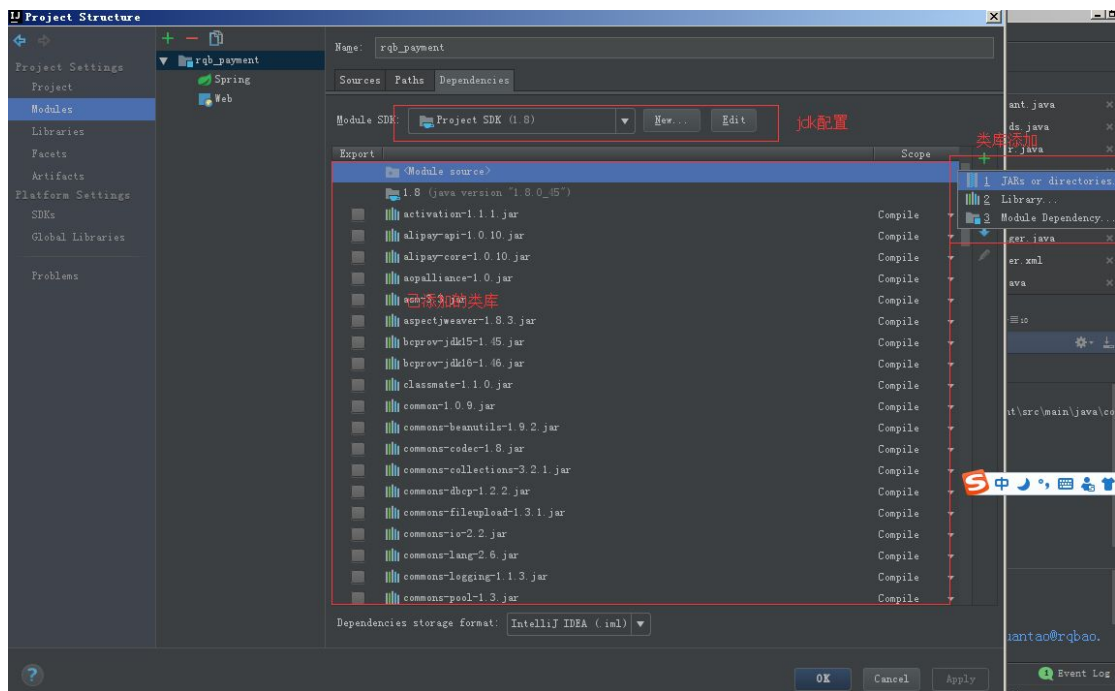
## 项目配置

### Project

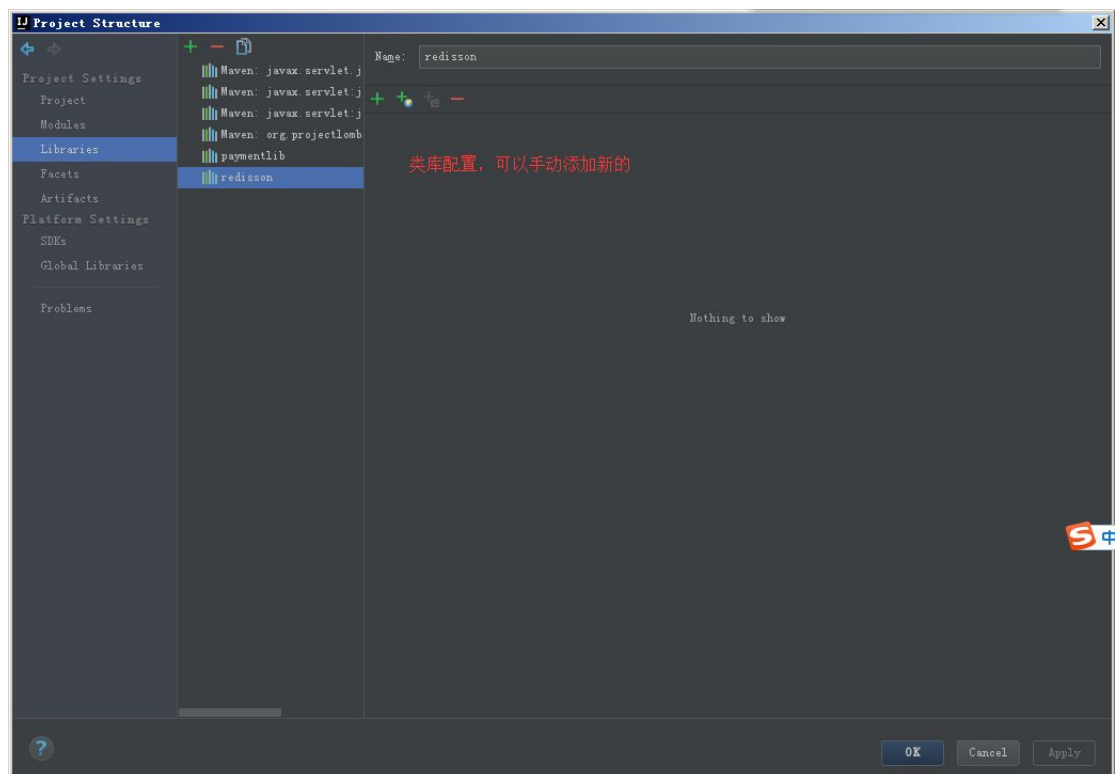


# Modules



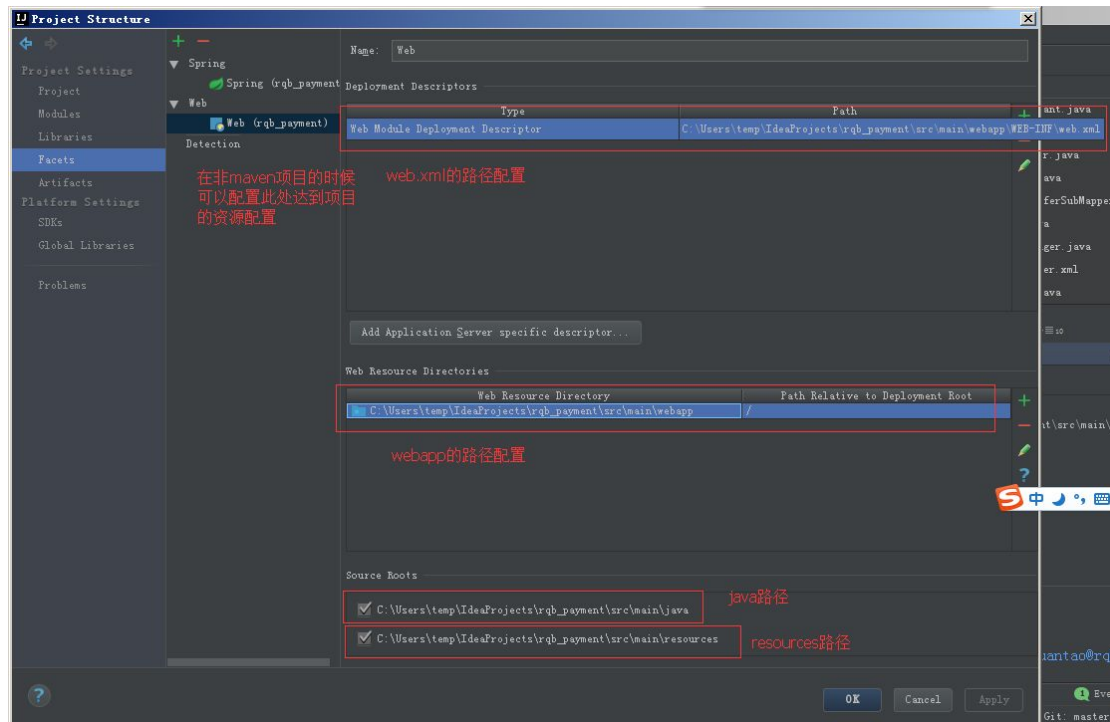


## Libraries

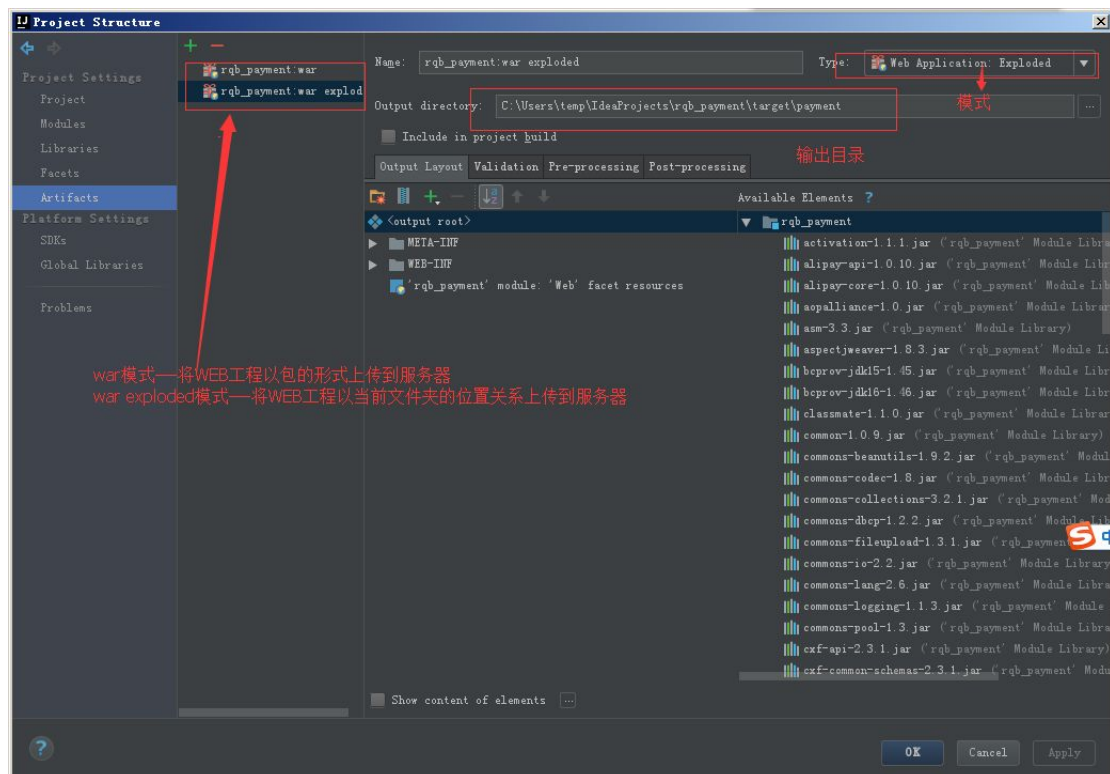




## Facets

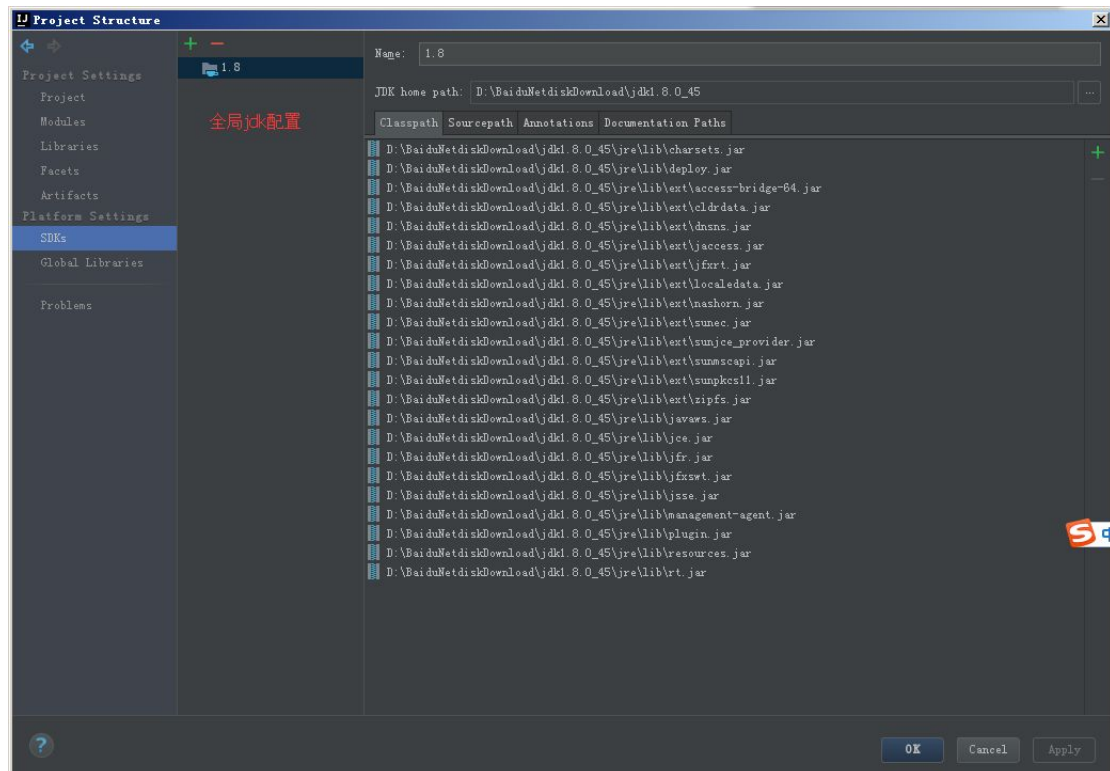


## Artifacts

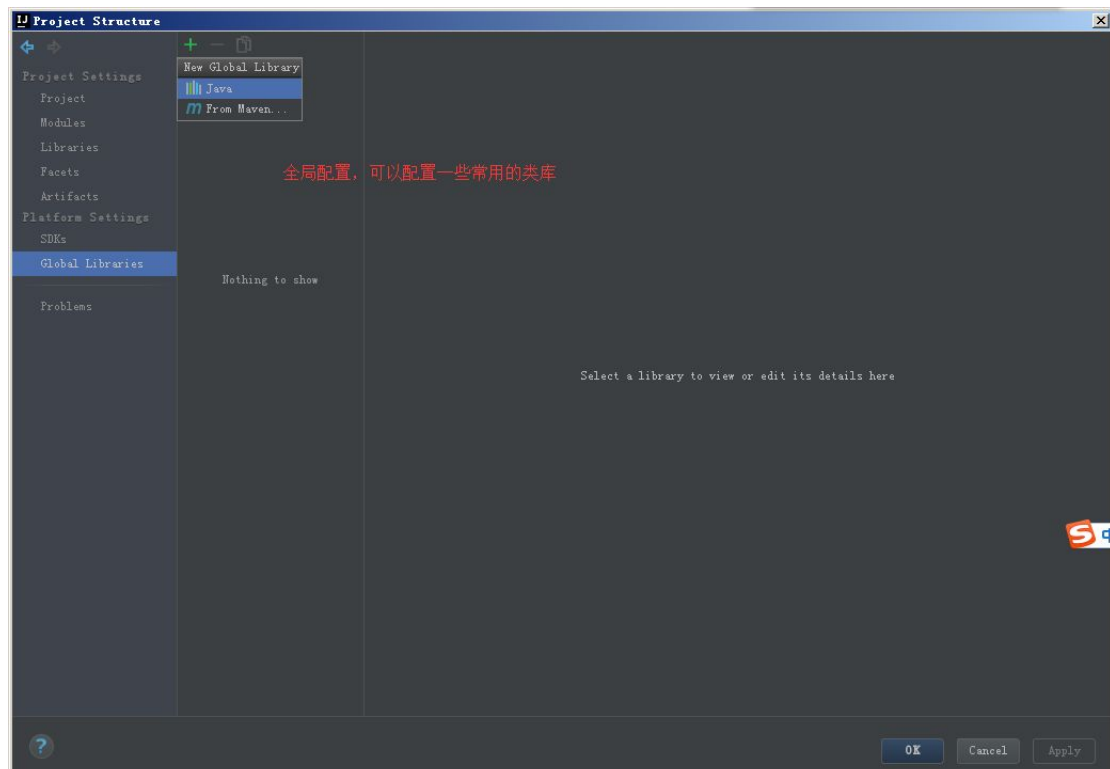




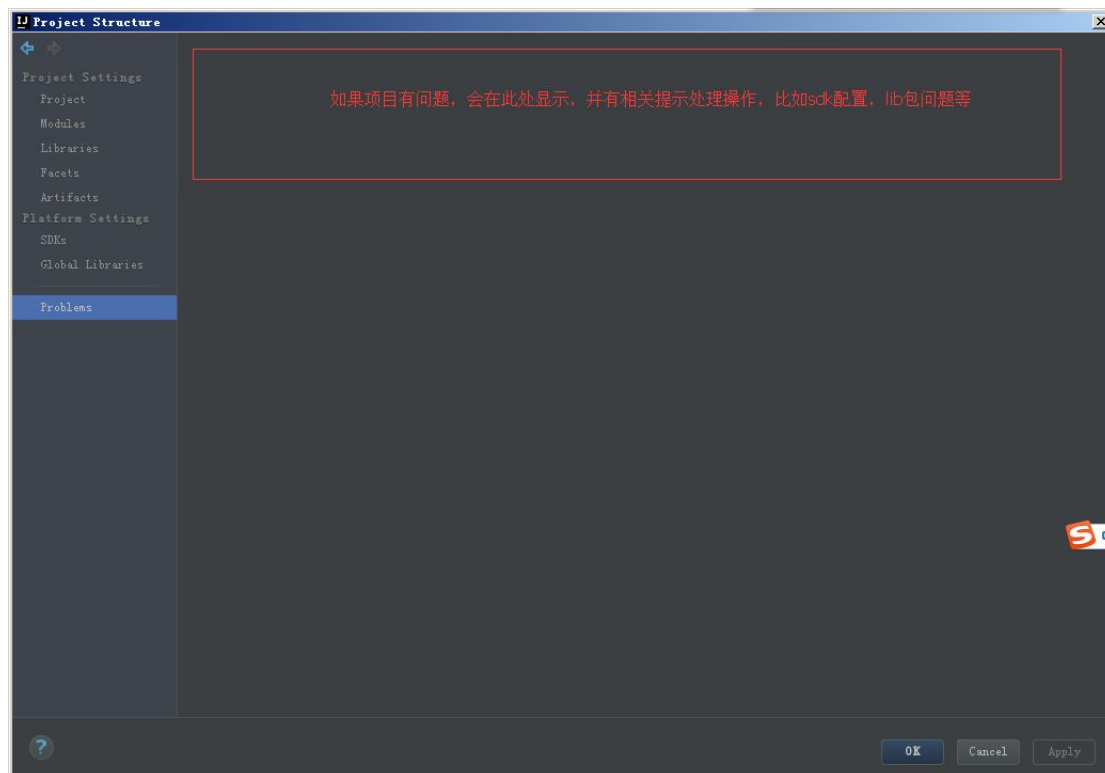
## SDK



## Global libraries



## Problems

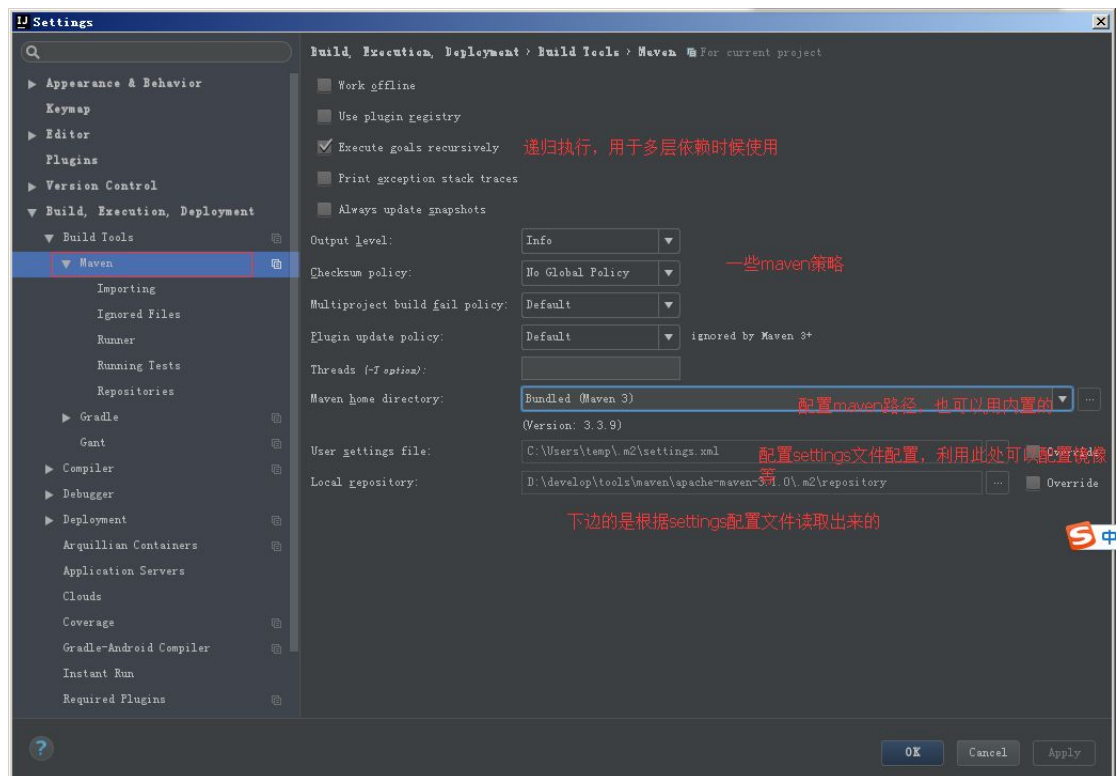


## Maven 专题

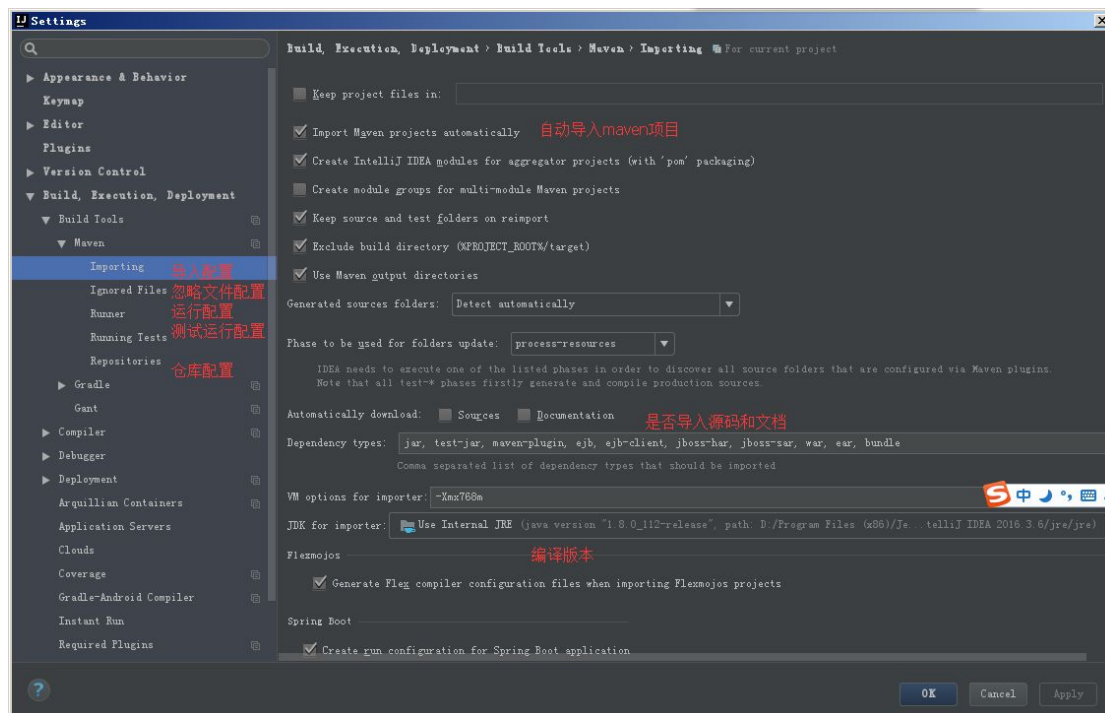
右键项目中的 pom 文件, 选择 add as maven Project, 即可加入 maven 管理

## 配置

## 主配置

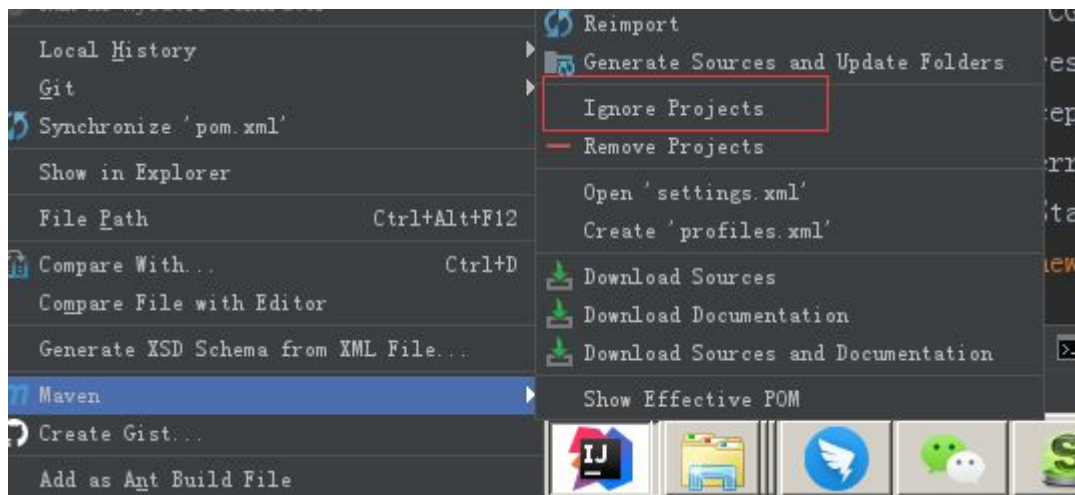


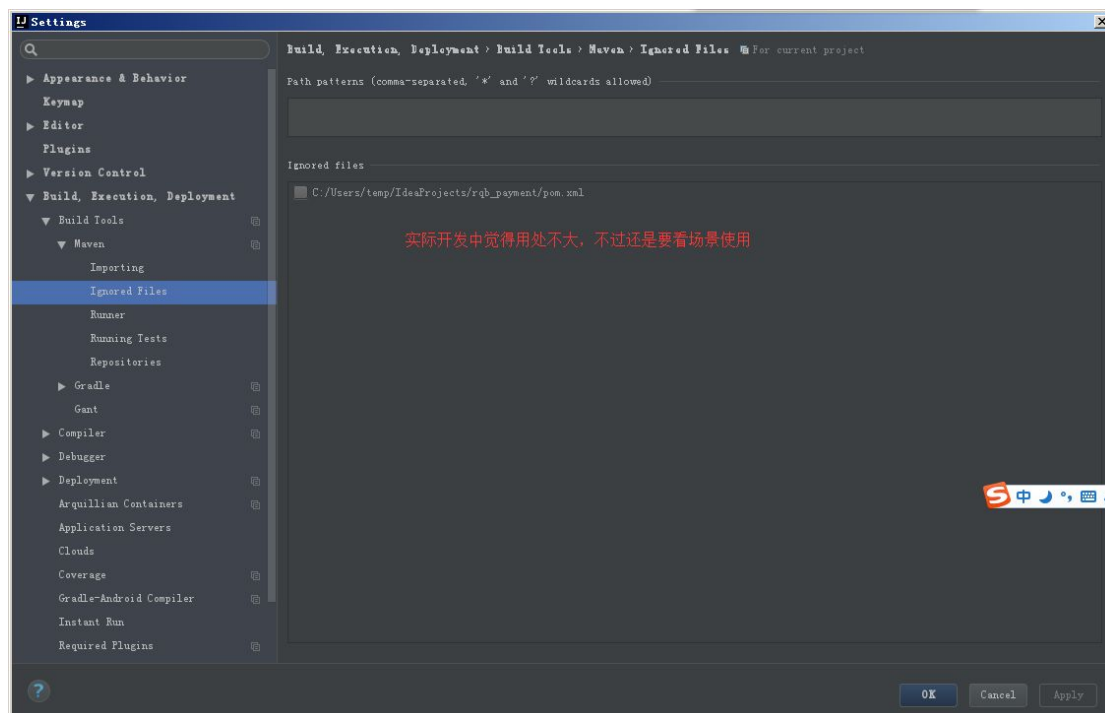
## Import 配置



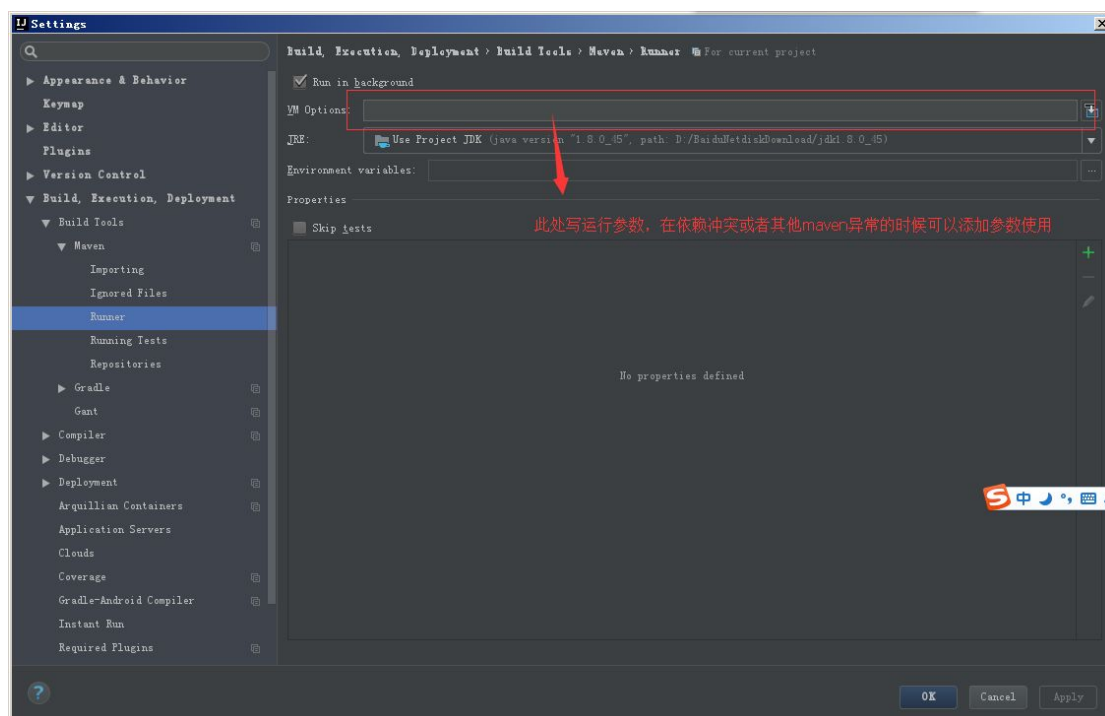
## Ignore Files 配置

通常与下面配合使用

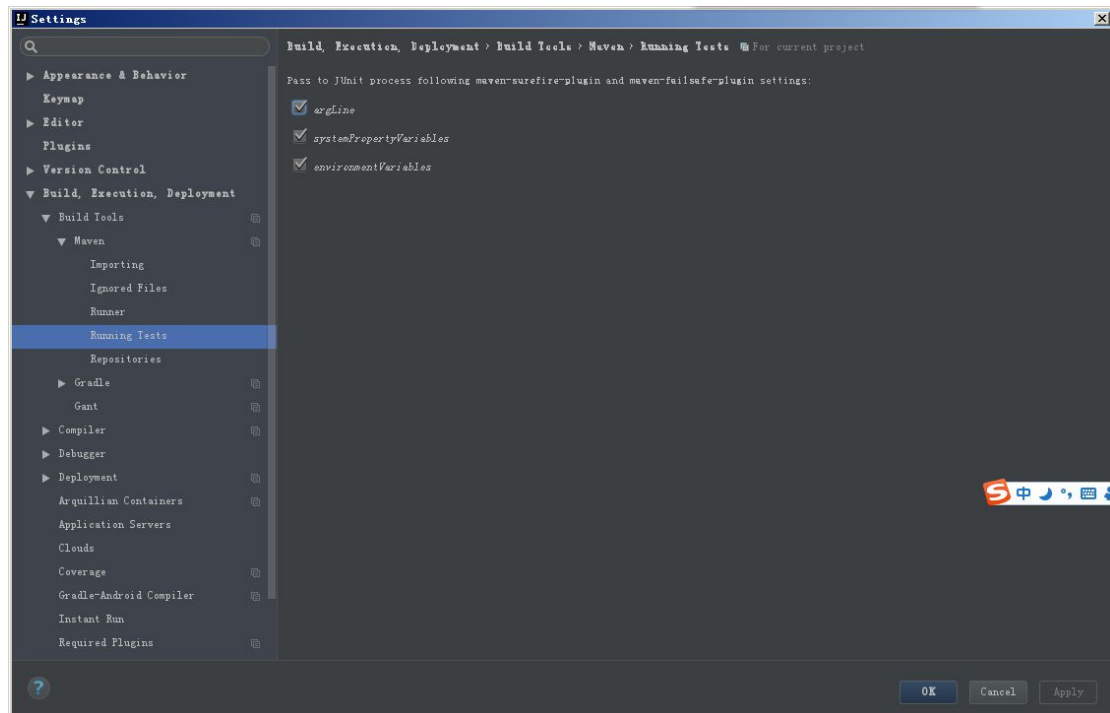




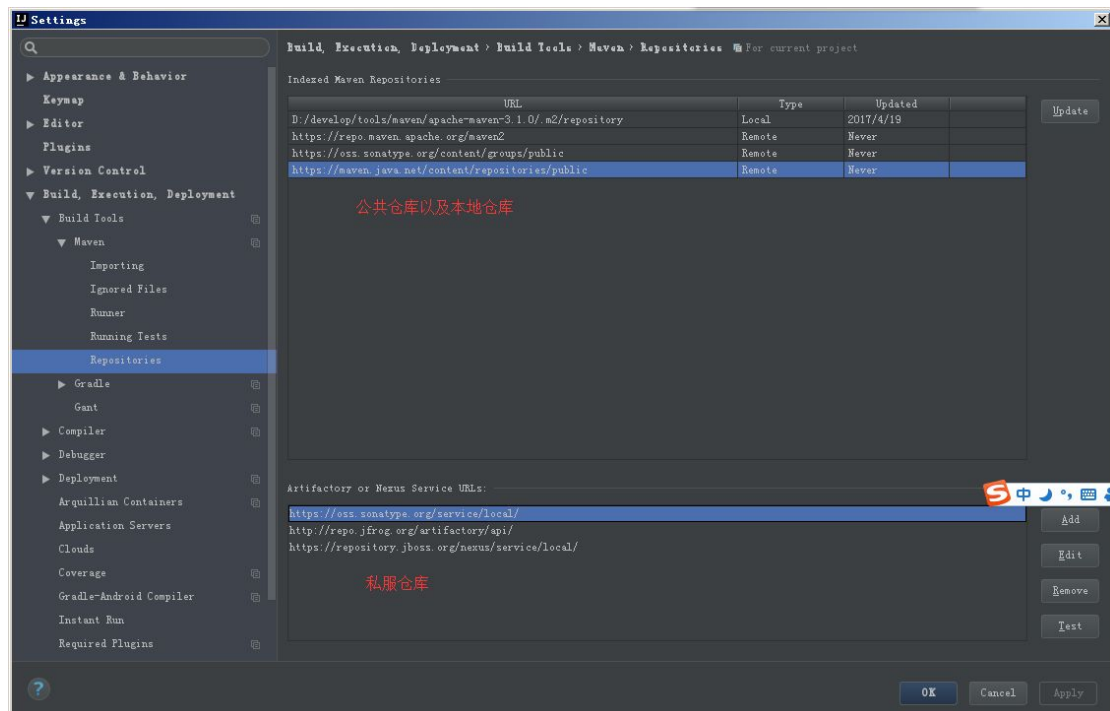
## Runner 配置



## RunnerTest

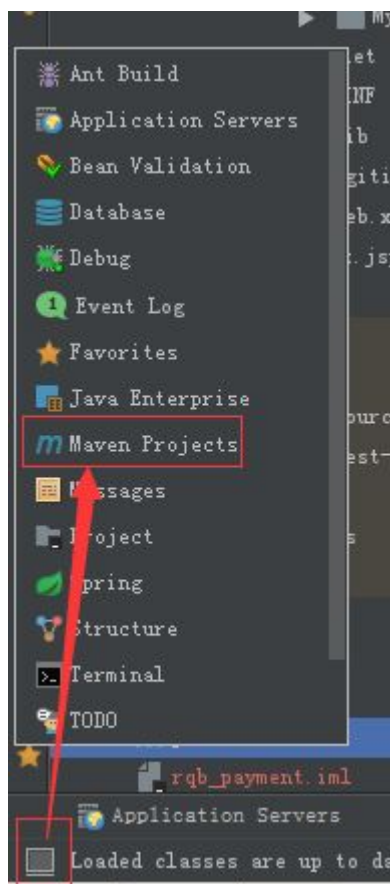


## Repositories 配置

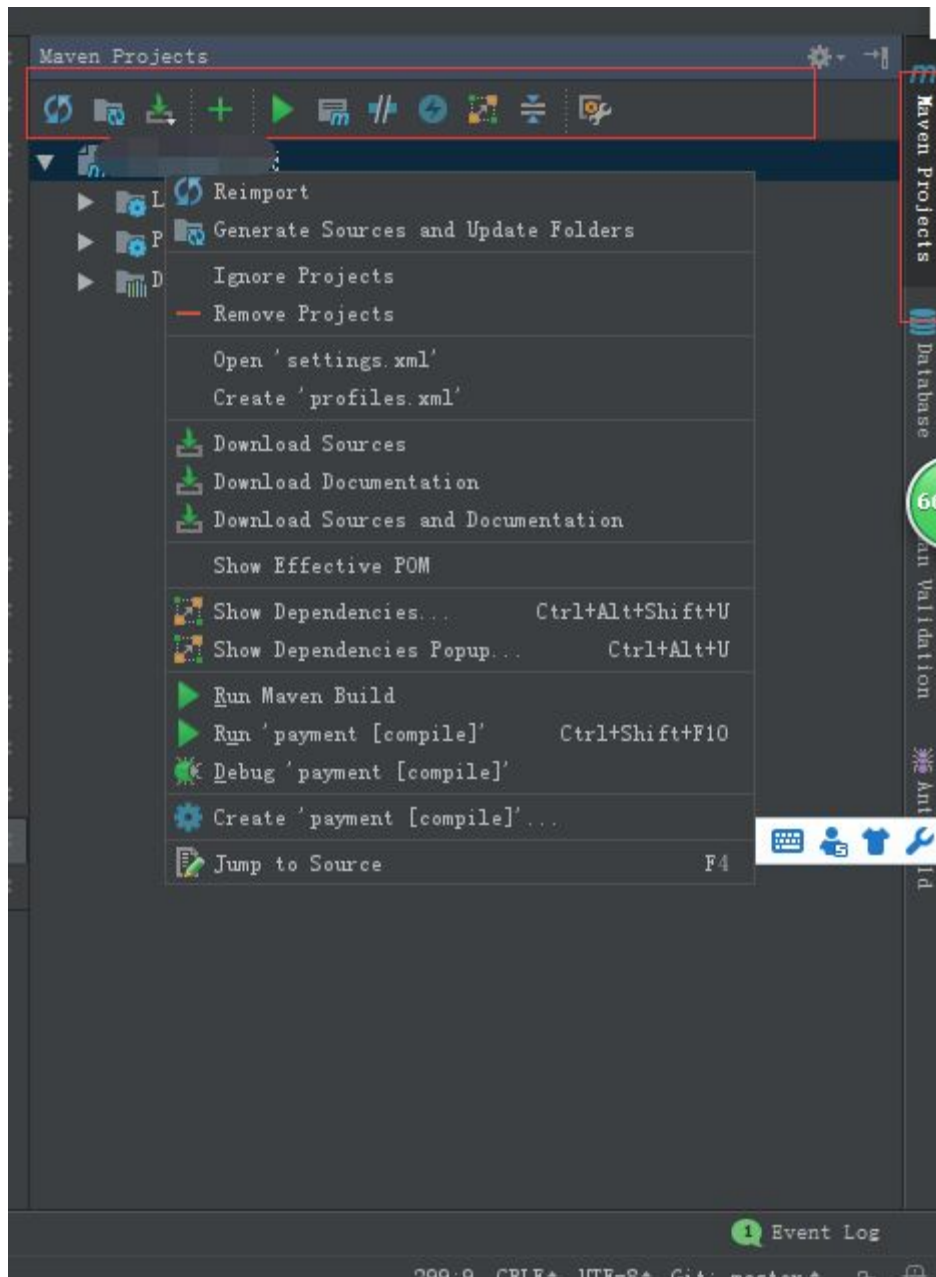


## 使用

通过如下方式调出面板



## 面板说明



上边一栏，依次为：



重新导入 **maven** 项目



对项目进行更新



下载文档和源码



添加 **maven** 项目





运行 maven 构建



全局运行



切换在线模式



切换测试模式



显示依赖拓扑结构图



折叠

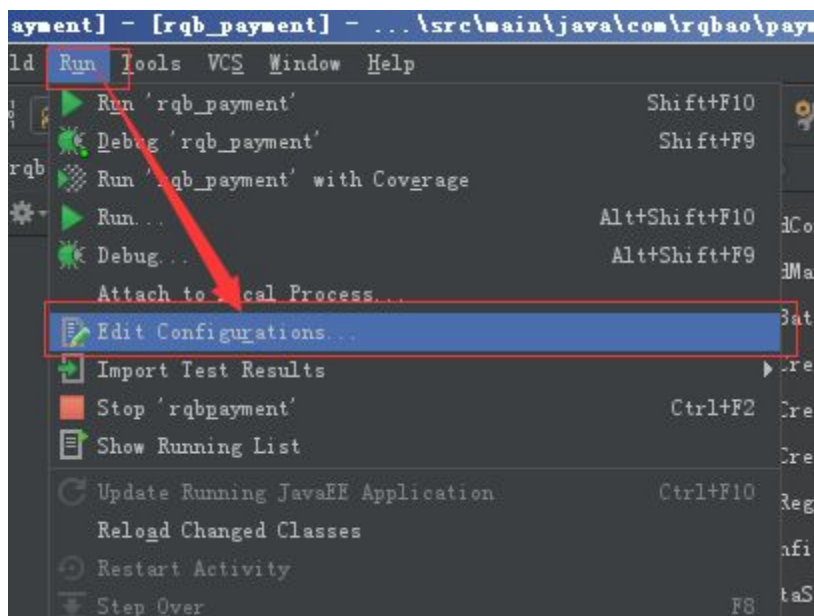


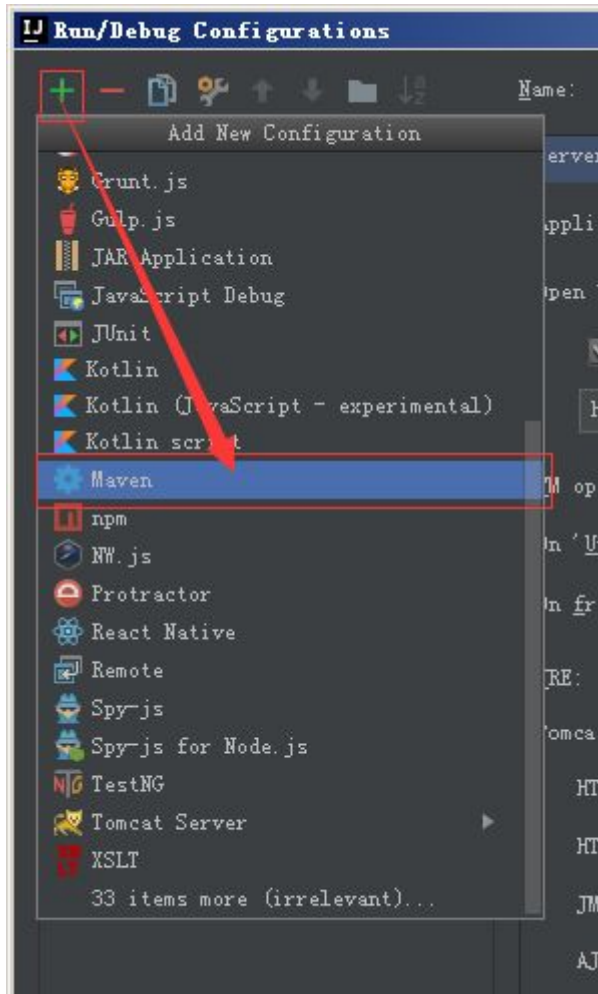
设置

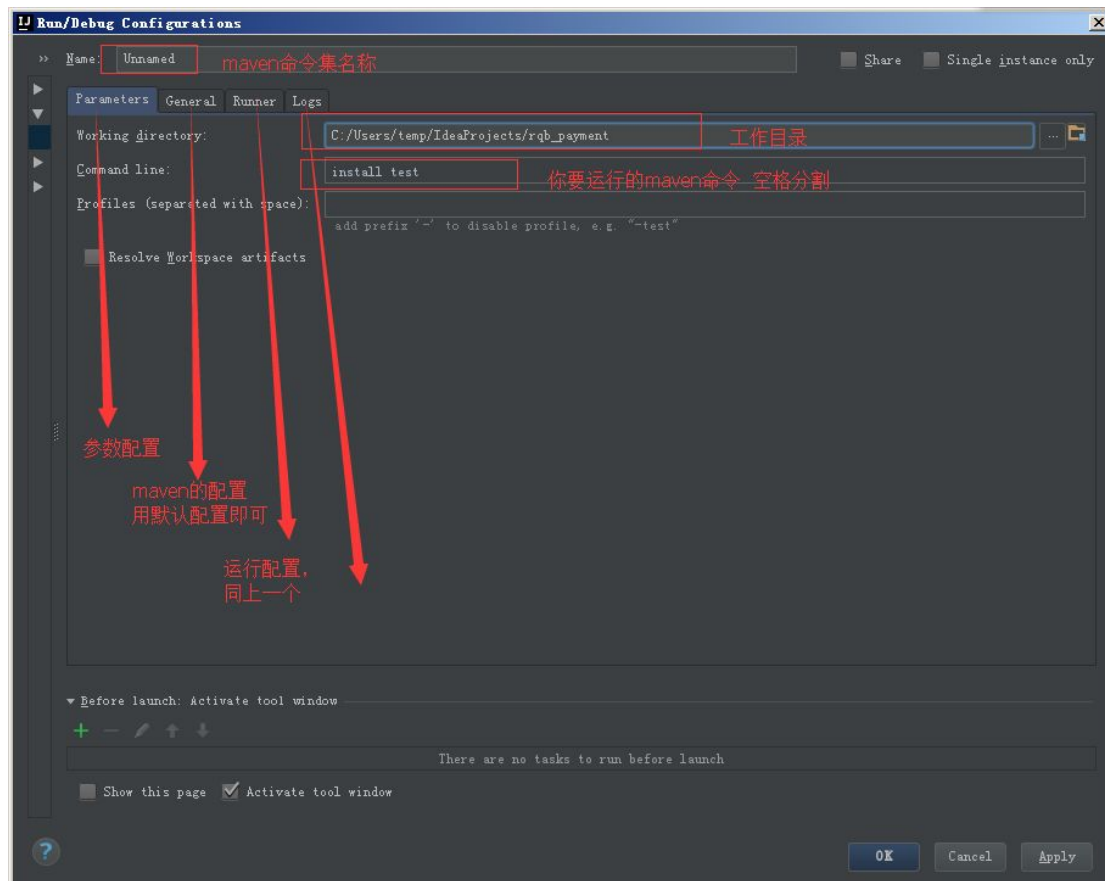
右键菜单的操作，基本类似

## 命令模式

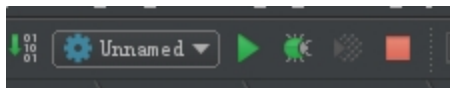
## 配置



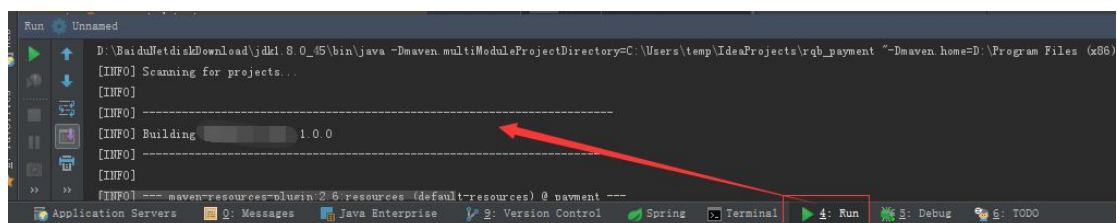




点击 ok 之后

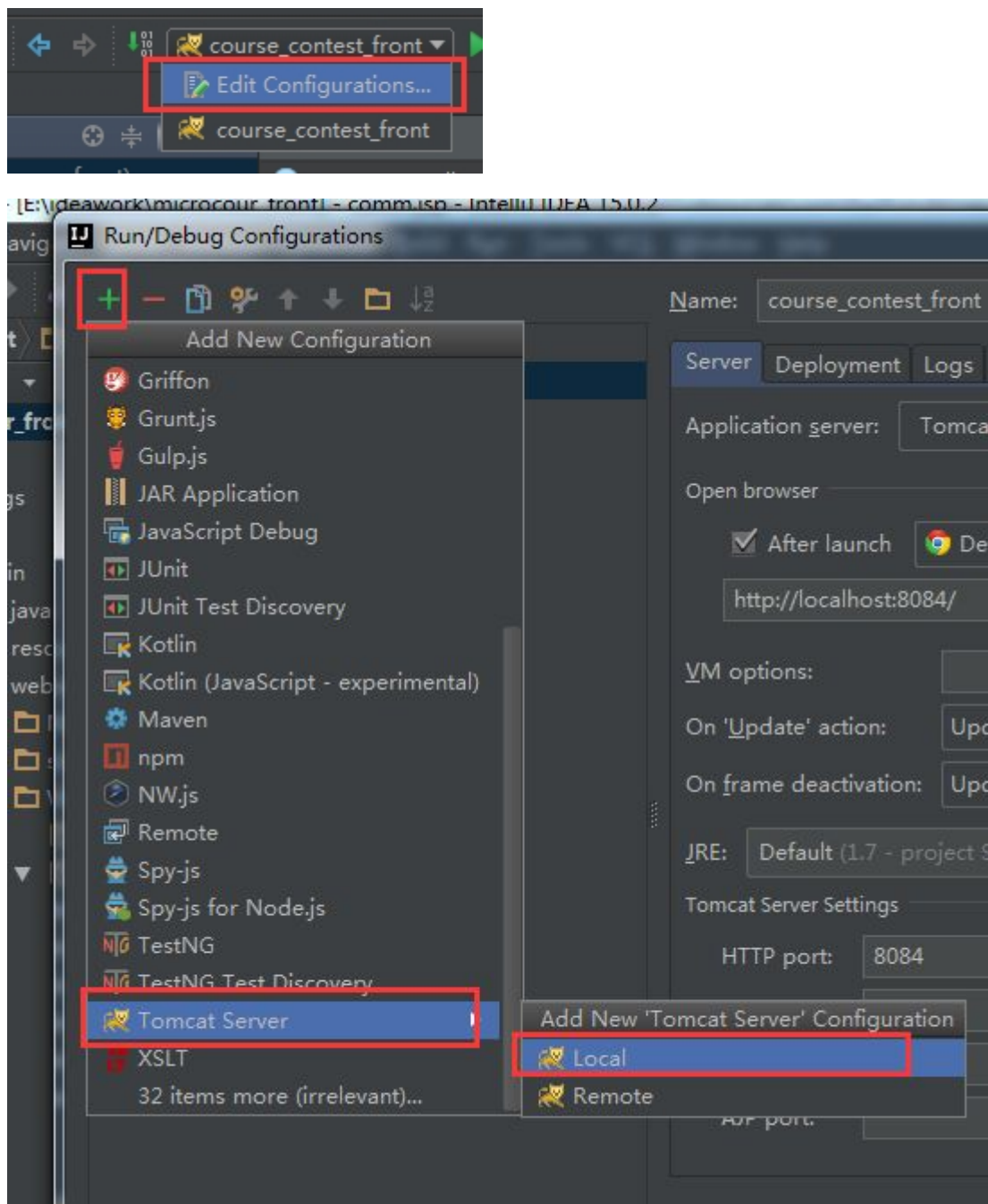


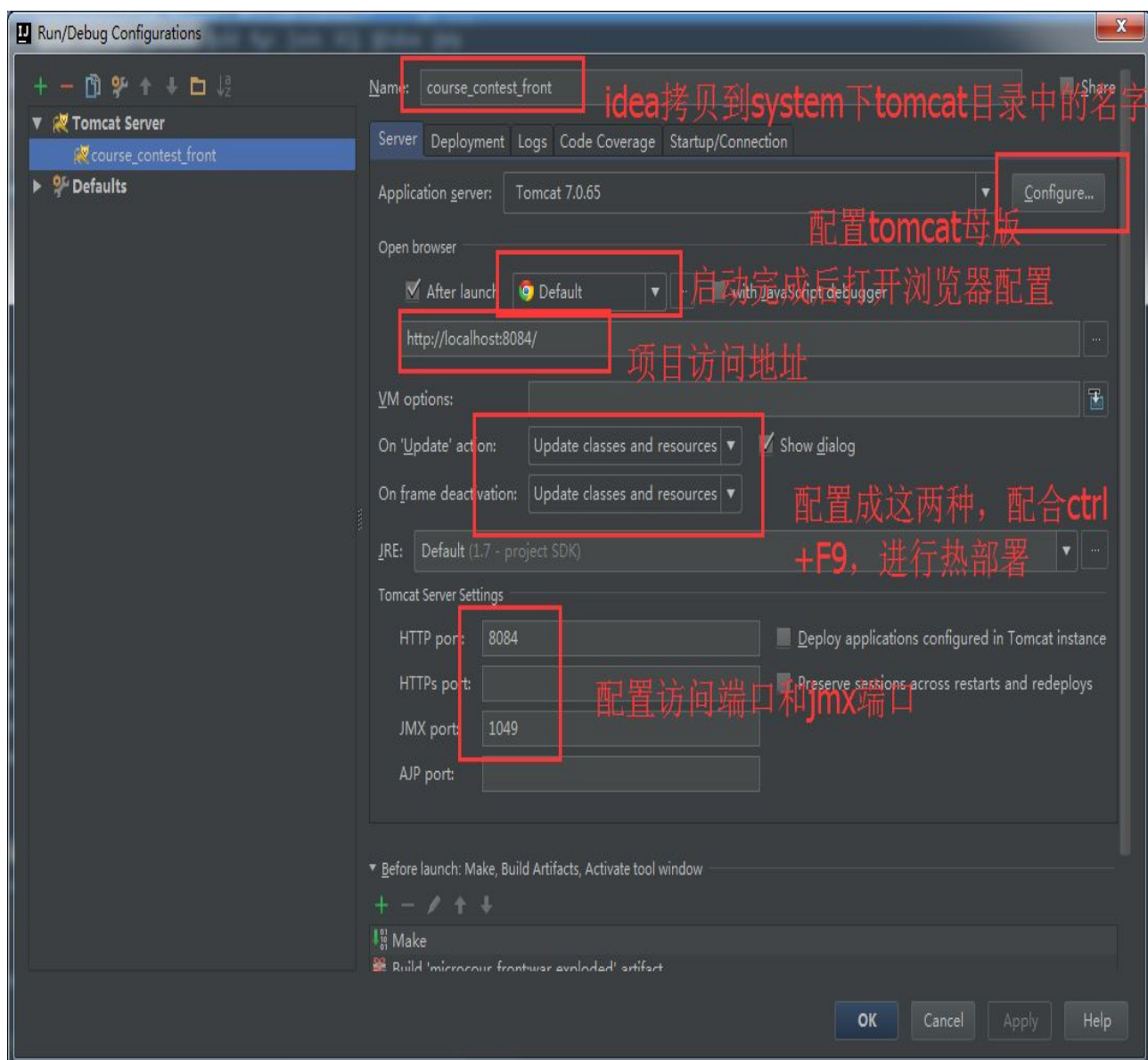
Run



# Tomcat 专题

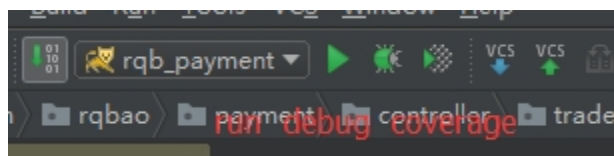
## 安装配置



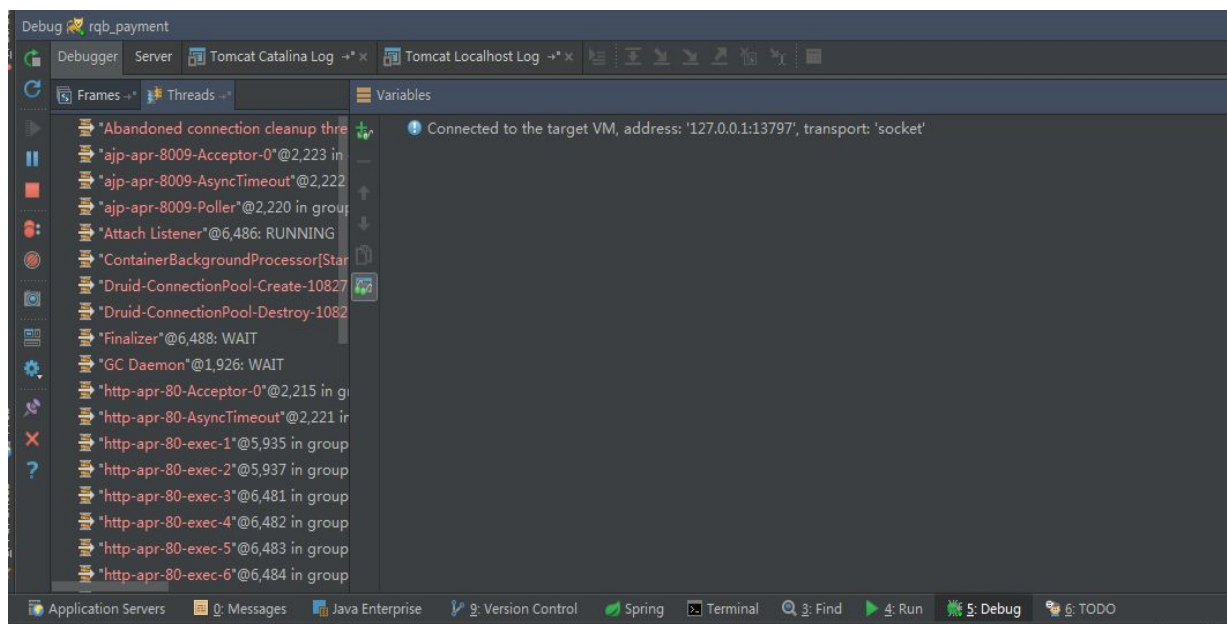


## 启动

三种方式, run, debug, coverage



## 面板说明



### 上面一排

**Debugger:** debug 模式的时候显示方法，调用关系，参数值等，

**Server:** 打印日志控制台

**tomcatCatalinaLog:** catalina 日志

**Localhostlog :** 本地日志

并排的一堆按钮是调试的时候进入跳出方法按钮：F8 下一步, F7 进入方法，F9 下一个断点

### 左侧竖列

依次为：

**Run:** 启动应用

**Update application:** 更新应用

**Resume application:** 恢复应用

**Pause application:** 暂停应用

Stop: 停止应用

View breakPoints: 查看断点

Mute breakpoints: 禁用断点

Get Thread Dump 获取线程堆

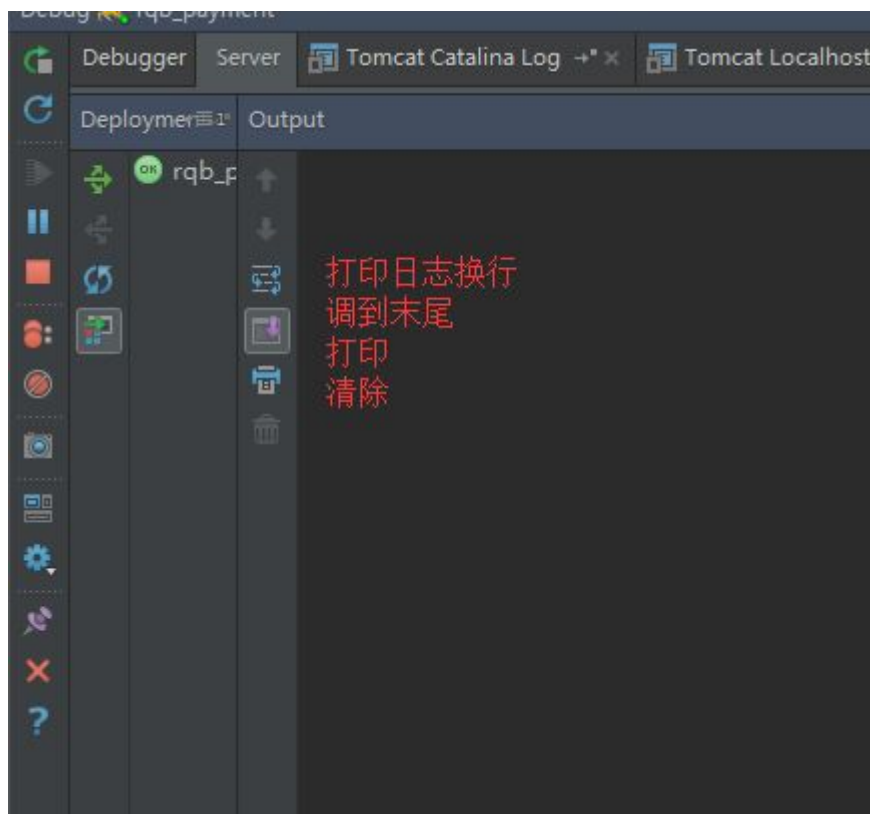
Restore layout: 重置布局

Settings: 设置

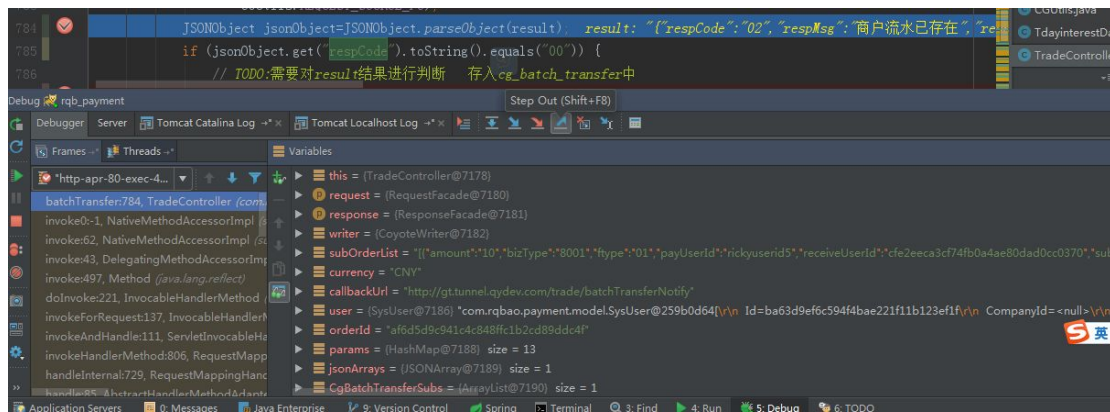
Pin tab: 固定面板

Close: 关闭

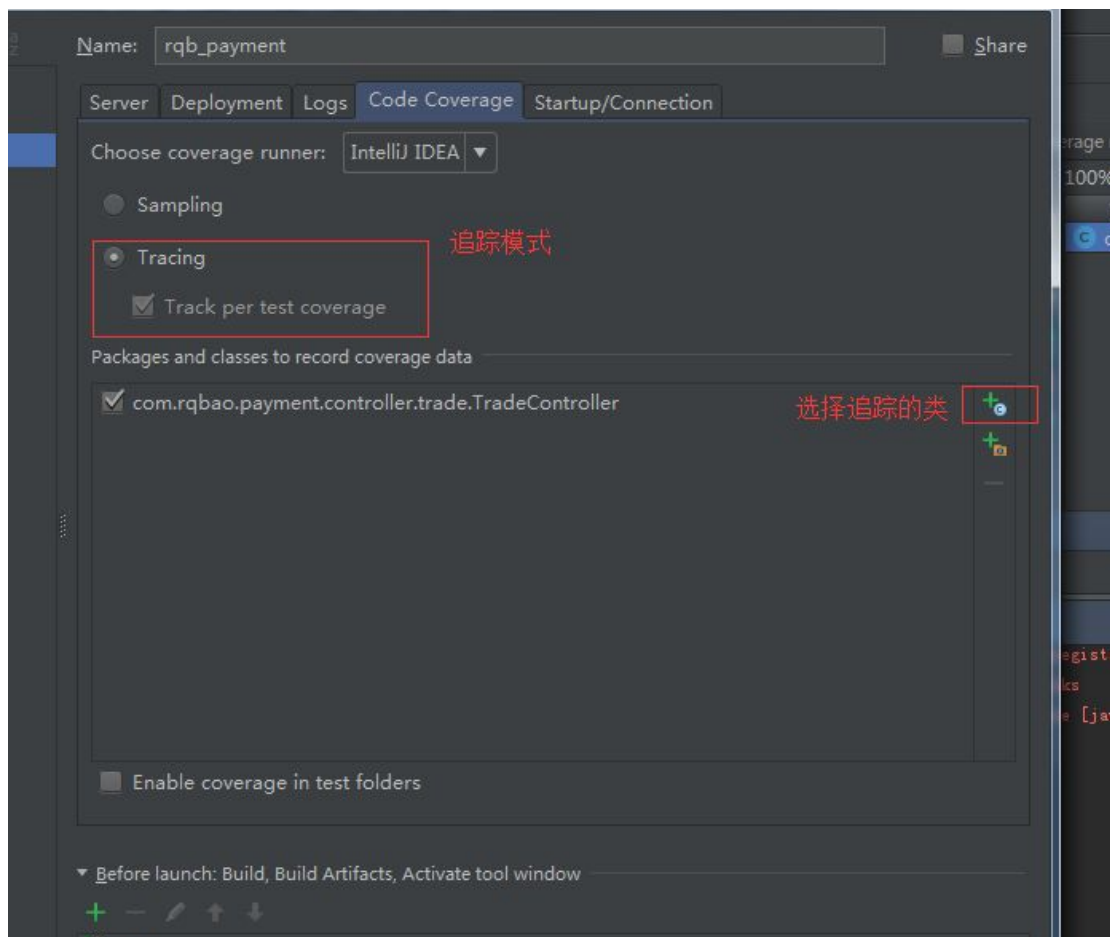
Help: 帮助





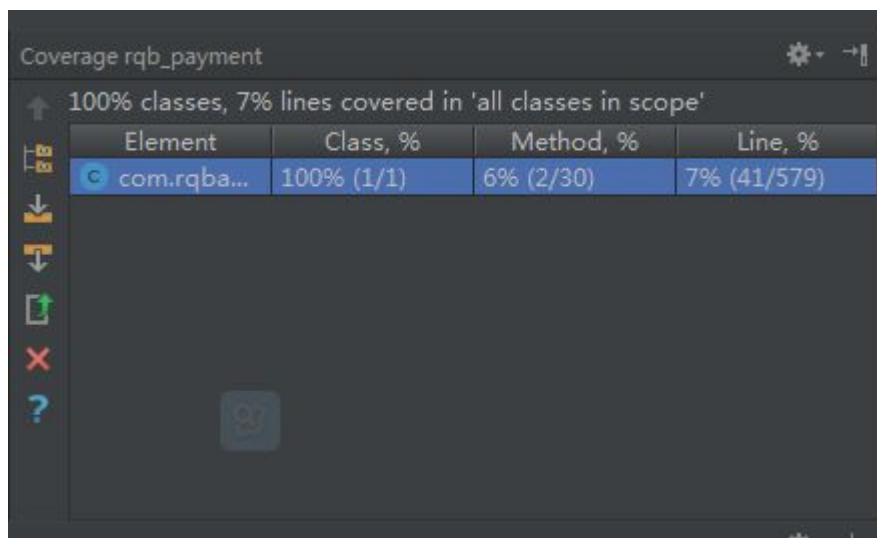


## Run with coverage



终止程序后会显示调用比例

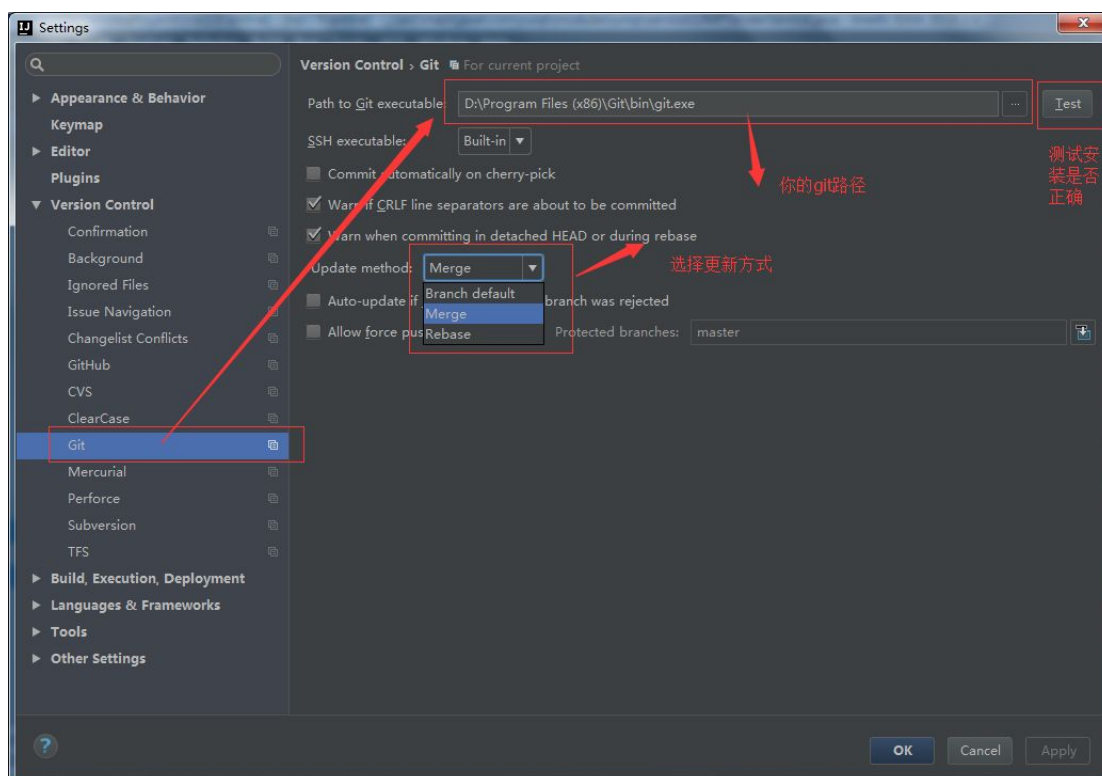




## GIT 专题

### 安装

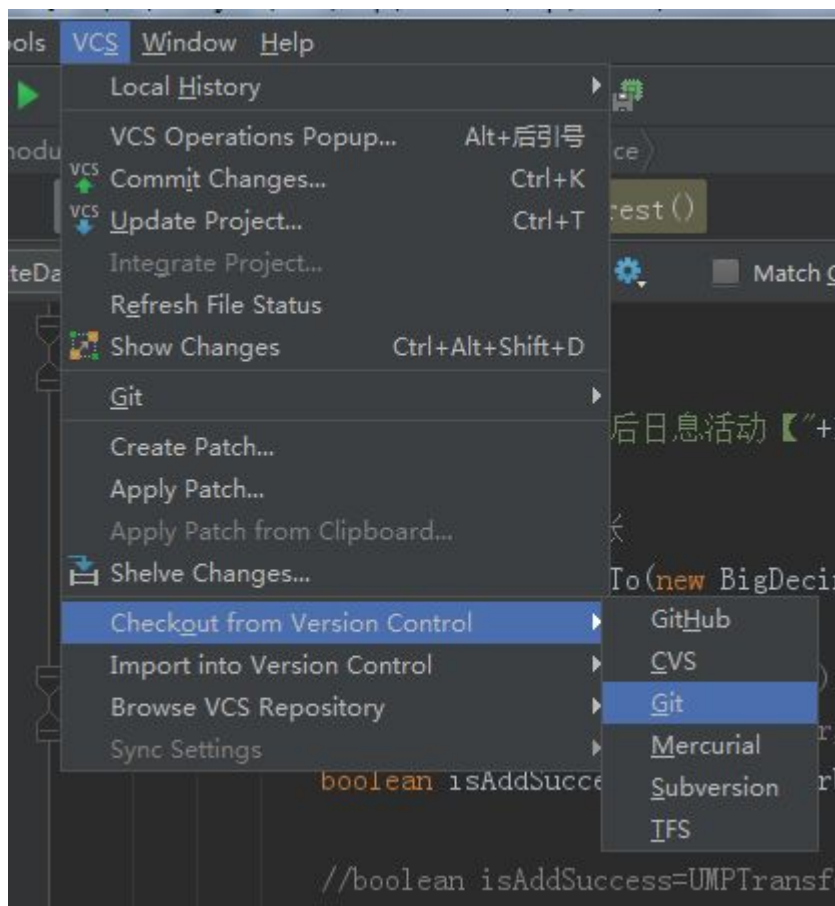
选择 VersionControl 下的 Git，依据下图进行 git 配置



## 使用

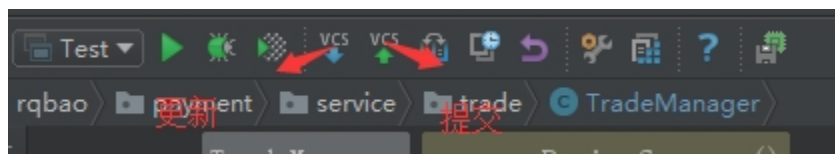
## 荡项目

从 VCS 菜单选择 checkout from version control-->git

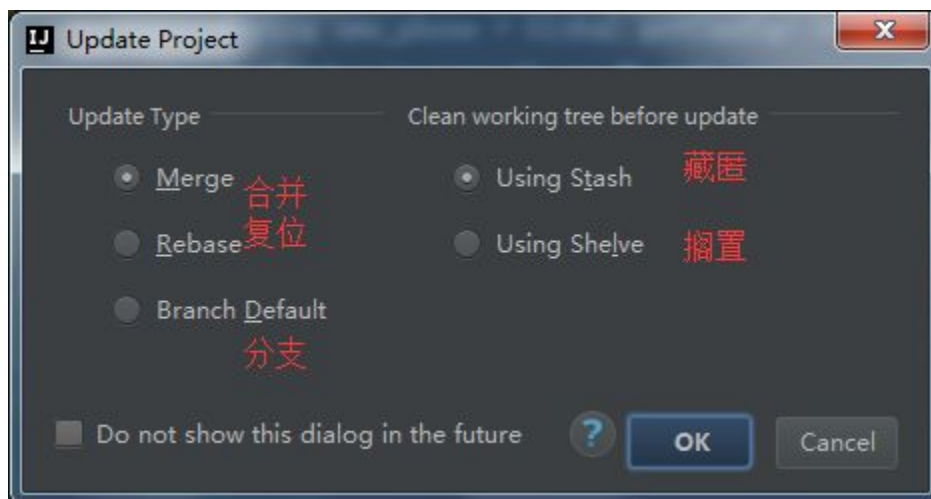


## 更新项目

如果本地有分支，一般是从本地，没有的话则是从远程



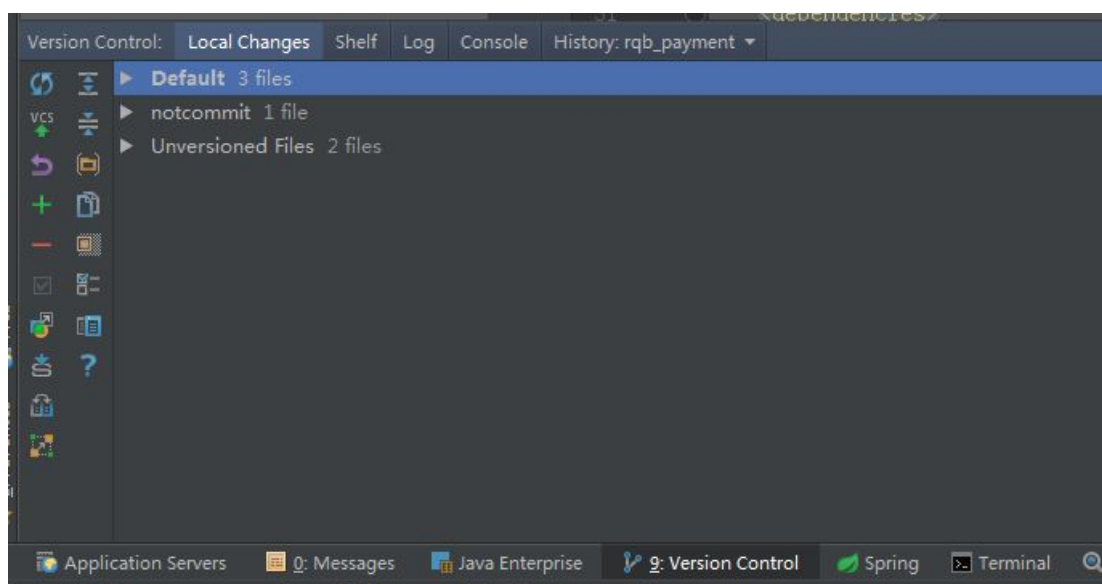
Ctrl+k 后



## 面板说明

Alt+9 跳转到 versionControl 面板

## Local Changes

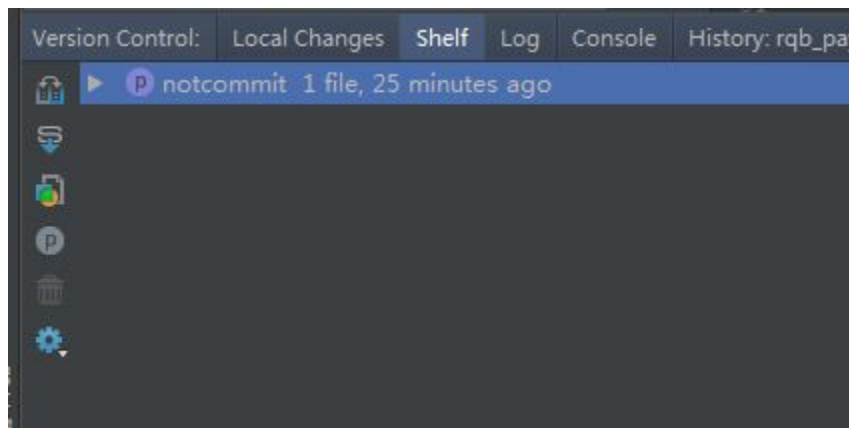


左侧菜单自上至下依次为:

刷新	展开
提交	折叠
还原	复制
新建版本列表	显示忽略文件
删除版本列表	设置忽略文件规则
设置列表活动态	预览不同
移动列表	帮助
搁置列表更改内容	
展示不同	

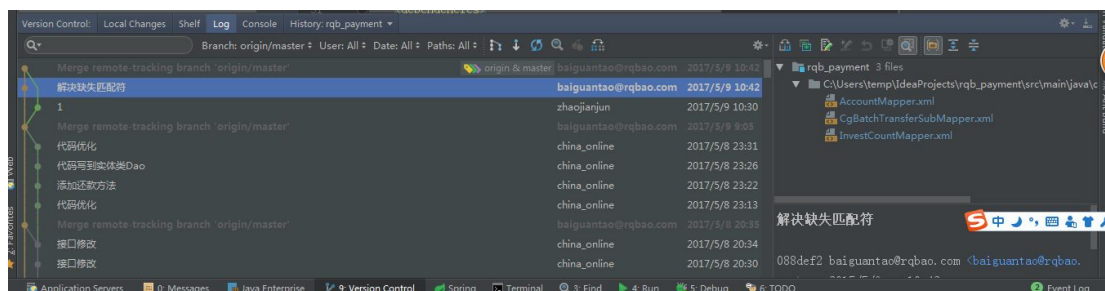
展示变动	
------	--

## Shelf 面板



搁置操作后的面板，可用于还原搁置的操作

## Log 面板

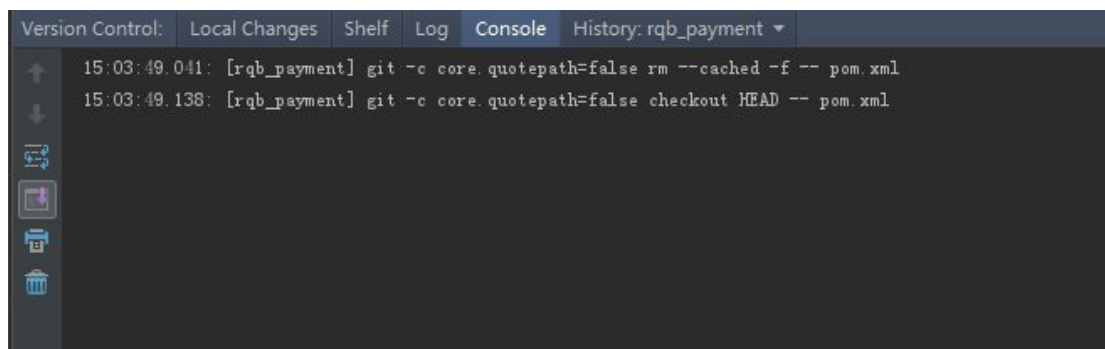


自上至下---》自左到右

搜索(提交消息)、分支筛选、用户筛选、时间筛选、路径、排序开关、显示长优势（边缘）、刷新、进入分支等、pick、高亮 pick
---

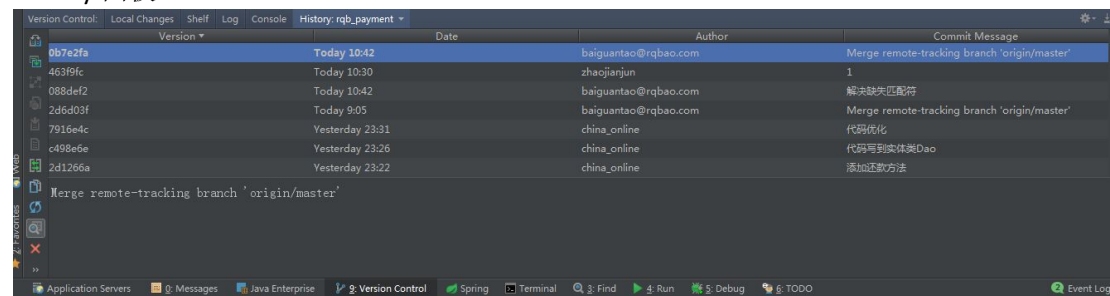
提交消息、用户信息、提交时间、提交明细
---------------------

## Console 面板



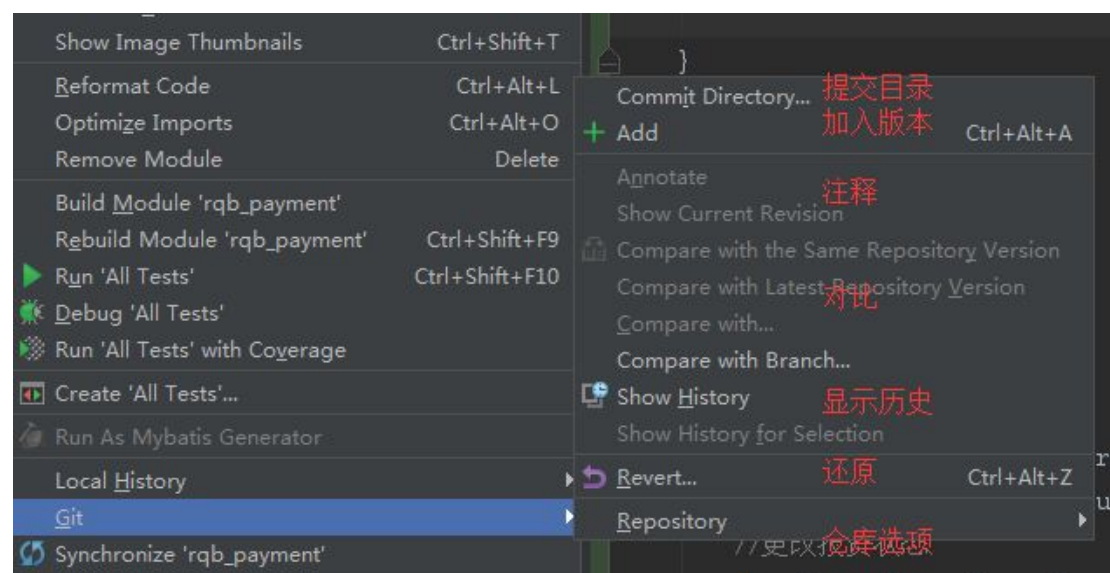
打印操作的 git 命令

History 面板

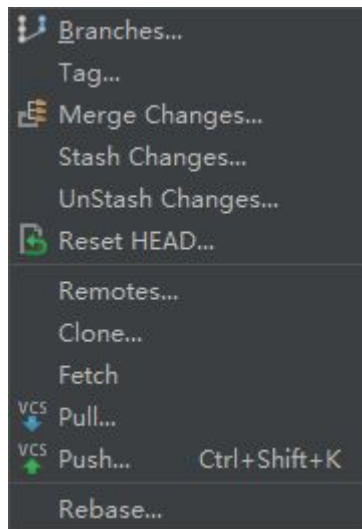


版本号 时间 作者 提交消息  
提交消息完整内容

项目 git 面板



## 仓库选项



分支、标签、合并、隐藏变动、不隐藏变动、重置 head、远程地址、克隆、获取、拉取、推送、复位

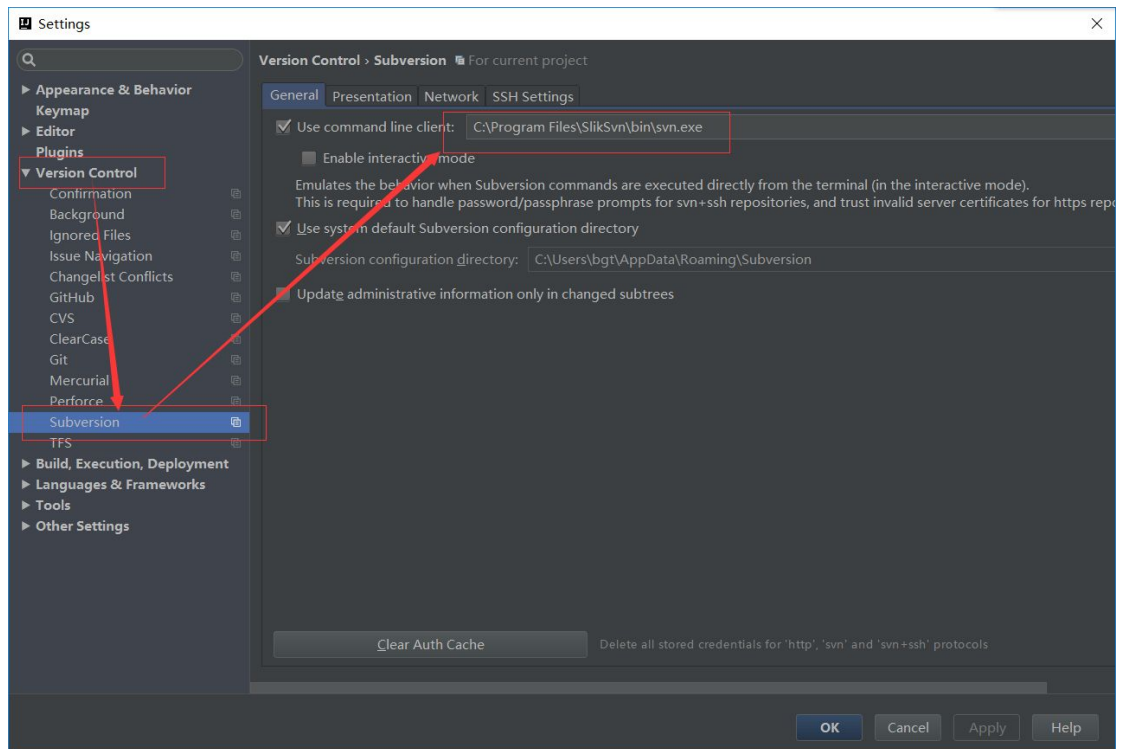
## SVN 专题

### 配置

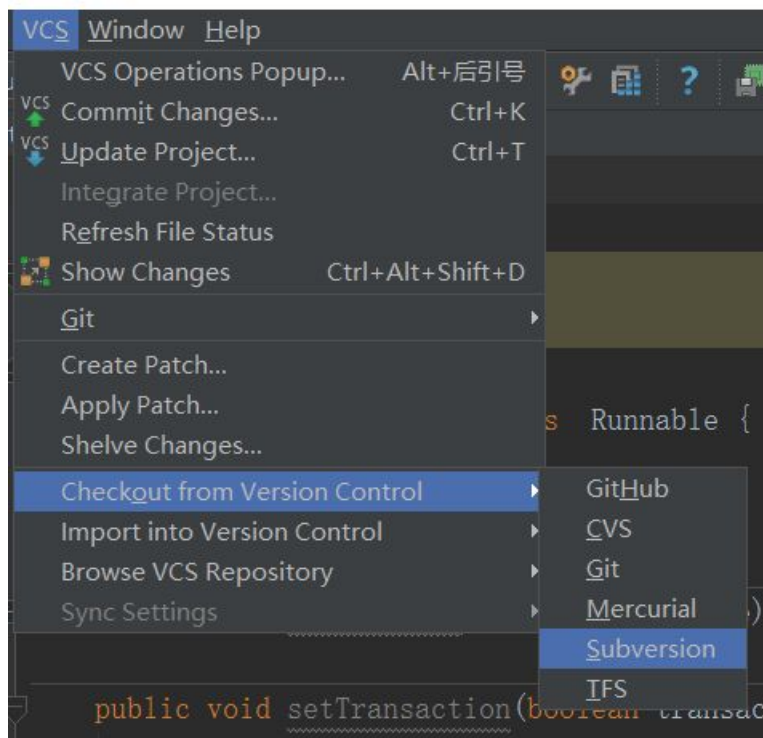
#### 1.1 下载&安装 svn

地址: <https://sliksvn.com/download/>

## 1.2 配置

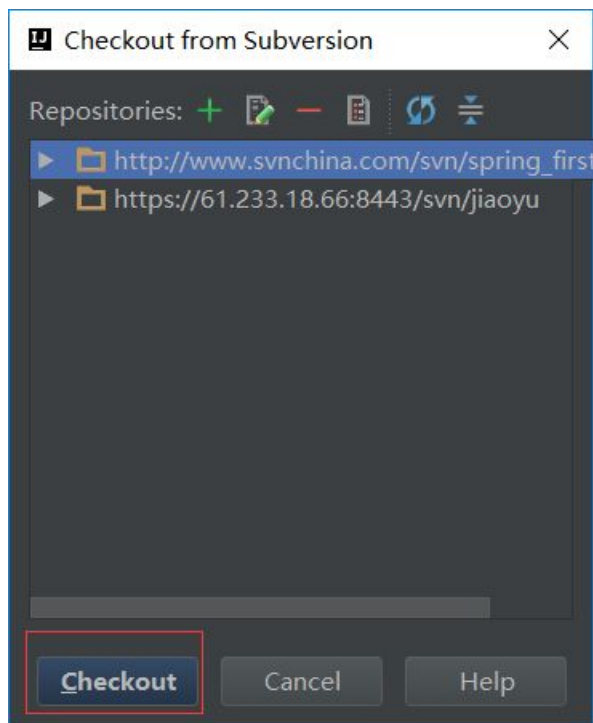
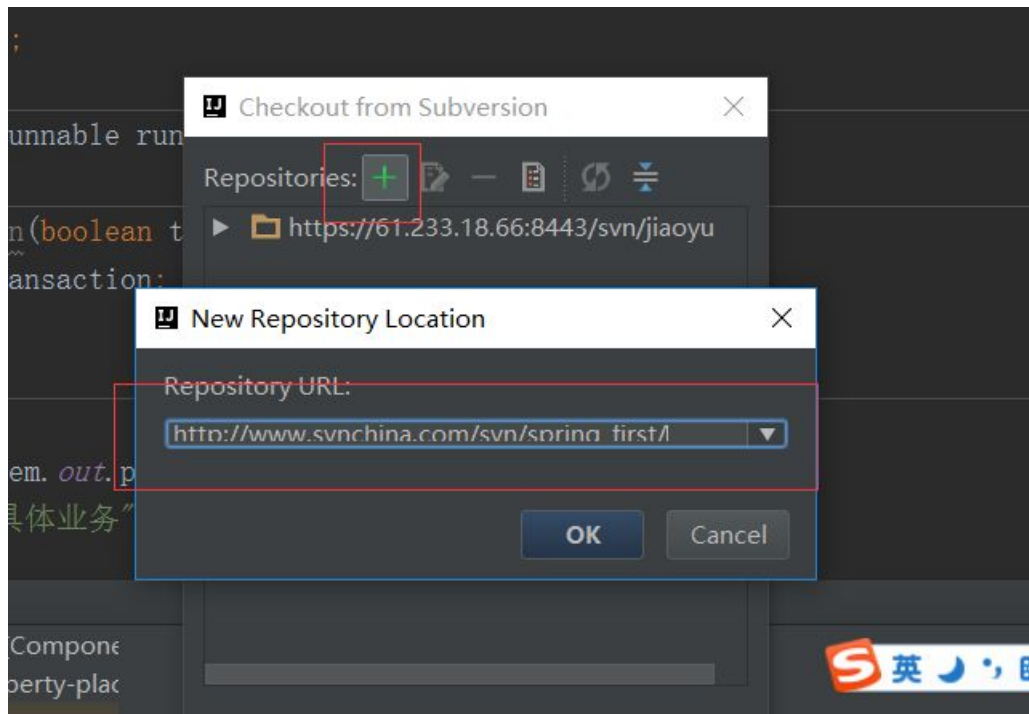


## 检出项目



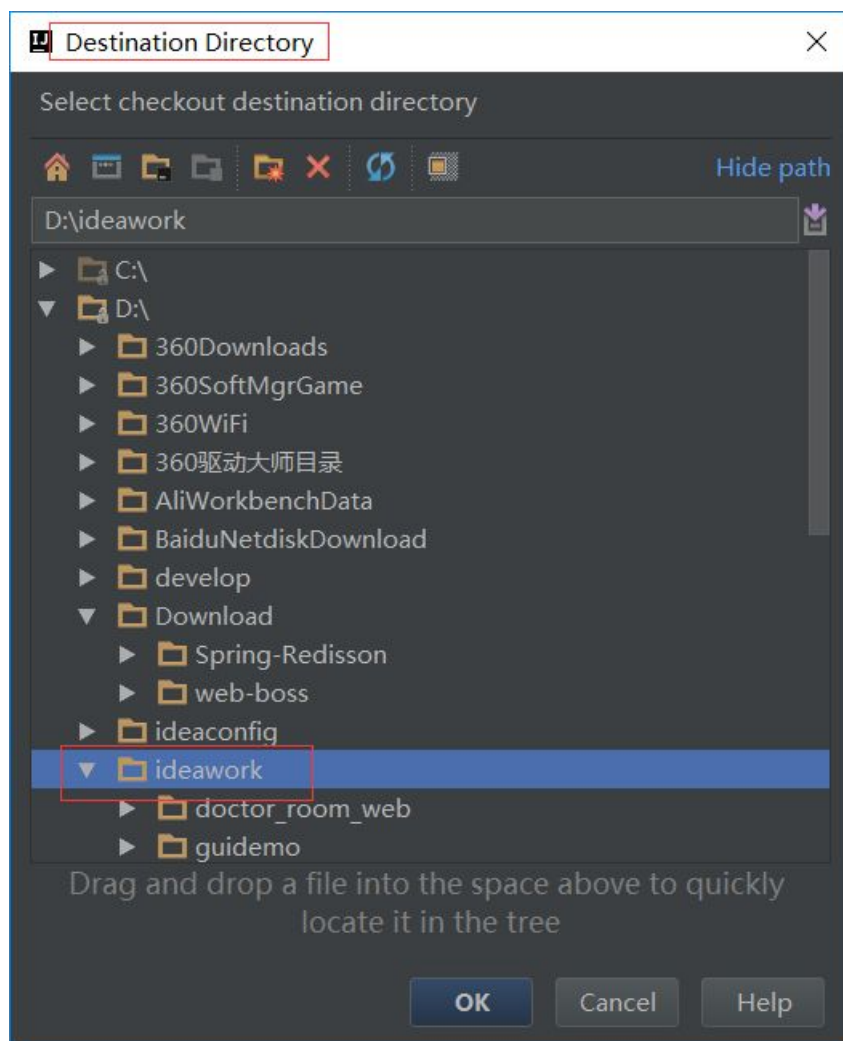
## 配置 svn 地址



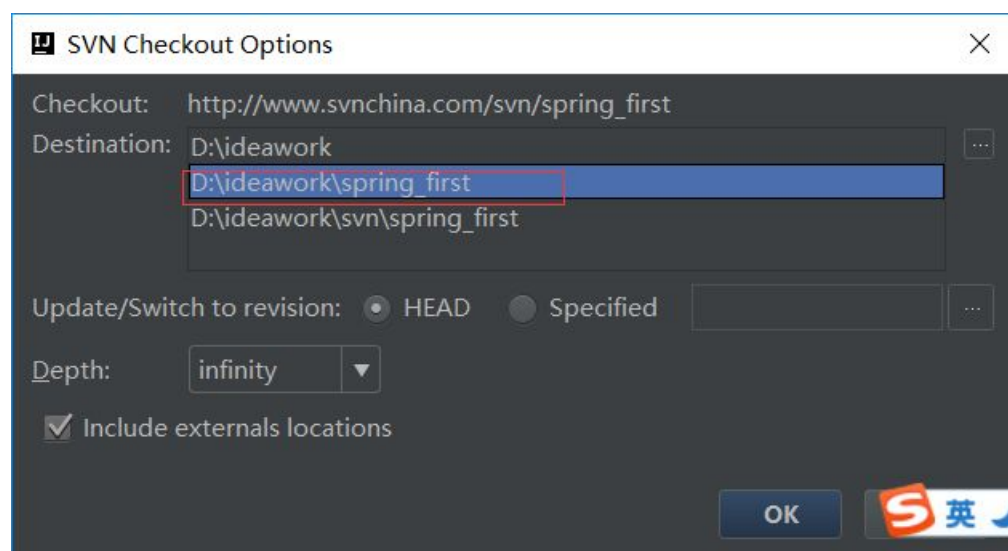


配置工作目录

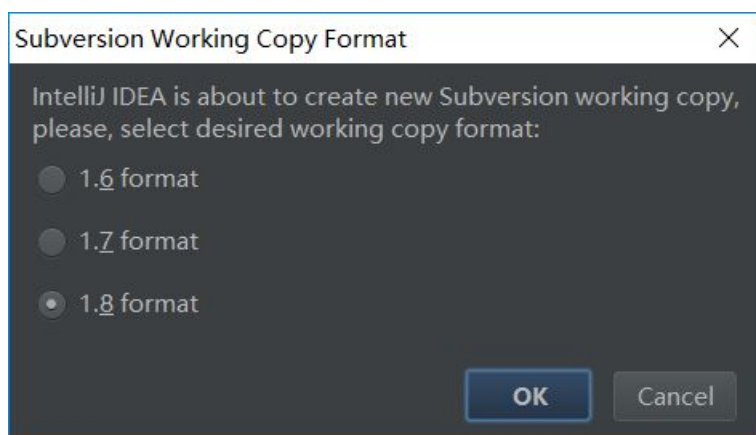




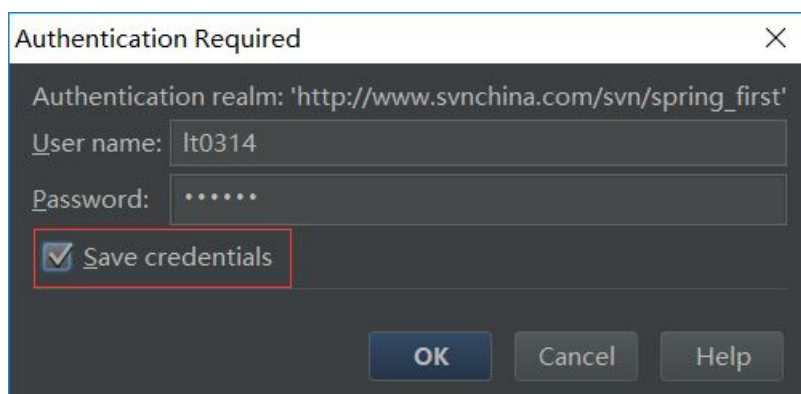
配置项目名称



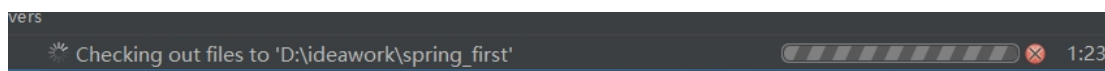
选择 format 版本



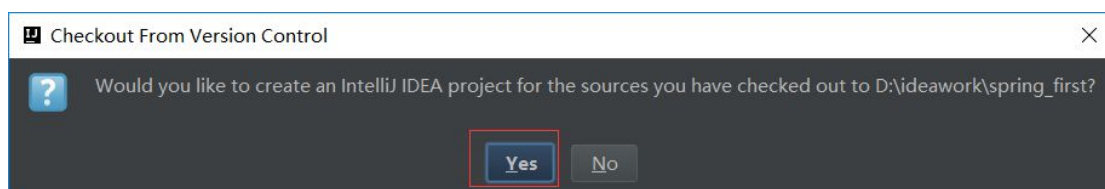
开始检出，验证账号密码



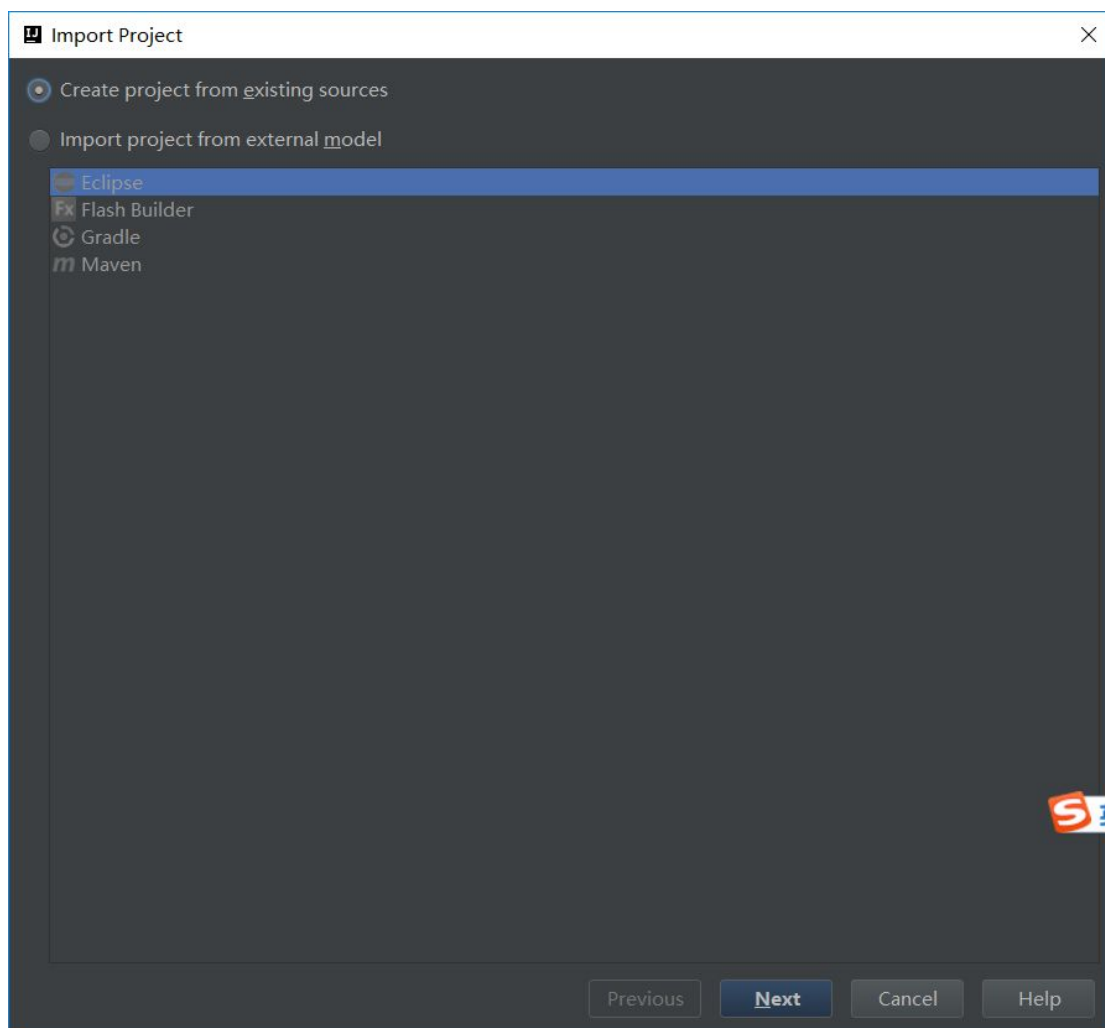
检出:



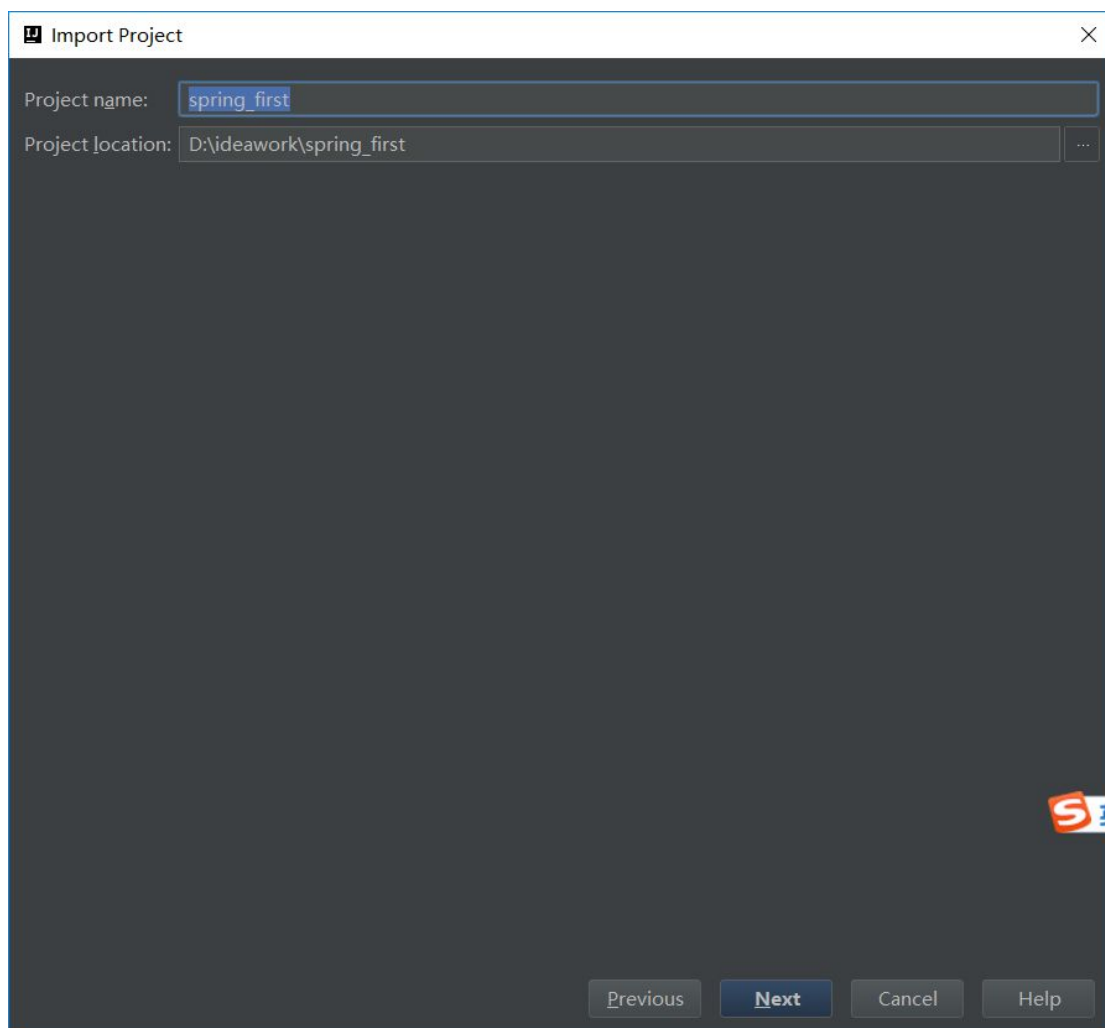
检出完成



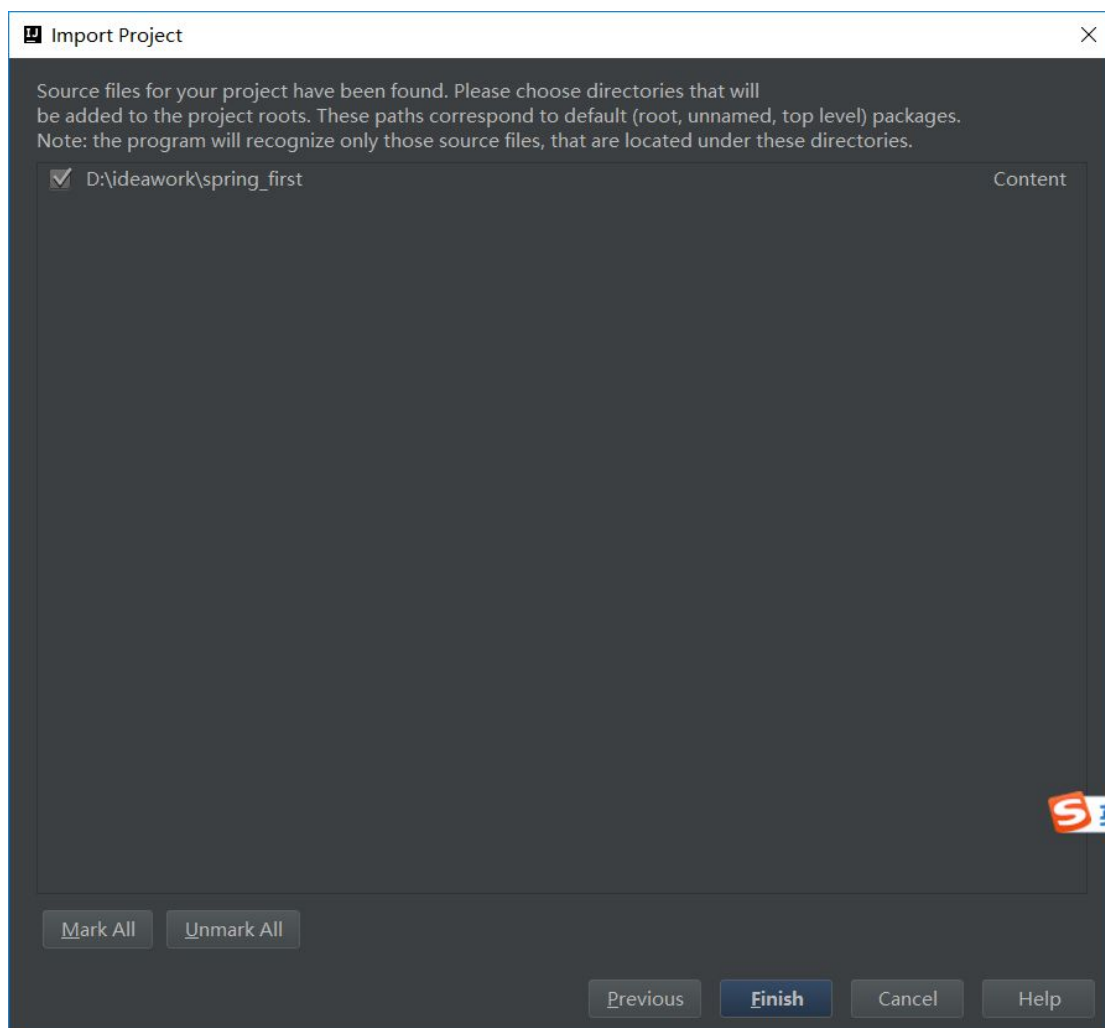
导入配置



Next



Next



Finish

完成项目检出全部工作。

## 面板说明

## 头部面板

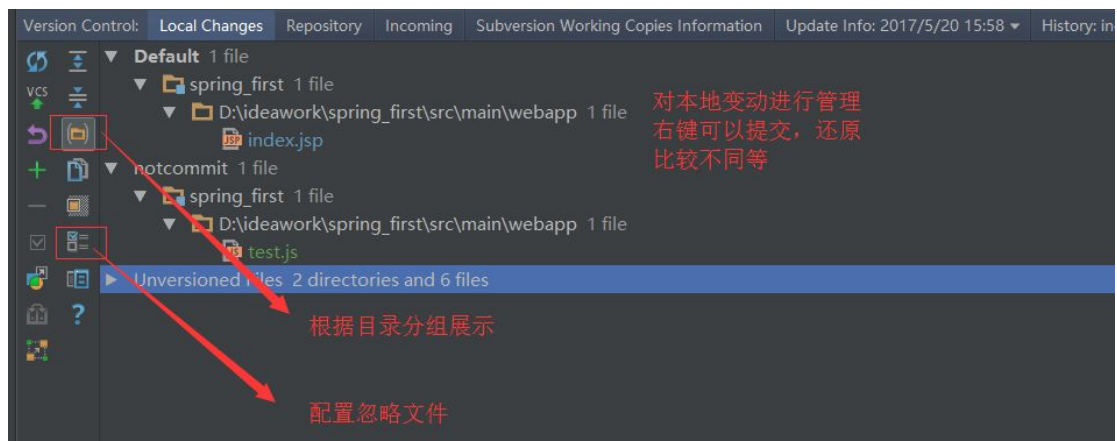


更新、提交、对比、显示历史、还原

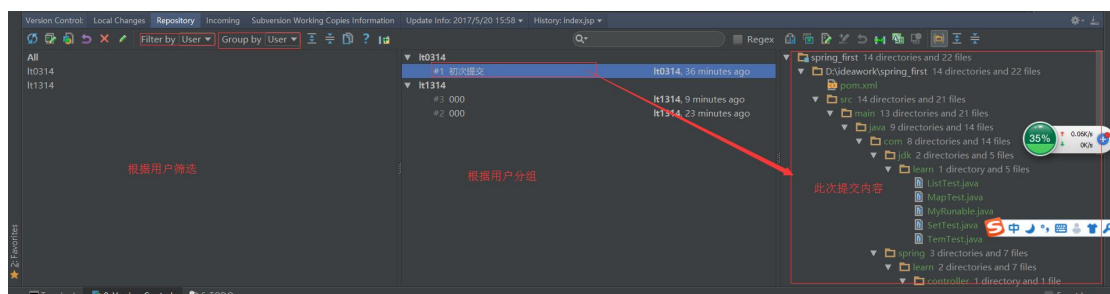
## VersionControl



## LocalChanges 本地更改



## Repository (仓库)



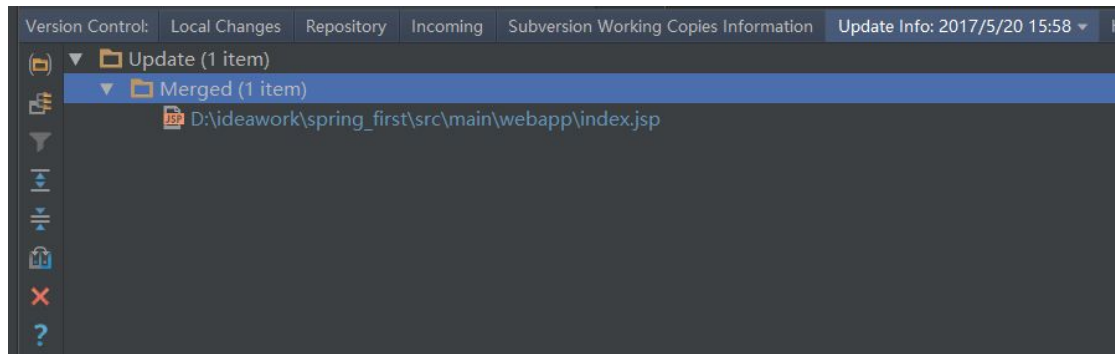
## Incoming(即将到来)



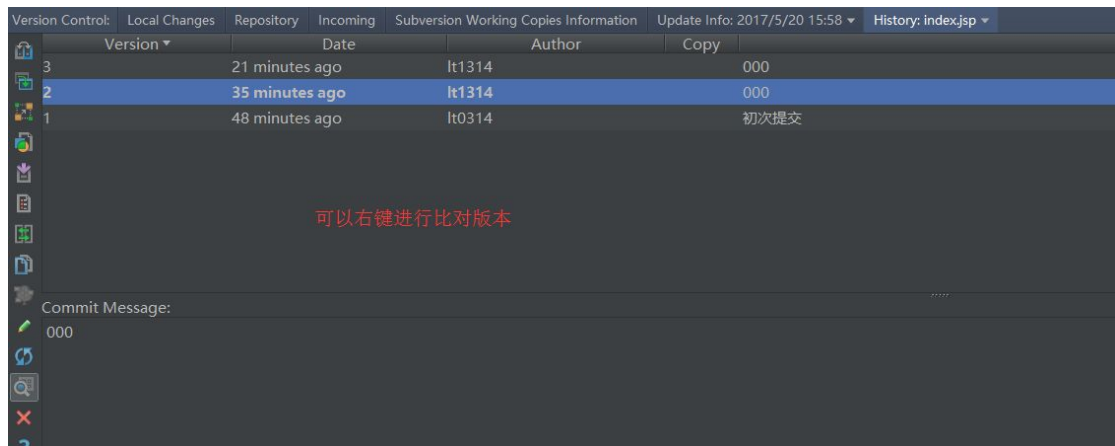
## Subversion Working Copies Information



## UpdateInfo(更新日志)



## History



## 常用操作

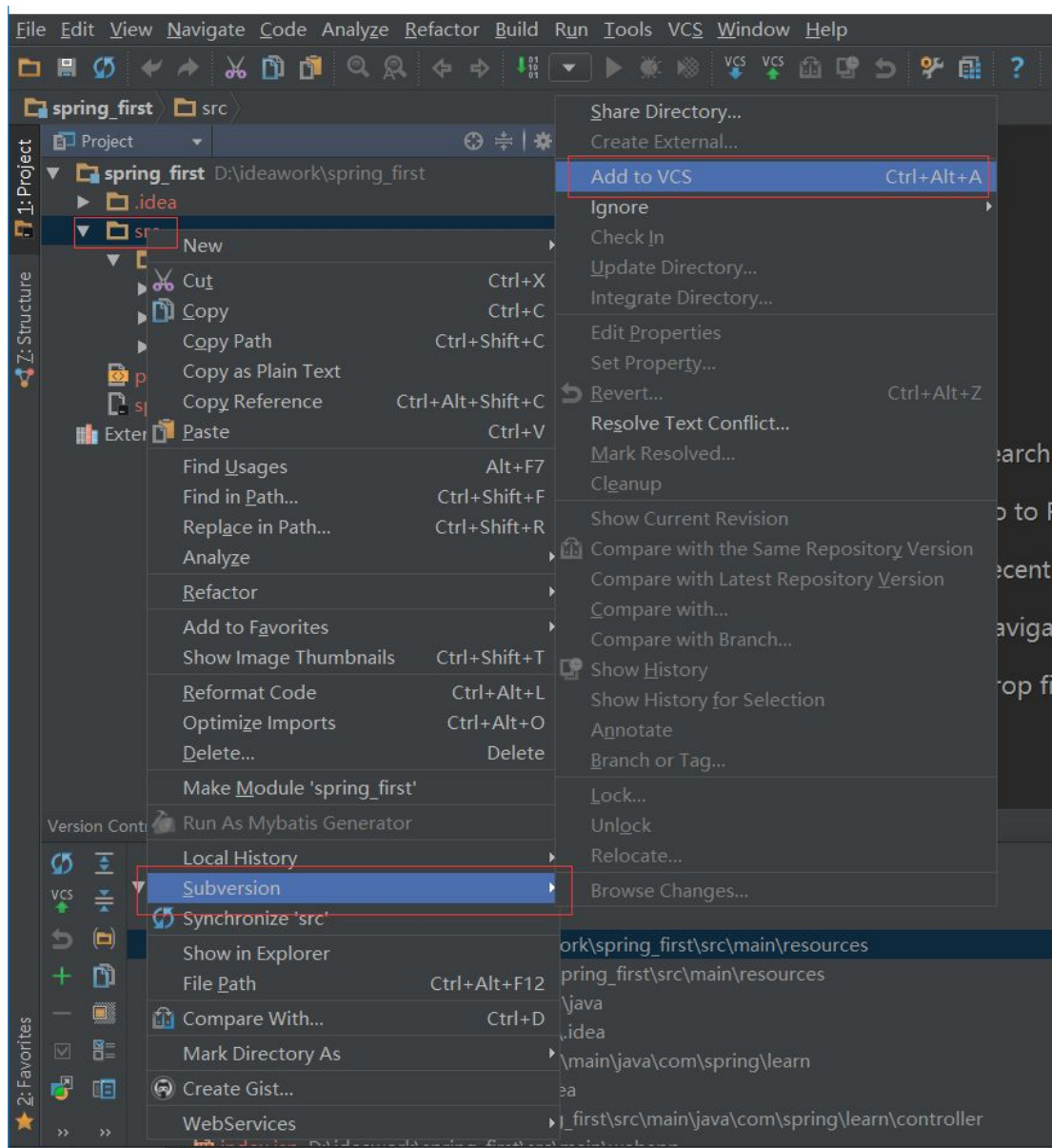
红色是未被版本控制  , 绿色是新加入版本 

蓝色是修改过的 

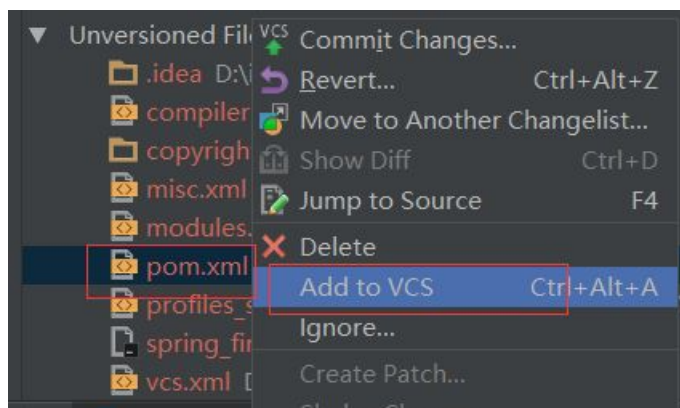
## 加入版本控制

第一种:



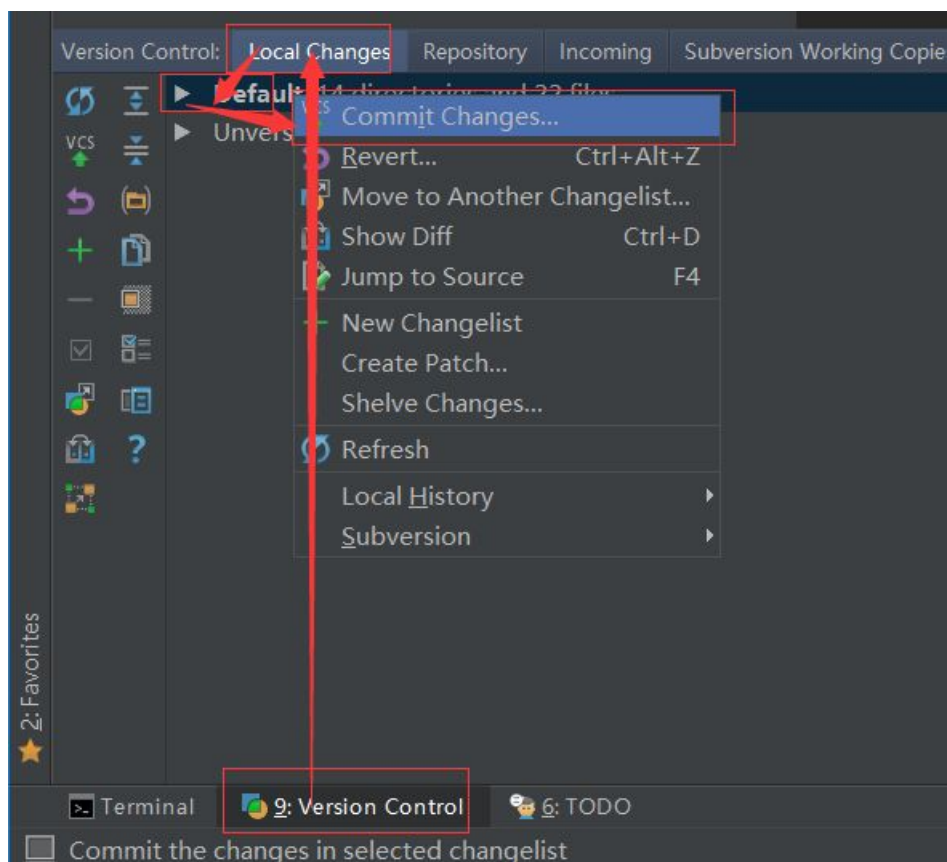


第二种

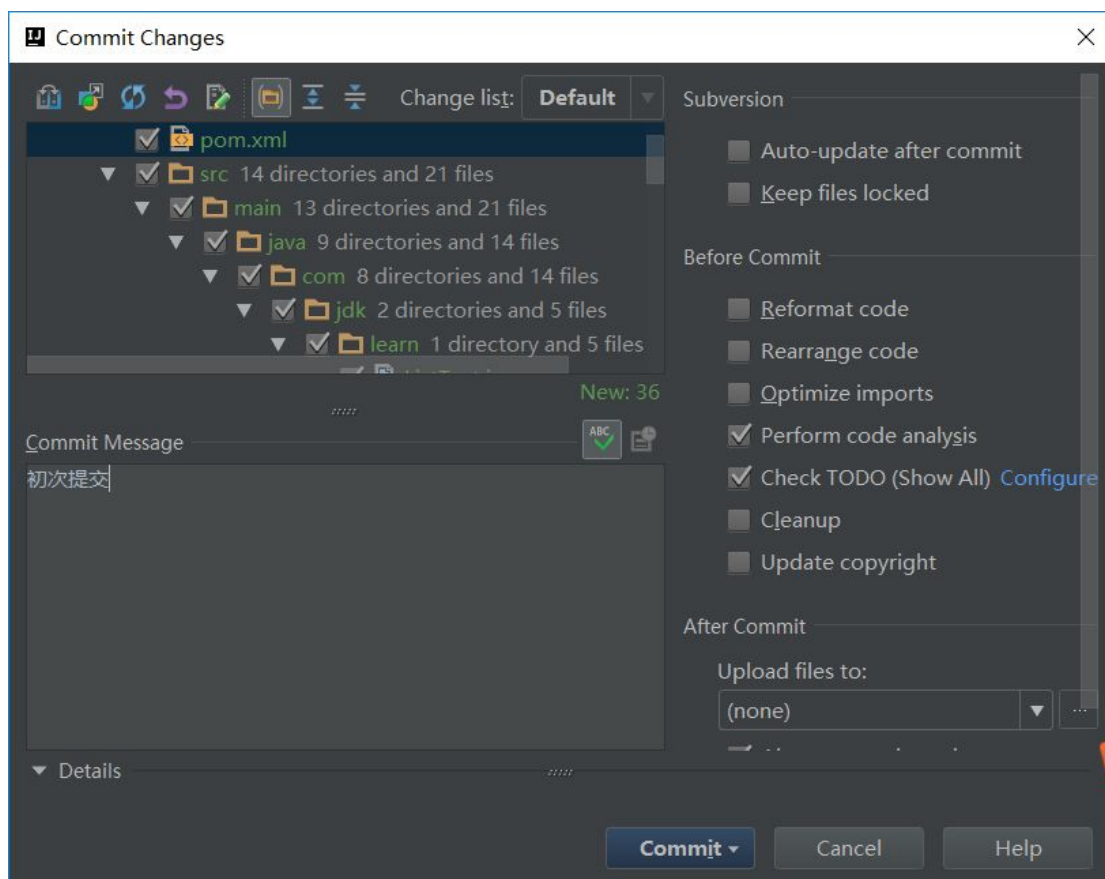


## 提交远程

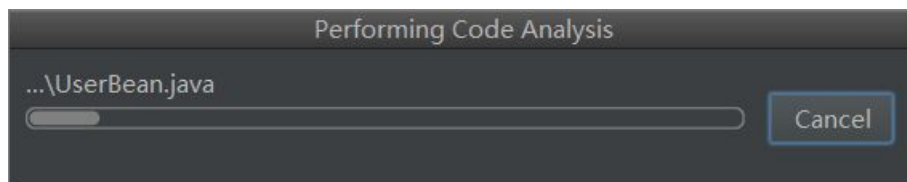
在 local changes 面板中，选择 default



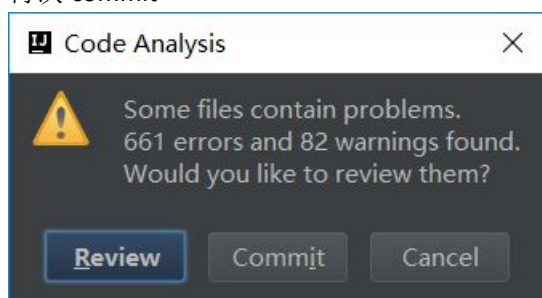
提交选项



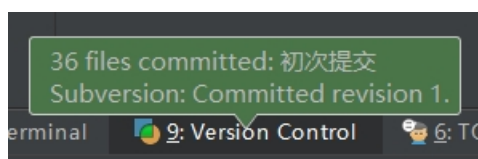
点击 commit



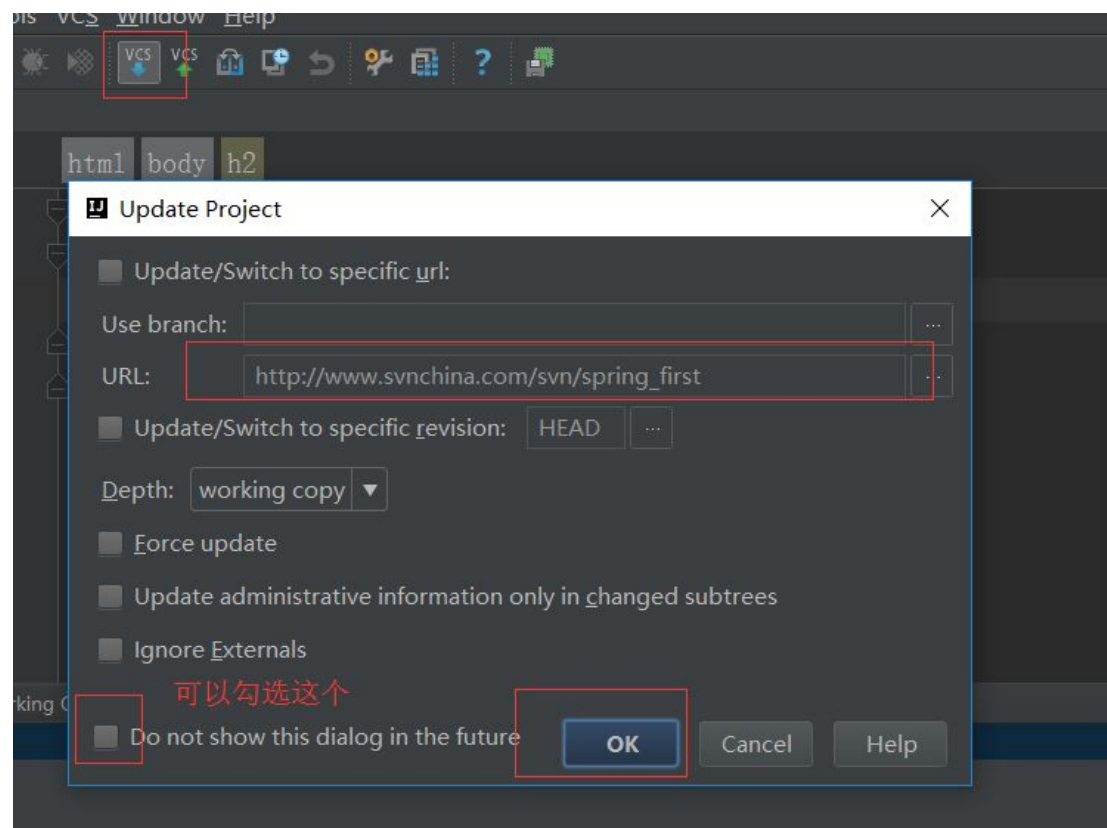
再次 commit



提交成功



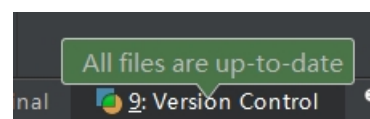
## 更新项目



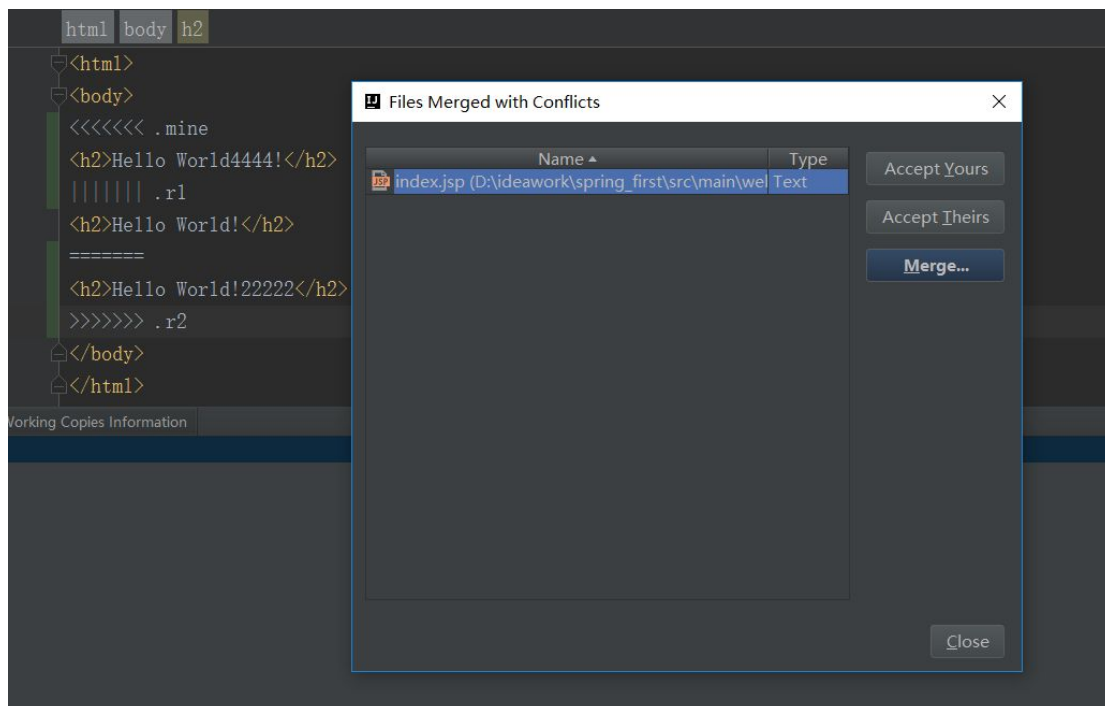
### 更新进度



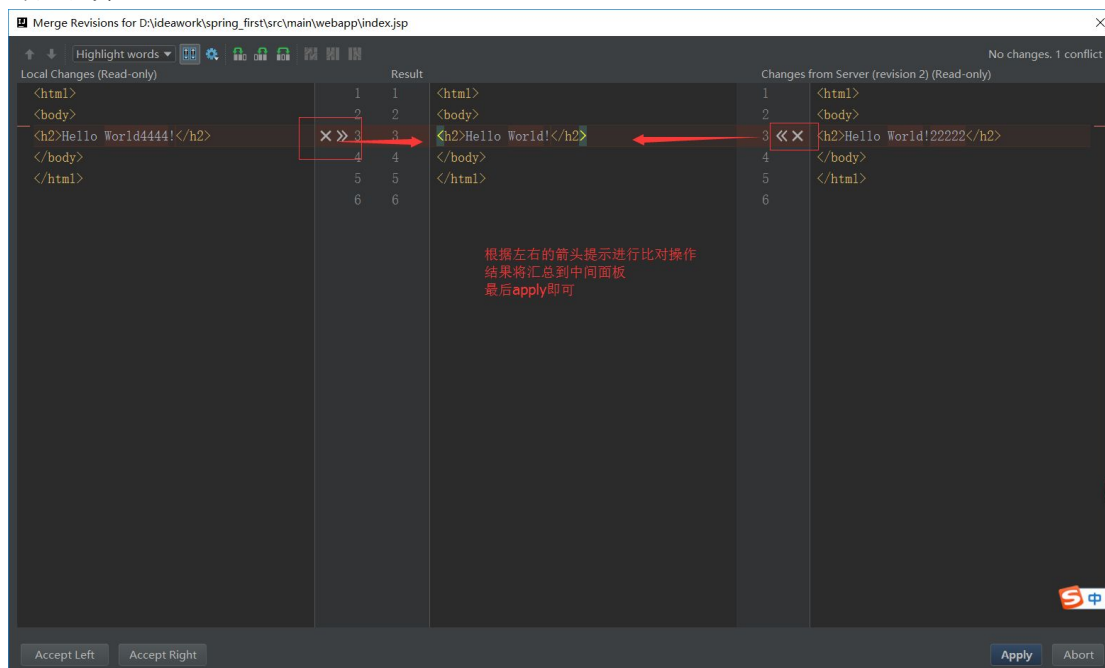
### 更新结果



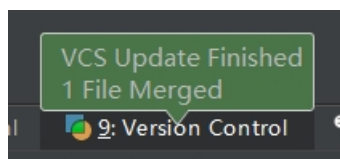
## 冲突解决



## 冲突合并



## 处理结果



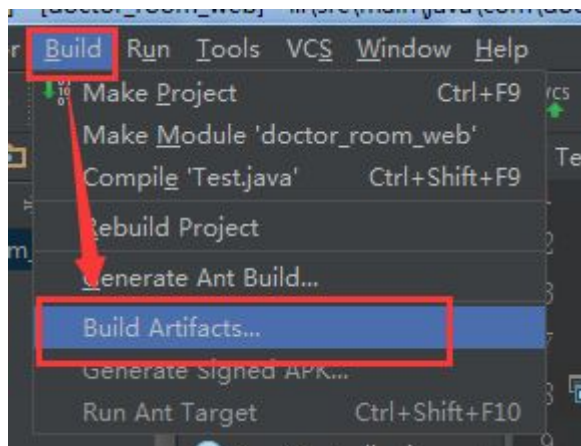
## 常用技巧&问题

### JSP 实时编译问题

在某些情况下，jsp 和 js 等静态资源更改时，不能实时生效。

1.可以通过 ctrl+s 达到实时编译。

2.通过如下方式进行构建。

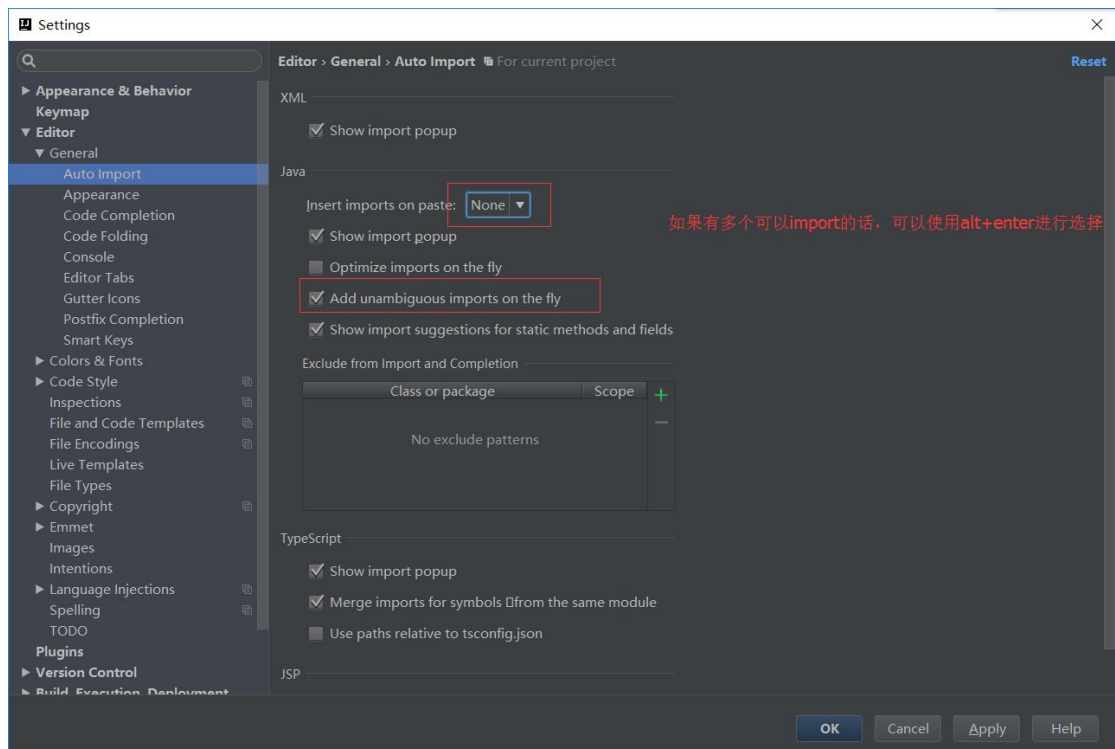


### 搜索功能失效

ctrl+shift+f 失效，一般是快捷键冲突，比如和搜狗输入法的冲突，修改快捷键即可。

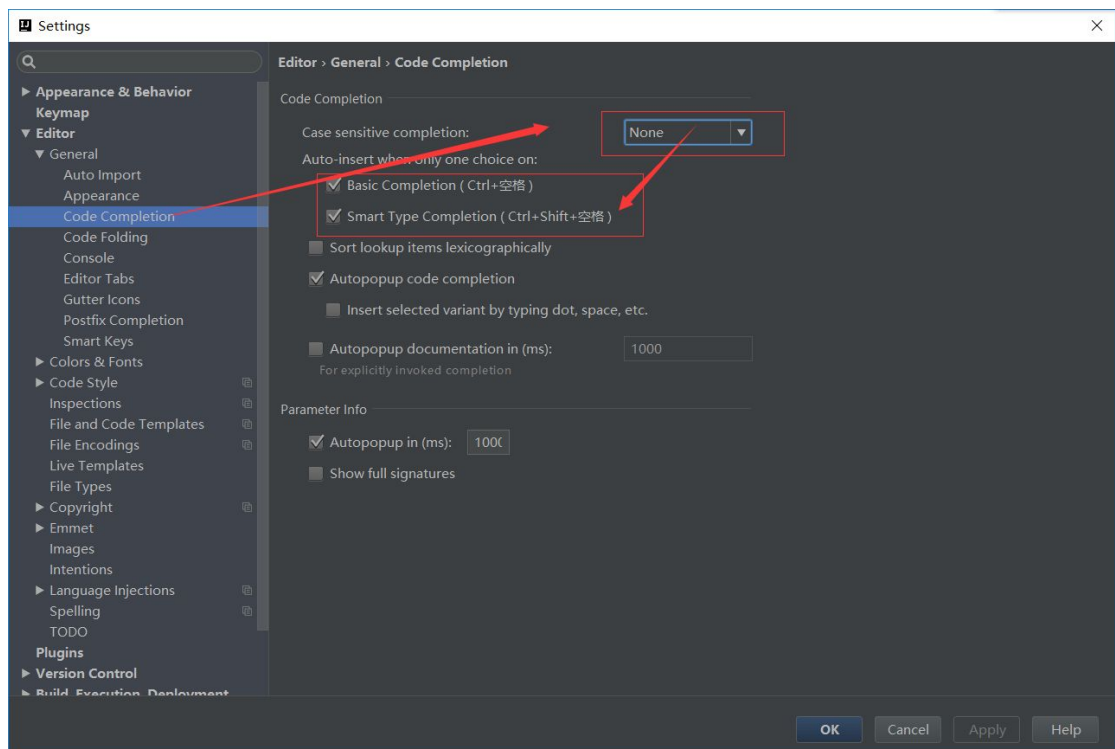
### 自动导入依赖

配置为下图设置，



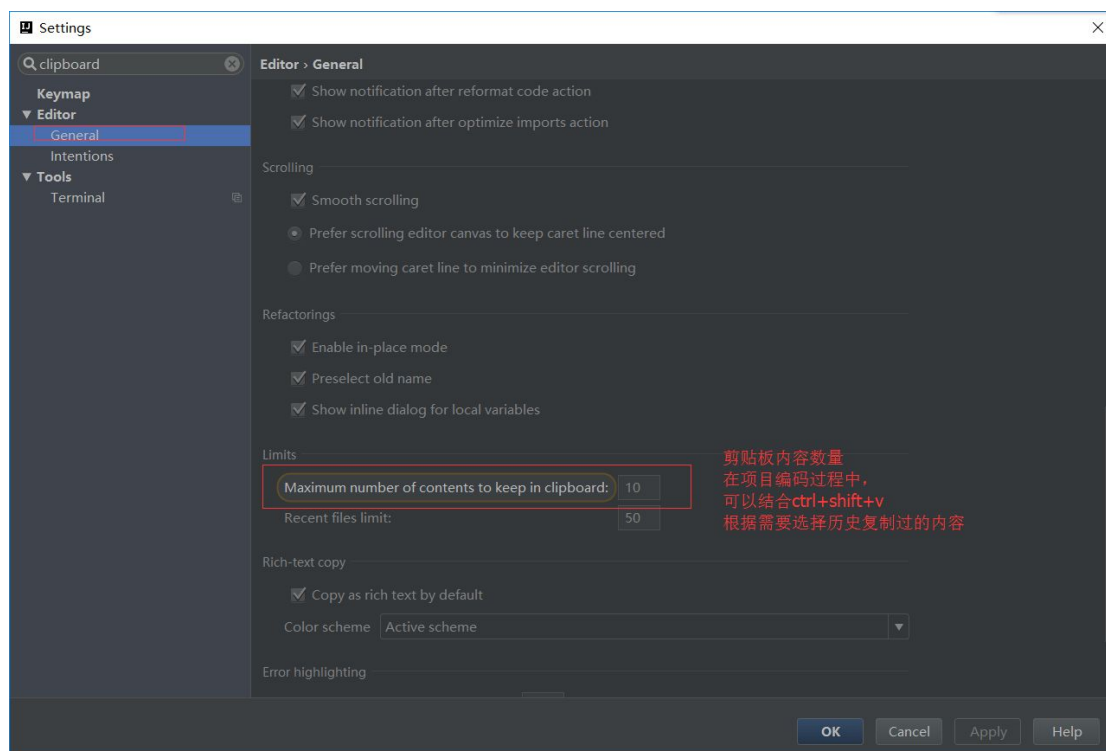
## 提示不区分大小写

修改为如下图配置即可，

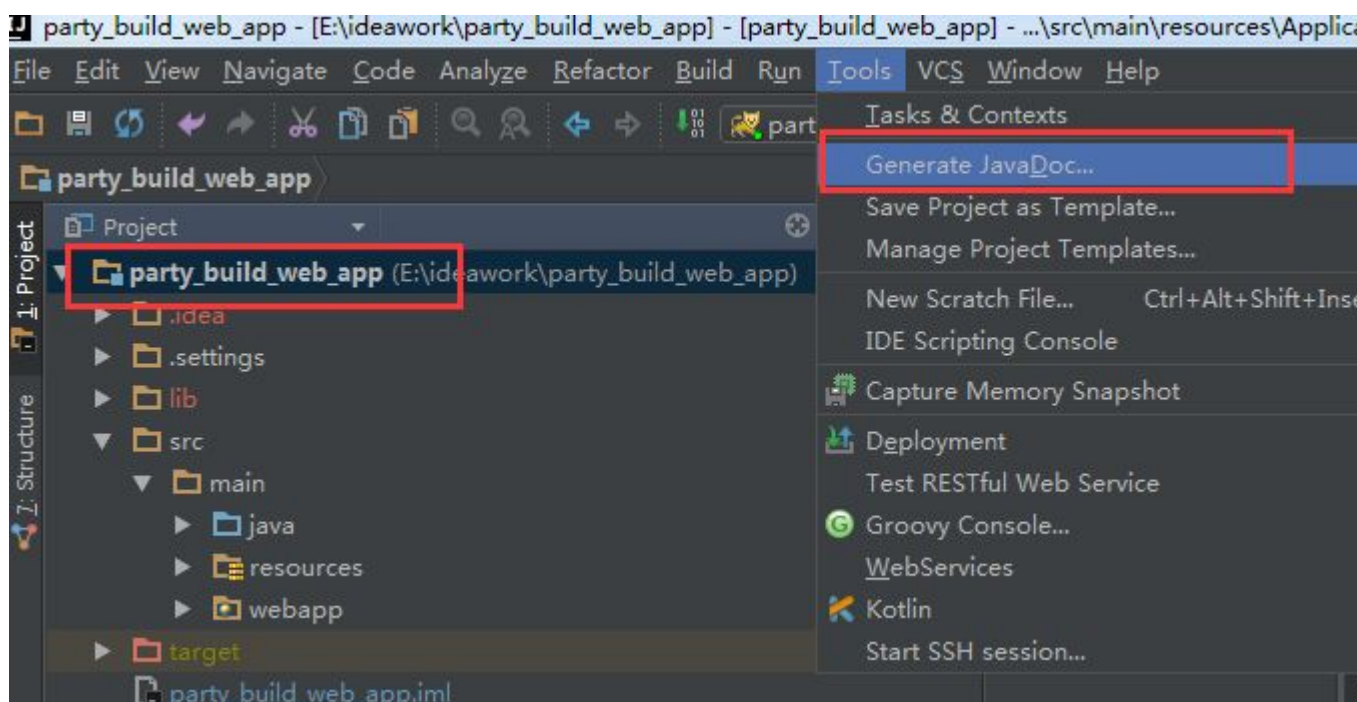




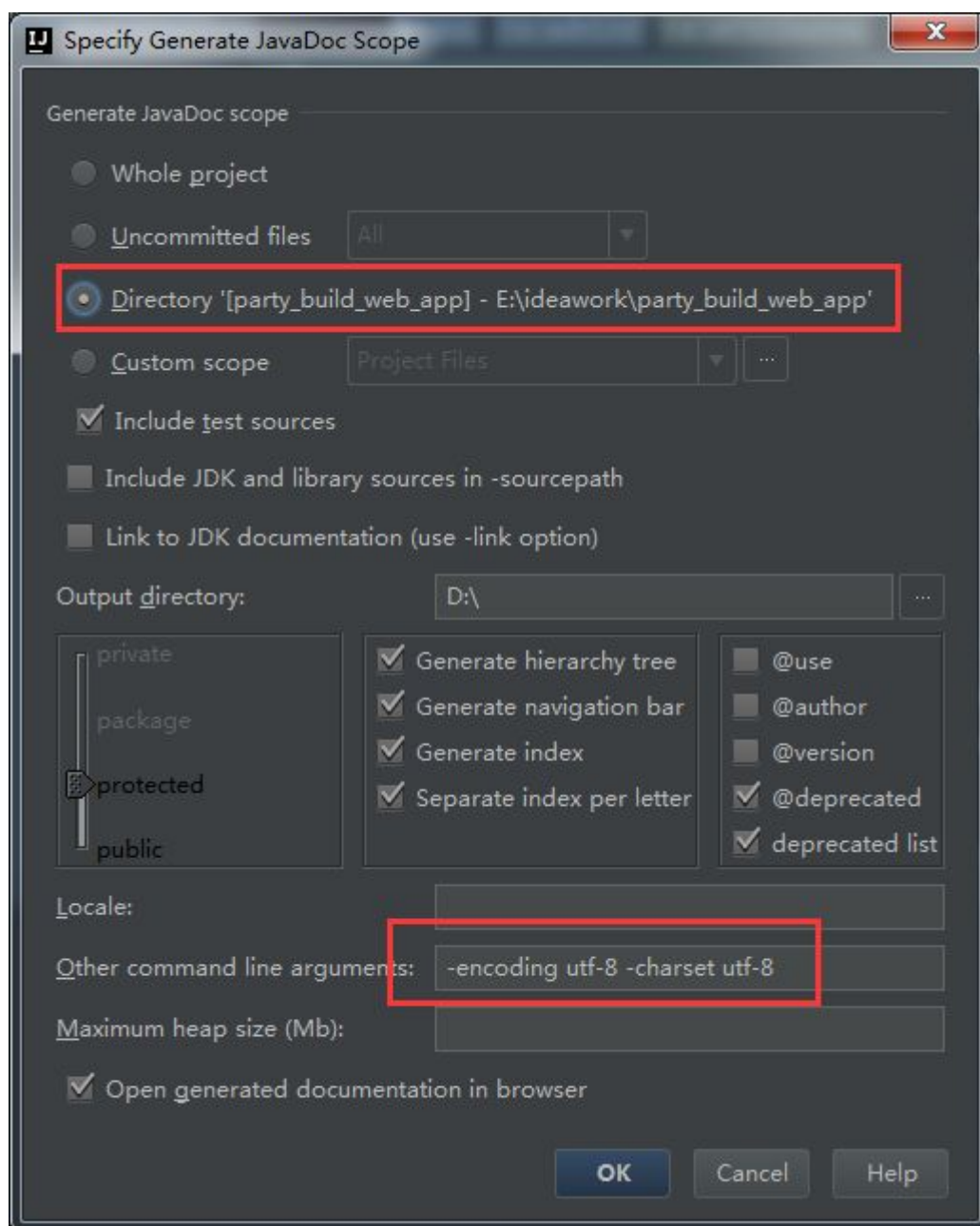
## 剪贴板数量设置



## 生成 JAVADOC

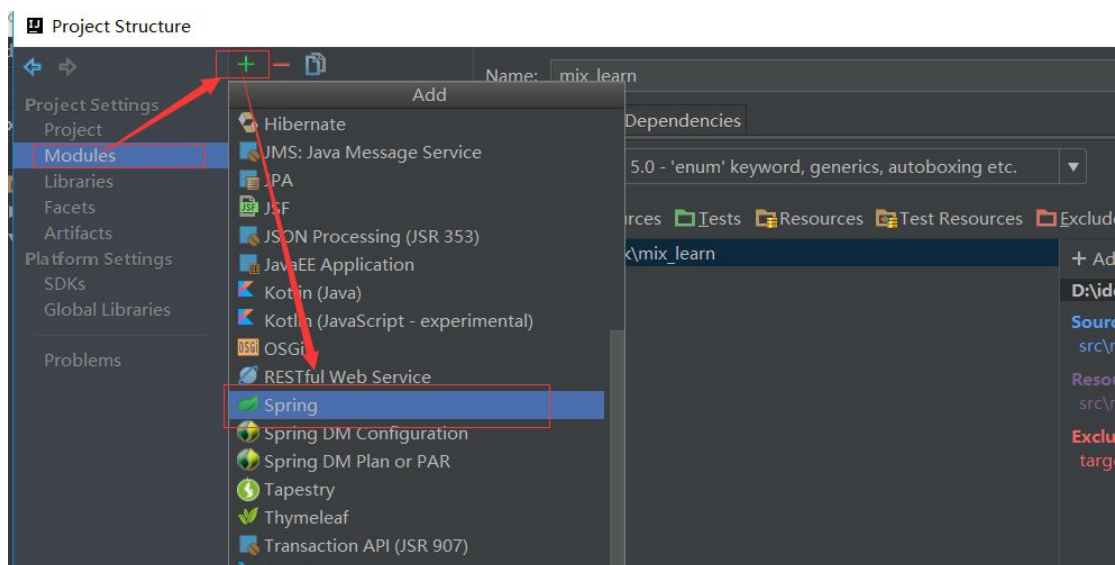




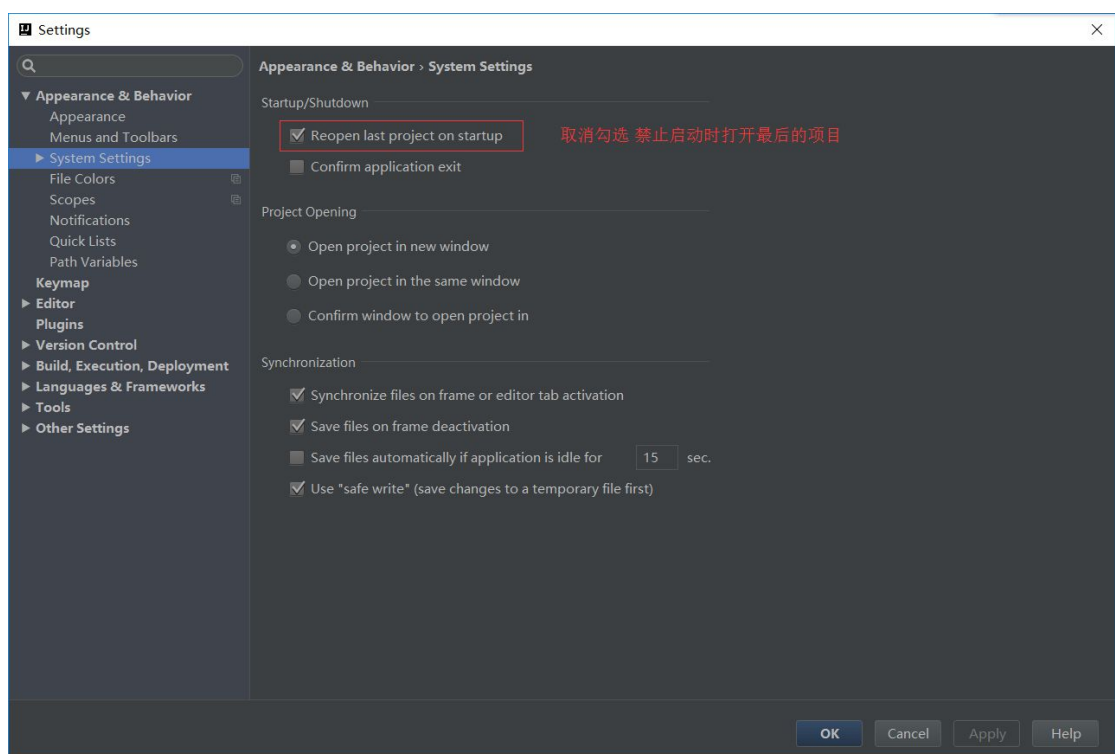


## 设置 Spring 支持

可以根据方法，直接调到对应的 jsp 页面

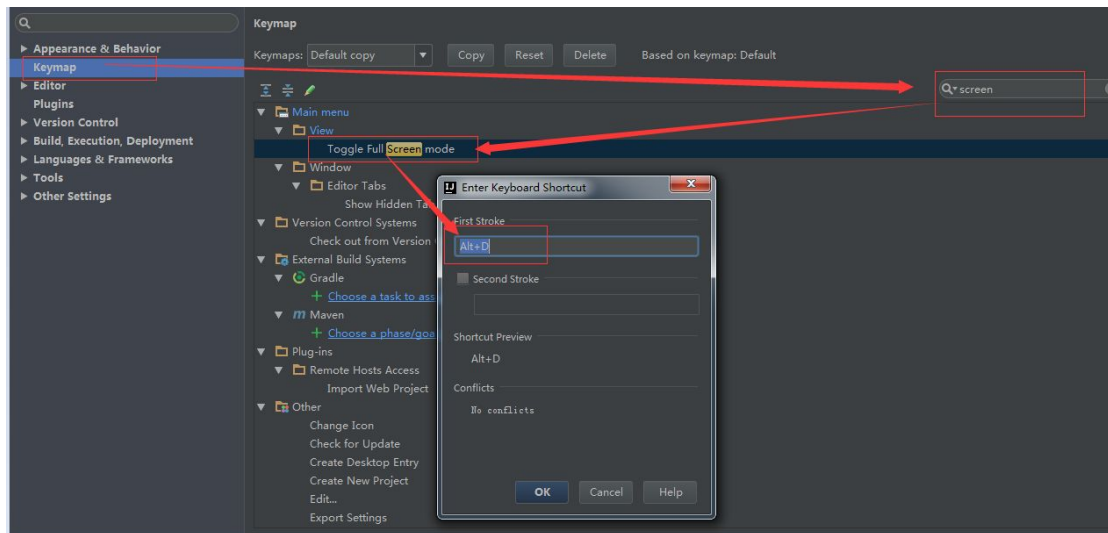


## 启动时不自动打开项目



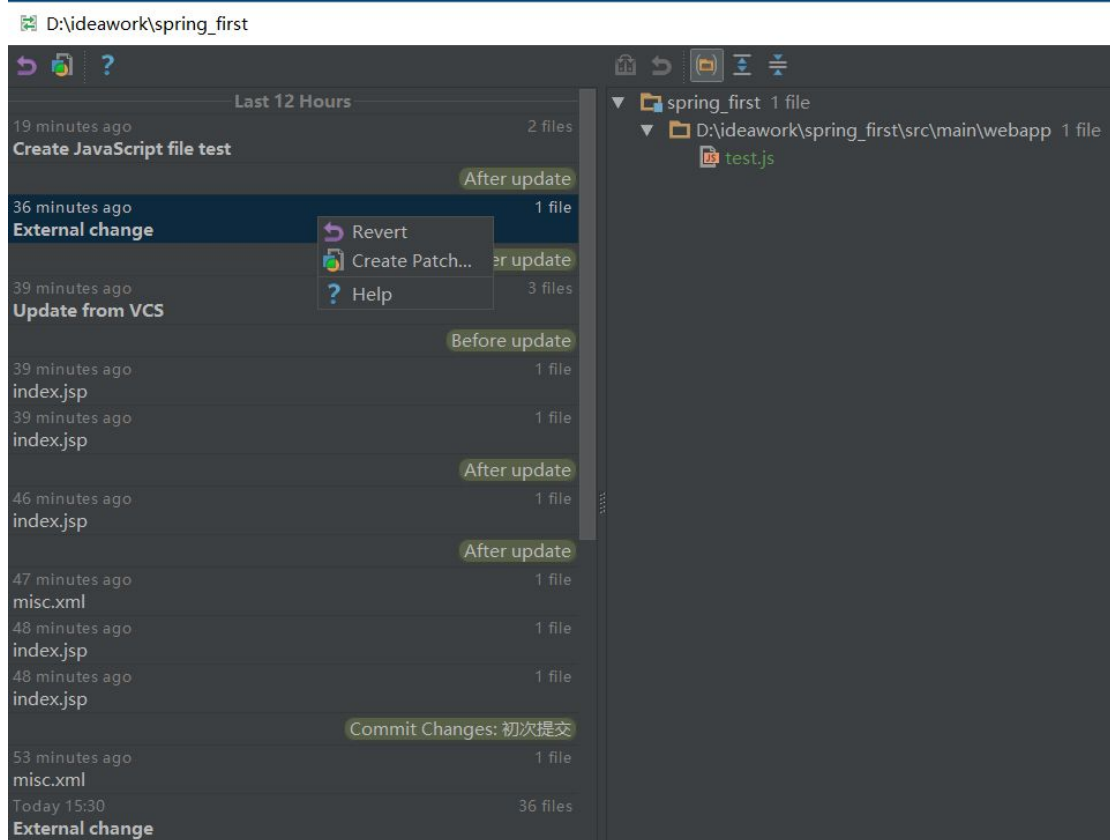
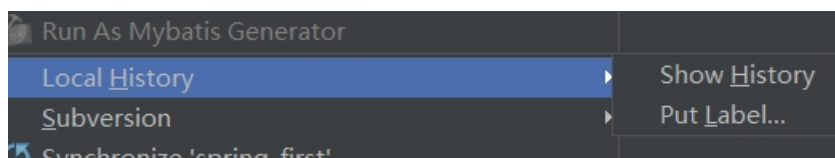
## 全屏设置

通过如下配置, 可以使用 **alt+d** 切换全屏



## 本地历史

右键文件或者项目，可以进行历史找回和还原

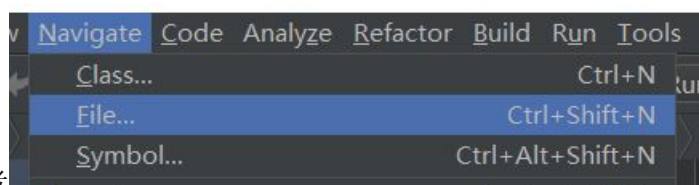


## 搜索

## 所有文件

双击 shift，或者选择右上角面板搜索按钮。

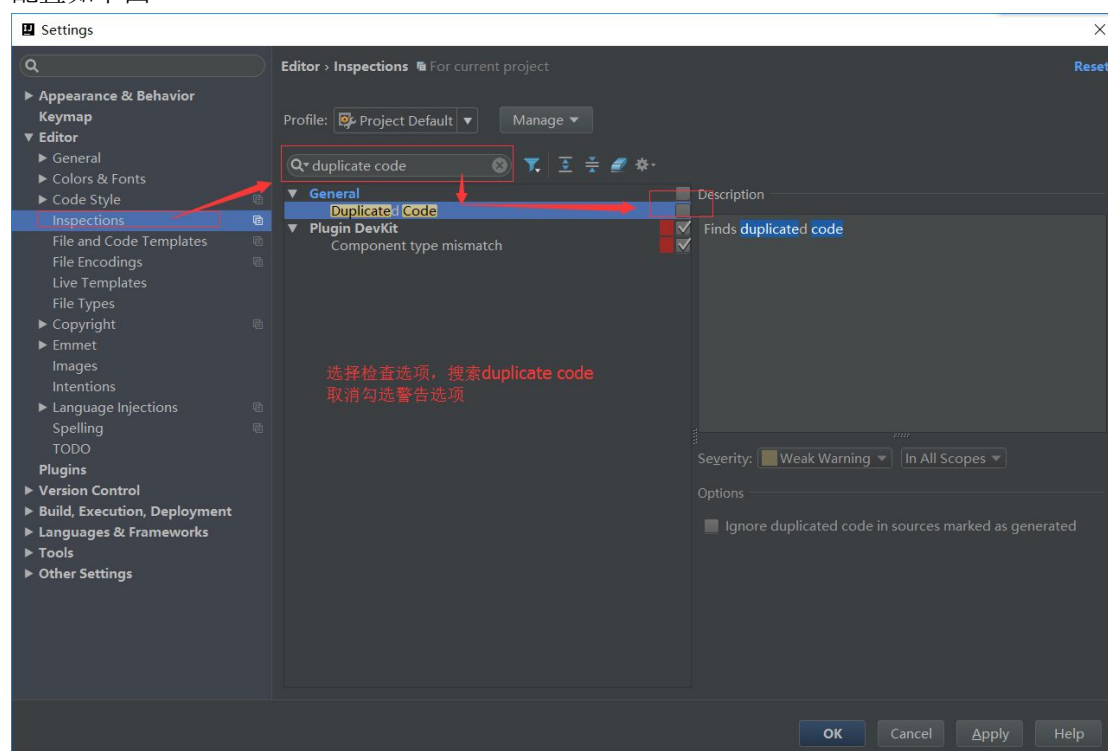
## 项目文件



Ctrl+shift+n 或者

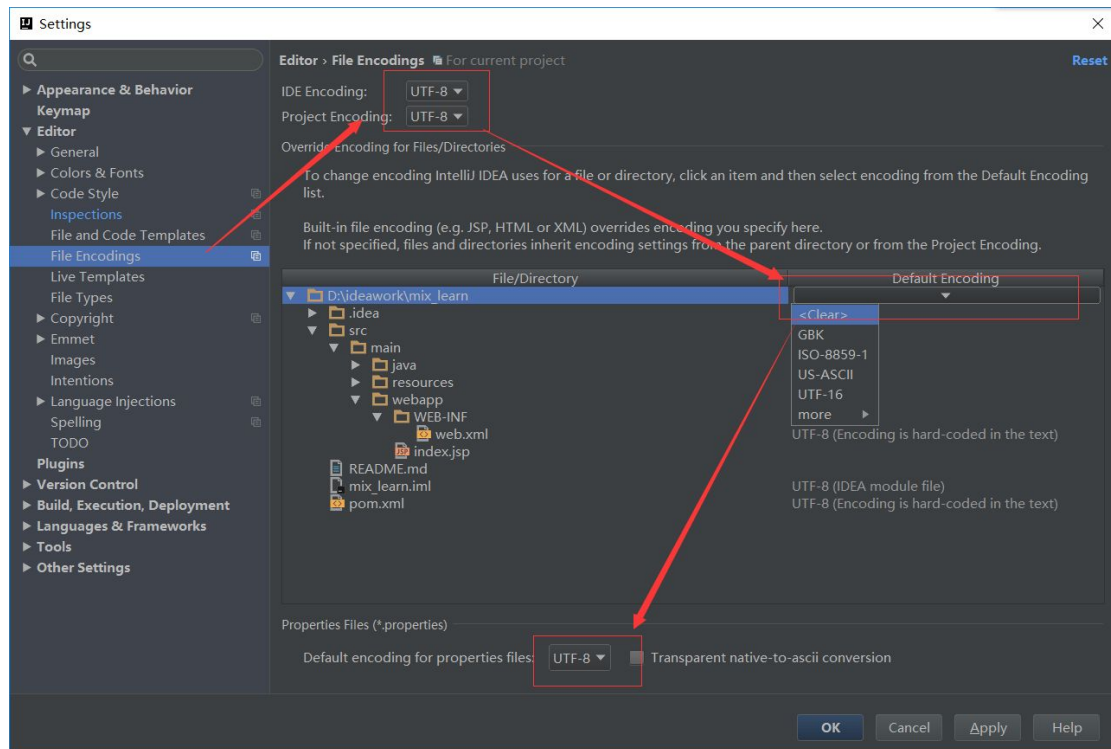
## 取消重复代码提示

配置如下图



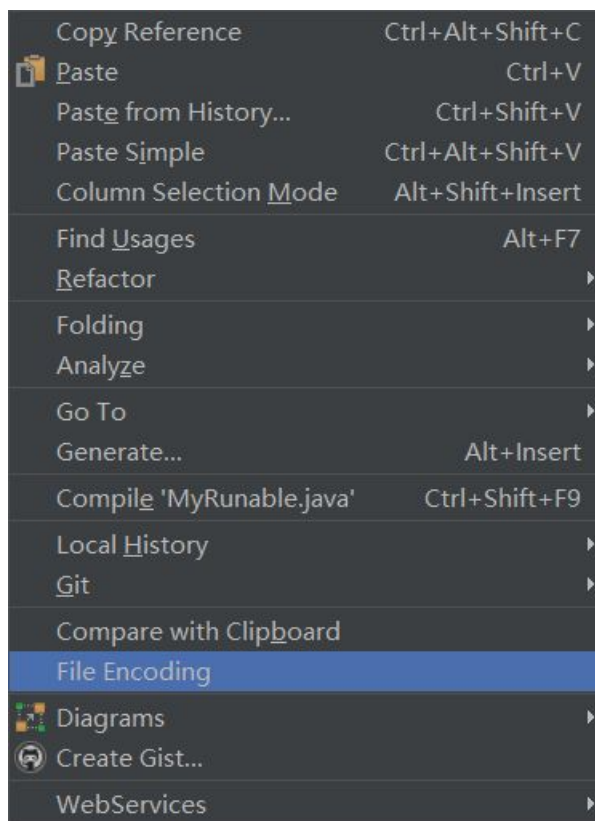
## 设置字符集

### 项目字符集



### 单个文件

右键，选择菜单中的 file encoding



# 快捷键

## Ctrl

快捷键	介绍
Ctrl + F	在当前文件进行文本查找 (必备)
Ctrl + R	在当前文件进行文本替换 (必备)
Ctrl + Z	撤销 (必备)
Ctrl + Y	删除光标所在行 或 删除选中的行 (必备)
Ctrl + X	剪切光标所在行 或 剪切选择内容
Ctrl + C	复制光标所在行 或 复制选择内容
Ctrl + D	复制光标所在行 或 复制选择内容, 并把复制内容插入光标位置下面 (必备)
Ctrl + W	递进式选择代码块。可选中光标所在的单词或段落, 连续按会在原有选中的基础上再扩展选中范围 (必备)
Ctrl + E	显示最近打开的文件记录列表 (必备)
Ctrl + N	根据输入的 类名 查找类文件 (必备)
Ctrl + G	在当前文件跳转到指定行处
Ctrl + J	插入自定义动态代码模板 (必备)
Ctrl + P	方法参数提示显示 (必备)

快捷键	介绍
Ctrl + Q	光标所在的变量 / 类名 / 方法名等上面（也可以在提示补充的时候按），显示文档内容
Ctrl + U	前往当前光标所在的方法的父类的方法 / 接口定义 （必备）
Ctrl + B	进入光标所在的方法/变量的接口或是定义处，等效于 Ctrl + 左键单击 （必备）
Ctrl + K	版本控制提交项目，需要此项目有加入到版本控制才可用
Ctrl + T	版本控制更新项目，需要此项目有加入到版本控制才可用
Ctrl + H	显示当前类的层次结构
Ctrl + O	选择可重写的方法
Ctrl + I	选择可继承的方法
Ctrl + +	展开代码
Ctrl + -	折叠代码
Ctrl + /	注释光标所在行代码，会根据当前不同文件类型使用不同的注释符号 （必备）
Ctrl + [	移动光标到当前所在代码的花括号开始位置
Ctrl + ]	移动光标到当前所在代码的花括号结束位置
Ctrl + F1	在光标所在的错误代码处显示错误信息 （必备）
Ctrl + F3	调到所选中的词的下一个引用位置 （必备）
Ctrl + F4	关闭当前编辑文件
Ctrl + F8	在 Debug 模式下，设置光标当前行为断点，如果当前已经是断点则去掉断点
Ctrl + F9	执行 Make Project 操作
Ctrl + F11	选中文件 / 文件夹，使用助记符设定 / 取消书签 （必备）
Ctrl + F12	弹出当前文件结构层，可以在弹出的层上直接输入，进行筛选
Ctrl + Tab	编辑窗口切换，如果在切换的过程又加按上 delete，则是关闭对应选中的窗口
Ctrl + End	跳到文件尾
Ctrl + Home	跳到文件头
Ctrl + Space	基础代码补全，默认在 Windows 系统上被输入法占用，需要进行修改，建议修改为 Ctrl + 逗号 （必备）
Ctrl + Delete	删除光标后面的单词或是中文句 （必备）
Ctrl + BackSpace	删除光标前面的单词或是中文句 （必备）
Ctrl + 1, 2, 3...9	定位到对应数值的书签位置 （必备）
Ctrl + 左键单击	在打开的文件标题上，弹出该文件路径 （必备）
Ctrl + 光标定位按	Ctrl 不要松开，会显示光标所在的类信息摘要
Ctrl + 左方向键	光标跳转到当前单词 / 中文句的左侧开头位置 （必备）
Ctrl + 右方向键	光标跳转到当前单词 / 中文句的右侧开头位置 （必备）
Ctrl + 前方向键	等效于鼠标滚轮向前效果 （必备）
Ctrl + 后方向键	等效于鼠标滚轮向后效果 （必备）

## Alt

快捷键	介绍
Alt + `	显示版本控制常用操作菜单弹出层 （必备）
Alt + Q	弹出一个提示，显示当前类的声明 / 上下文信息
Alt + F1	显示当前文件选择目标弹出层，弹出层中有很多目标可以进行选择 （必备）



快捷键	介绍
Alt + F2	对于前面页面，显示各类浏览器打开目标选择弹出层
Alt + F3	选中文本，逐个往下查找相同文本，并高亮显示
Alt + F7	查找光标所在的方法 / 变量 / 类被调用的地方
Alt + F8	在 Debug 的状态下，选中对象，弹出可输入计算表达式调试框，查看该输入内容的调试结果
Alt + Home	定位 / 显示到当前文件的 Navigation Bar
Alt + Enter	IntelliJ IDEA 根据光标所在问题，提供快速修复选择，光标放在的位置不同提示的结果也不同（必备）
Alt + Insert	代码自动生成，如生成对象的 set / get 方法，构造函数，toString() 等（必备）
Alt + 左方向键	切换当前已打开的窗口中的子视图，比如 Debug 窗口中有 Output、Debugger 等子视图，用此快捷键就可以在子视图中切换（必备）
Alt + 右方向键	按切换当前已打开的窗口中的子视图，比如 Debug 窗口中有 Output、Debugger 等子视图，用此快捷键就可以在子视图中切换（必备）
Alt + 前方向键	当前光标跳转到当前文件的前一个方法名位置（必备）
Alt + 后方向键	当前光标跳转到当前文件的后一个方法名位置（必备）
Alt + 1, 2, 3...9	显示对应数值的选项卡，其中 1 是 Project 用得最多（必备）

## Shift

快捷键	介绍
Shift + F1	如果有外部文档可以连接外部文档
Shift + F2	跳转到上一个高亮错误 或 警告位置
Shift + F3	在查找模式下，查找匹配上一个
Shift + F4	对当前打开的文件，使用新 Windows 窗口打开，旧窗口保留
Shift + F6	对文件 / 文件夹 重命名
Shift + F7	在 Debug 模式下，智能步入。断点所在行上有多个方法调用，会弹出进入哪个方法
Shift + F8	在 Debug 模式下，跳出，表现出来的效果跟 F9 一样
Shift + F9	等效于点击工具栏的 Debug 按钮
Shift + F10	等效于点击工具栏的 Run 按钮
Shift + F11	弹出书签显示层（必备）
Shift + Tab	取消缩进（必备）
Shift + ESC	隐藏当前 或 最后一个激活的工具窗口
Shift + End	选中光标到当前行尾位置
Shift + Home	选中光标到当前行头位置
Shift + Enter	开始新一行。光标所在行下空出一行，光标定位到新行位置（必备）
Shift + 左键单击	在打开的文件名上按此快捷键，可以关闭当前打开文件（必备）
Shift + 滚轮前后滚动	当前文件的横向滚动轴滚动（必备）



## Ctrl + Alt

快捷键	介绍
Ctrl + Alt + L	格式化代码, 可以对当前文件和整个包目录使用 (必备)
Ctrl + Alt + O	优化导入的类, 可以对当前文件和整个包目录使用 (必备)
Ctrl + Alt + I	光标所在行 或 选中部分进行自动代码缩进, 有点类似格式化
Ctrl + Alt + T	对选中的代码弹出环绕选项弹出层 (必备)
Ctrl + Alt + J	弹出模板选择窗口, 将选定的代码加入动态模板中
Ctrl + Alt + H	调用层次
Ctrl + Alt + B	在某个调用的方法名上使用会跳到具体的实现处, 可以跳过接口
Ctrl + Alt + V	快速引进变量
Ctrl + Alt + Y	同步、刷新
Ctrl + Alt + S	打开 IntelliJ IDEA 系统设置 (必备)
Ctrl + Alt + F7	显示使用的地方。寻找被该类或是变量被调用的地方, 用弹出框的方式找出来
Ctrl + Alt + F11	切换全屏模式
Ctrl + Alt + Enter	光标所在行上空出一行, 光标定位到新行 (必备)
Ctrl + Alt + Home	弹出跟当前文件有关联的文件弹出层
Ctrl + Alt + Space	类名自动完成
Ctrl + Alt + 左方向键	退回到上一个操作的地方 (必备)
Ctrl + Alt + 右方向键	前进到上一个操作的地方 (必备)
Ctrl + Alt + 前方向键	在查找模式下, 跳到上个查找的文件
Ctrl + Alt + 后方向键	在查找模式下, 跳到下个查找的文件

## Ctrl + Shift

快捷键	介绍
Ctrl + Shift + F	根据输入内容查找整个项目 或 指定目录内文件 (必备)
Ctrl + Shift + R	根据输入内容替换对应内容, 范围为整个项目 或 指定目录内文件 (必备)
Ctrl + Shift + J	自动将下一行合并到当前行末尾 (必备)
Ctrl + Shift + Z	取消撤销 (必备)
Ctrl + Shift + W	递进式取消选择代码块。可选中光标所在的单词或段落, 连续按会在原有选中的基础上再扩展取消选中范围 (必备)
Ctrl + Shift + N	通过文件名定位 / 打开文件 / 目录, 打开目录需要在输入的内容后面多加一个正斜杠 (必备)
Ctrl + Shift + U	对选中的代码进行大 / 小写轮流转换 (必备)
Ctrl + Shift + T	对当前类生成单元测试类, 如果已经存在的单元测试类则可以进行选择 (必备)
Ctrl + Shift + C	复制当前文件磁盘路径到剪贴板 (必备)
Ctrl + Shift + V	弹出缓存的最近拷贝的内容管理器弹出层
Ctrl + Shift + E	显示最近修改的文件列表的弹出层
Ctrl + Shift + H	显示方法层次结构
Ctrl + Shift + B	跳转到类型声明处 (必备)
Ctrl + Shift + I	快速查看光标所在的方法 或 类的定义
Ctrl + Shift + A	查找动作 / 设置

快捷键	介绍
Ctrl + Shift + /	代码块注释 （必备）
Ctrl + Shift + [	选中从光标所在位置到它的顶部中括号位置 （必备）
Ctrl + Shift + ]	选中从光标所在位置到它的底部中括号位置 （必备）
Ctrl + Shift + +	展开所有代码 （必备）
Ctrl + Shift + -	折叠所有代码 （必备）
Ctrl + Shift + F7	高亮显示所有该选中文本，按 Esc 高亮消失 （必备）
Ctrl + Shift + F8	在 Debug 模式下，指定断点进入条件
Ctrl + Shift + F9	编译选中的文件 / 包 / Module
Ctrl + Shift + F12	编辑器最大化 （必备）
Ctrl + Shift + Space	智能代码提示
Ctrl + Shift + Enter	自动结束代码，行末自动添加分号 （必备）
Ctrl + Shift + Backspace	退回到上次修改的地方 （必备）
Ctrl + Shift + 1, 2, 3...9	快速添加指定数值的书签 （必备）
Ctrl + Shift + 左键单击	把光标放在某个类变量上，按此快捷键可以直接定位到该类中 （必备）
Ctrl + Shift + 左方向键	在代码文件上，光标跳转到当前单词 / 中文句的左侧开头位置，同时选中该单词 / 中文句 （必备）
Ctrl + Shift + 右方向键	在代码文件上，光标跳转到当前单词 / 中文句的右侧开头位置，同时选中该单词 / 中文句 （必备）
Ctrl + Shift + 前方向键	光标放在方法名上，将方法移动到上一个方法前面，调整方法排序 （必备）
Ctrl + Shift + 后方向键	光标放在方法名上，将方法移动到下一个方法前面，调整方法排序 （必备）

## Alt + Shift

快捷键	介绍
Alt + Shift + N	选择 / 添加 task （必备）
Alt + Shift + F	显示添加到收藏夹弹出层 / 添加到收藏夹
Alt + Shift + C	查看最近操作项目的变化情况列表
Alt + Shift + I	查看项目当前文件
Alt + Shift + F7	在 Debug 模式下，下一步，进入当前方法体内，如果方法体还有方法，则会进入该内嵌的方法中，依此循环进入
Alt + Shift + F9	弹出 Debug 的可选择菜单
Alt + Shift + F10	弹出 Run 的可选择菜单
Alt + Shift + 左键双击	选择被双击的单词 / 中文句，按住不放，可以同时选择其他单词 / 中文句 （必备）
Alt + Shift + 前方向键	移动光标所在行向上移动 （必备）
Alt + Shift + 后方向键	移动光标所在行向下移动 （必备）

## Ctrl + Shift + Alt

快捷键	介绍
Ctrl + Shift + Alt + V	无格式黏贴 (必备)
Ctrl + Shift + Alt + N	前往指定的变量 / 方法
Ctrl + Shift + Alt + S	打开当前项目设置 (必备)
Ctrl + Shift + Alt + C	复制参考信息

## 其他

快捷键	介绍
F2	跳转到下一个高亮错误 或 警告位置 (必备)
F3	在查找模式下, 定位到下一个匹配处
F4	编辑源 (必备)
F7	在 Debug 模式下, 进入下一步, 如果当前行断点是一个方法, 则进入当前方法体内, 如果该方法体还有方法, 则不会进入该内嵌的方法中
F8	在 Debug 模式下, 进入下一步, 如果当前行断点是一个方法, 则不进入当前方法体内
F9	在 Debug 模式下, 恢复程序运行, 但是如果该断点下面代码还有断点则停在下一个断点上
F11	添加书签 (必备)
F12	回到前一个工具窗口 (必备)
Tab	缩进 (必备)
ESC	从工具窗口进入代码文件窗口 (必备)
连接两次 Shift	弹出 Search Everywhere 弹出层