

1.设置maven

1. 在File->settings->搜索maven
2. Mavan home directory--设置maven安装包的bin文件夹所在的位置
3. User settings file--设置setting文件所在的位置
4. Local repository--设置本地仓库的

2.IDEA 设置代码行宽度

1. 在File->settings->Editor->Code Style
2. 有人会问，如果输入的代码超出宽度界线时，如何让IDE自动将代码换行？有两种方式！
3. 第一种，在上述的“Right margin (columns)”的下方，有“wrap when typing reaches right margin”选项，选中它，是什么效果呢？
4. 随着输入的字符的增加，当代码宽度到达界线时，IDEA会自动将代码换行。
5. 第一种方式是在输入代码时触发，还有第二种方式，在File->settings->Code Style->Java中，选中“Wrapping and Braces”选项卡，
6. 在“Keep when reformatting”中有一个“Ensure rigth margin is not exceeded”，选中它，是什么效果呢？
7. 从配置项的字面意思很容易理解，在格式化Java代码时，确保代码没有超过宽度界线。
8. 即输入的代码超出界线后，

****3.IDEA**** 提示不区分大小写**

1. 首先打开File----->setting
2. 然后，输入：sensitive
3. 将右侧的 case sensitive completion 修改为NONE

4.Intellij强制更新Maven Dependencies

1. IntelliJ 自动载入Mave依赖的功能很好用，但有时候会碰到问题，导致pom文件修改却没有触发自动重新载入的动作，此时需要手动强制更新依赖。
2. 如下：
 1. 手动删除Project Settings里面的Libraries内容；
 2. 在Maven Project的试图里clean一下，删除之前编译过的文件；
 3. 项目右键-》Maven-》Reimport
 4. Ok， 此时发现依赖已经建立！

5.idea的环境配置默认保存位置

1. idea的环境配置默认保存位置:C:\Users\xxxxxxx\IntelliJ Idea14 ,xxxxxx代表用户目录,
2. 可以对该目录进行备份,一旦环境出问题恢复此配置即可.
3. 可以在%IDEA_HOME%/bin/idea.properties中修改该配置路径.

6.隐藏不想看到的文件或者文件夹（类似eclipse的filter功能）

1. intellij idea 隐藏不想看到的文件或者文件夹（类似eclipse的filter功能）
- 2.
3. 打开intellij -->:File -->Settings-->搜索File Types

****7.修改为Eclipse快捷键******

1. File -> Settings -> Keymap => Keymaps改为 Eclipse copy

8.修改默认设置--default setting

1. 修改默认设置--default setting

9. 修改智能提示快捷键

1. File -> Settings -> Keymap -> Main menu -> Code -> Completion -> Basic=>修改为Ctrl+Alt+Enter
- 2.
3. 保存时把冲突的Remove掉。
- 4.
5. File -> Settings -> Keymap -> Editor Actions -> Complete Current Statement=>修改为Ctrl+;

10.查找快捷键冲突问题处理****

1. File -> Settings -> Keymap -> Main menu -> Edit ->Find =>修改Find...和Replace...分别改为Ctrl+F 和Ctrl+R

11.显示行号

1. File -> Settings -> Editor -> General -> Appearance => Show line numbers 选中

12.**代码智能提示，忽略大小写**

1. File -> Settings -> Editor -> Code Completion 里把 Case sensitive completion 设置为 None 就可以了

13.用*标识编辑过的文件

1. Editor->General -> Editor Tabs
2. 在IDEA中，你需要做以下设置，这样被修改的文件会以*号标识出来，你可以及时保存相关的文件。
3. "Mark modifyied tabs with asterisk"

14.**关闭自动代码提示**

1. Preferences => IDE Settings => Editor => Code Completion => Autopopup documentation in (ms)

15.常用快捷键

1. Ø Top #10 切来切去: Ctrl+Tab
2. Ø Top #9 选你所想【选中上下文相关联代码】: Ctrl+W
3. Ø Top #8 代码生成: Template/Postfix +Tab
4. Ø Top #7 发号施令: Ctrl+Shift+A
5. Ø Top #6 无处藏身: Shift+Shift
6. Ø Top #5 自动完成: Ctrl+Shift+Enter
7. Ø Top #4 创造万物: Alt+Insert

使用前三名！

1. Ø Top #1 智能补全: Ctrl+Shift+Space
2. Ø Top #1 自我修复: Alt+Enter
3. Ø Top #1 重构一切: Ctrl+Shift+Alt+T

其他辅助

1. 以上这些神键配上一些辅助快捷键，即可让你的双手90%以上的时间摆脱鼠标，专注于键盘仿佛在进行钢琴表演。这些不起眼却是至关重要的最后一块拼图有：

2. **命令:** `Ctrl+Shift+A`可以查找所有IntelliJ的命令, 并且每个命令后面还有其快捷键。所以它不仅是一大神键, 也是查找学习快捷键的工具。
3. **新建:** `Alt+Insert`可以新建类、方法等任何东西。
4. **格式化代码:** 格式化import列表`Ctrl+Alt+O`, 格式化代码`Ctrl+Alt+L`。
5. **切换窗口:** `Alt+Num`, 常用的有1-项目结构, 3-搜索结果, 4/5-运行调试。`Ctrl+Tab`切换标签页, `Ctrl+E/Ctrl+Shift+E`打开最近打开过的或编辑过的文件。
6. **单元测试:** `Ctrl+Alt+T`创建单元测试用例。
7. **运行:** `Alt+Shift+F10`运行程序, `Shift+F9`启动调试, `Ctrl+F2`停止。
8. **调试:** `F7/F8/F9`分别对应Step into, Step over, Continue。
9. 此外还有些我自定义的, 例如水平分屏`Ctrl+|`等, 和一些神奇的小功能`Ctrl+Shift+V`粘贴 很早以前拷贝过的, `Alt+Shift+Insert`(块选)进入到列模式进行按列选中

16.svn *****不能同步代码问题修正*****

File -> Settings -> Subversion -> General => Use command line client 选中

1. 使用command line方式需要指定svn.exe的路径, 例如:
`D:\tools\TortoiseSVN\bin\svn.exe`
2. 注意, 安装TortoiseSVN时路径中不要带空格, 例如:`C:\Program Files\TortoiseSVN\bin\svn.exe`就会报错。
3. 安装TortoiseSVN选择全部安装组件, 否则可能没有svn.exe

17.设置idea的SVN忽略掉*.iml文件

1. Editor->File Types=>Ignore files and folders增加*.iml;
2. 在Ignore files and folders中输入.idea;注意要";"结尾。你就可以隐藏.idea文件夹了

18.改变编辑文本字体大小

1. File -> settings -> EDITOR COLORS & FONTS -> FONT -> SIZE

*19.IDEA编码设置

1. FILE -> SETTINGS -> FILE ENCODINGS => IDE ENCODING
- 2.
3. FILE -> SETTINGS -> FILE ENCODINGS => Project Encoding
- 4.
5. FILE -> SETTINGS -> FILE ENCODINGS => Default encoding for properties files
- 6.
7. FILE -> SETTINGS -> FILE ENCODINGS => Transparent native-to-ascii conversion

*20.Live Templates

```
System.out.println 快捷输出
"abc".sout => System.out.println("abc");
在eclipse中使用方式: sysout=> System.out.println();
```

```
for循环
List list = new ArrayList();
输入: list.for 即可输出
for(String s:list){``}
```

21.配置tomcat参数

1. `vm options: -Xms256m -Xmx512m -XX:PermSize=128m -XX:MaxPermSize=256m`

22.idea安装插件的方法

1. 以IntelliJ IDEA 14.0.1安装findbugs插件为例:
- 2.
3. (1)在线方式:进入File->setting->plugins->browse repositories 搜索你要下载的插件名称,
4. 右侧可以找到下载地址,完成后按提示重启即可.
- 5.
6. (2)离线安装: 下载findbugs插件地址: <http://plugins.jetbrains.com/plugin/3847>,
7. 将下载的FindBugs-IDEA-0.9.994.zip,安装插件: 进入File->setting->plugins=> `Install plugin from disk...`
8. 定位到到刚才下载的jar,点击ok,完成后按提示重启即可.
9. 插件安装的位置在C:\Users\xxxxxxx\IntelliJ IDEA 14\config\plugins\插件名下.
- 10.
11. 安装iBATIS/MyBatis min-plugin插件

23.调整idea启动时的内存配置参数*

1. `%IDEA_HOME%/bin/idea.exe.vmoptions`

*24.*导入eclipse web项目发布到Tomcat如果找不到

1. 导入eclipse web项目发布到Tomcat如果找不到,可以在环境配置的Facets增加web支持,在Artifacts中增加项目部署模块名

*25.*每次打开一个新jsp或java文件时,cpu都占用很高,去掉检验即可

每次打开一个新jsp或java文件时,cpu都占用很高,去掉检验即可:
file->settings->editor->inspections

*26.*idea增加spring/struts关联文件支持

1. project Settings->Modules->选中项目右键可添加

27. IDEA开启类修改后自动编译

1. File->setting->Buil, Execution, Deployment->compiler=>Make project automatically
2. 编译错误问题解决
3. Error:java: Compilation failed: internal java compiler error
4. set中Java compiler 设置的问题, 项目中有人用jdk1.6 有人用jdk1.7 版本不一样 会一起这个错误

*28.*提示实现Serializable接口

1. 使用 Eclipse 或 MyEclipse 的同学可能知道, 如果 implements Serializable 接口时, 会提示你生成 serialVersionUID。
2. 但 IntelliJ IDEA 默认没启用这个功能。
3. Preferences->IEditor->nspections->Serialization issues->Serializable class without 'serialVersionUID',
4. 选中以上后, 在你的class中: 光标定位在类名前, 按 Alt+Enter 就会提示自动创建 serialVersionUID了

*29.*演出模式

我们可以使用【Presentation Mode】, 将IDEA弄到最大, 可以让你只关注一个类里面的代码, 进行毫无干扰的coding。

可以使用Alt+V快捷键, 弹出View视图, 然后选择Enter Presentation Mode。效果如下:

```

99 // 跳转回调redirect_uri
100 String REDIRECT_URI = SysConstant.HOST_SERVICE_CLOUDPAYMENT + "code/cashDesk.do";
101 // 自定义回调参数
102 StringBuilder STATE = new StringBuilder();
103 STATE.append(schoolCode).append("@").append(orderNO);
104 if (userAgent != null && userAgent.contains("AlipayClient")) {
105     // 支付宝
106     redirectAttr.addAttribute(ControllerConstant.USER_AGENT_TYPE, ControllerConstant.USER_AGENT_TYPE);
107     redirectAttr.addAttribute(ControllerConstant.REDIRECT_URI, REDIRECT_URI);
108     redirectAttr.addAttribute(ControllerConstant.STATE, STATE.toString());
109     redirectAttr.addAttribute(ControllerConstant.HOST_CUSTOMER_CODE, schoolCode);
110     return "redirect:/third/auth.do";
111 } else if (userAgent != null && userAgent.contains("MicroMessenger")) {
112     // 微信
113     redirectAttr.addAttribute(ControllerConstant.USER_AGENT_TYPE, ControllerConstant.USER_AGENT_TYPE);
114     redirectAttr.addAttribute(ControllerConstant.REDIRECT_URI, REDIRECT_URI);
115     redirectAttr.addAttribute(ControllerConstant.STATE, STATE.toString());
116     redirectAttr.addAttribute(ControllerConstant.HOST_CUSTOMER_CODE, schoolCode);
117     return "redirect:/third/auth.do";
118 } else {
119     redirectAttr.addAttribute(ControllerConstant.STATE, STATE.toString());
120     return "redirect:/code/cashDesk.do";
121 }
122
123 }

```

<https://blog.csdn.net/y10905>

这个模式的好处就是，可以让你更加专注，因为你只能看到特定某个类的代码。可能读者会问，进入这个模式后，我想看其他类的代码怎么办？这个时候，就要考验你快捷键的熟练程度了。你可以使用CTRL+E弹出最近使用的文件。又或者使用CTRL+N和CTRL+SHIFT+N定位文件。

如何退出这个模式呢？很简单，使用ALT+V弹出view视图，然后选择Exit Presentation Mode 即可。

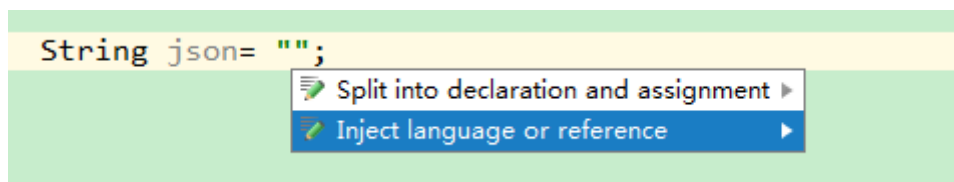
但是我强烈建议你不要这么做，因为你是可以在Enter Presentation Mode模式下在IDEA里面做任何事情的。当然前提是，你对IDEA足够熟练。

*30.*神奇的Inject language

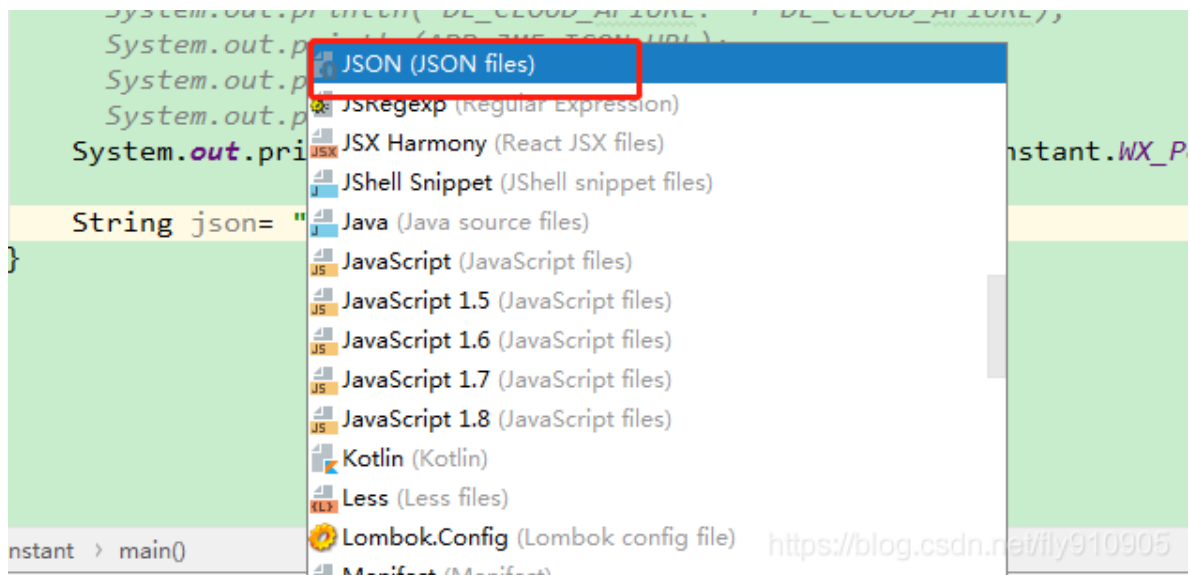
如果你使用IDEA在编写JSON字符串的时候，然后要一个一个去转义双引号的话，就实在太不应该了，又烦又容易出错。

在IDEA可以使用Inject language帮我们自动转义双引号。

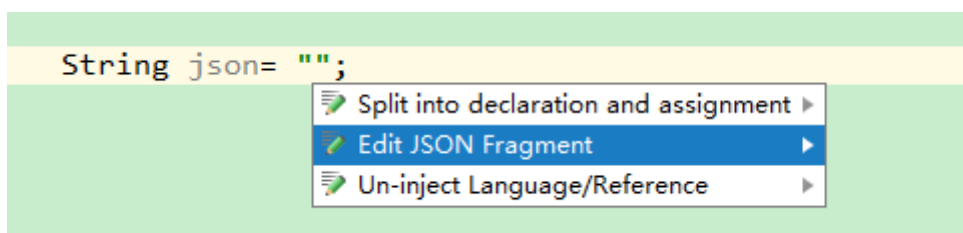
- 先将焦点定位到双引号里面，使用alt+enter快捷键弹出inject language视图，并选中Inject language or reference。



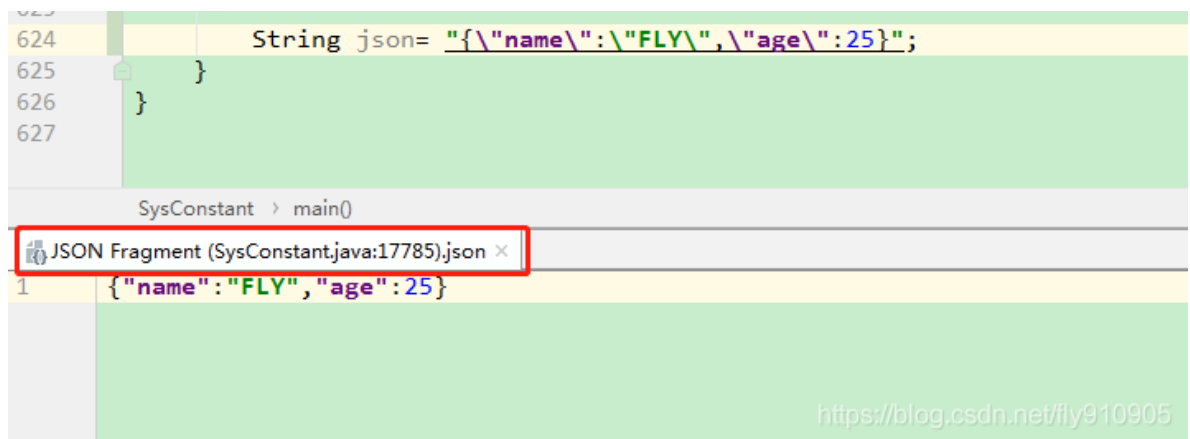
- 选择后,切记，要直接按下enter回车键，才能弹出inject language列表。在列表中选择 json组件。



- 选择完后。鼠标焦点会自动会定位在双引号里面，这个时候你再次使用alt+enter就可以看到



- 选中Edit JSON Fragment并回车，就可以看到编辑JSON文件的视图了。

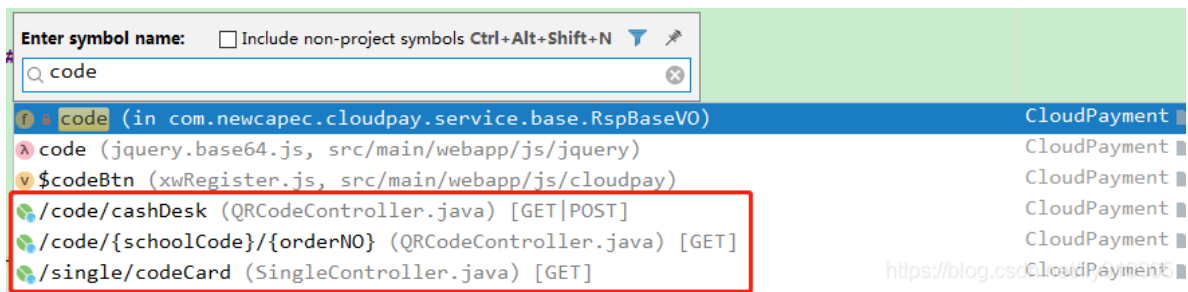


- 可以看到IDEA确实帮我们自动转义双引号了。如果要退出编辑JSON信息的视图，只需要使用ctrl+F4快捷键即可。
- Inject language可以支持的语言和操作多到你难以想象，读者可以自行研究。

*31. *强大的symbol

如果你依稀记得某个方法名字几个字母，想在IDEA里面找出来，可以怎么做呢？

直接使用ctrl+shift+alt+n，使用symbol来查找即可。



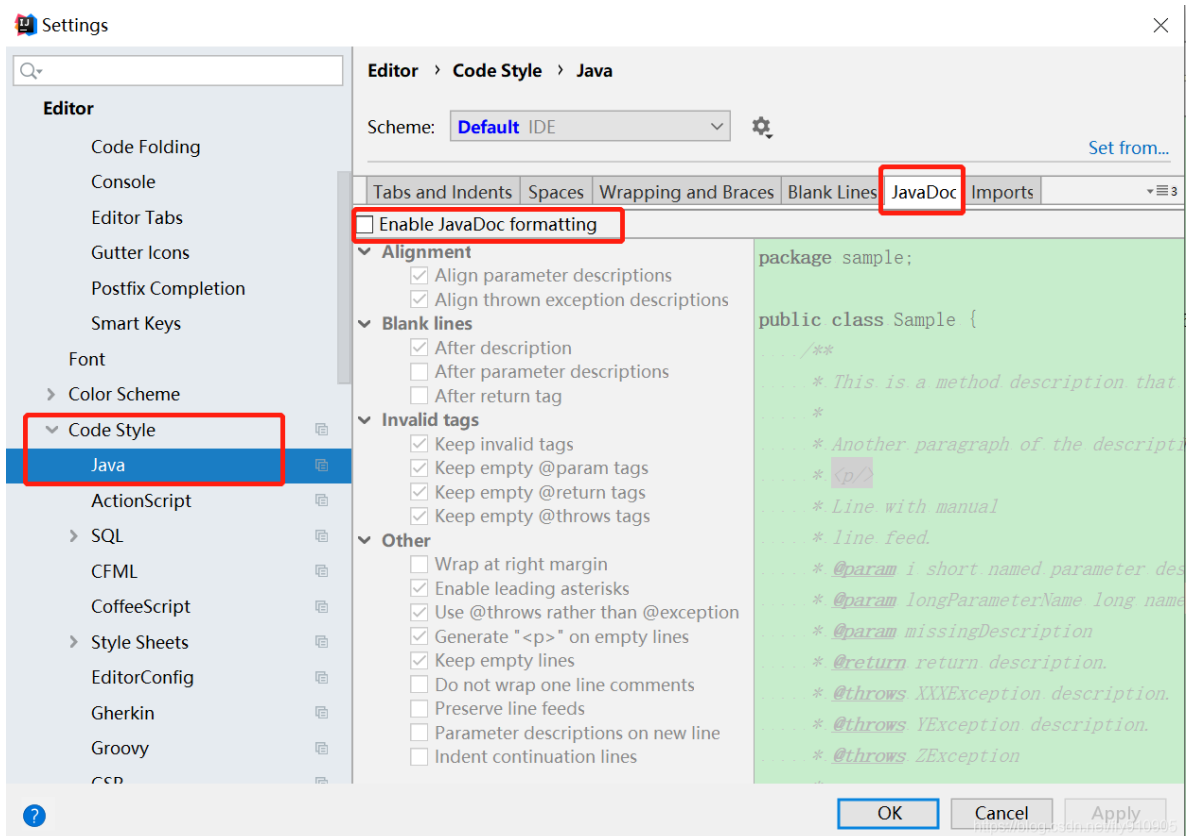
*32.idea 快捷键和Windows默认快捷键冲突解决 (如: *Ctrl+Alt+↑或Ctrl+Alt+F12*) *

解决方案: 在桌面右键 - 图形选项 - 快捷键 - 禁止 就可以了

*33.*格式化代码时，注释被格式化问题

解决方案

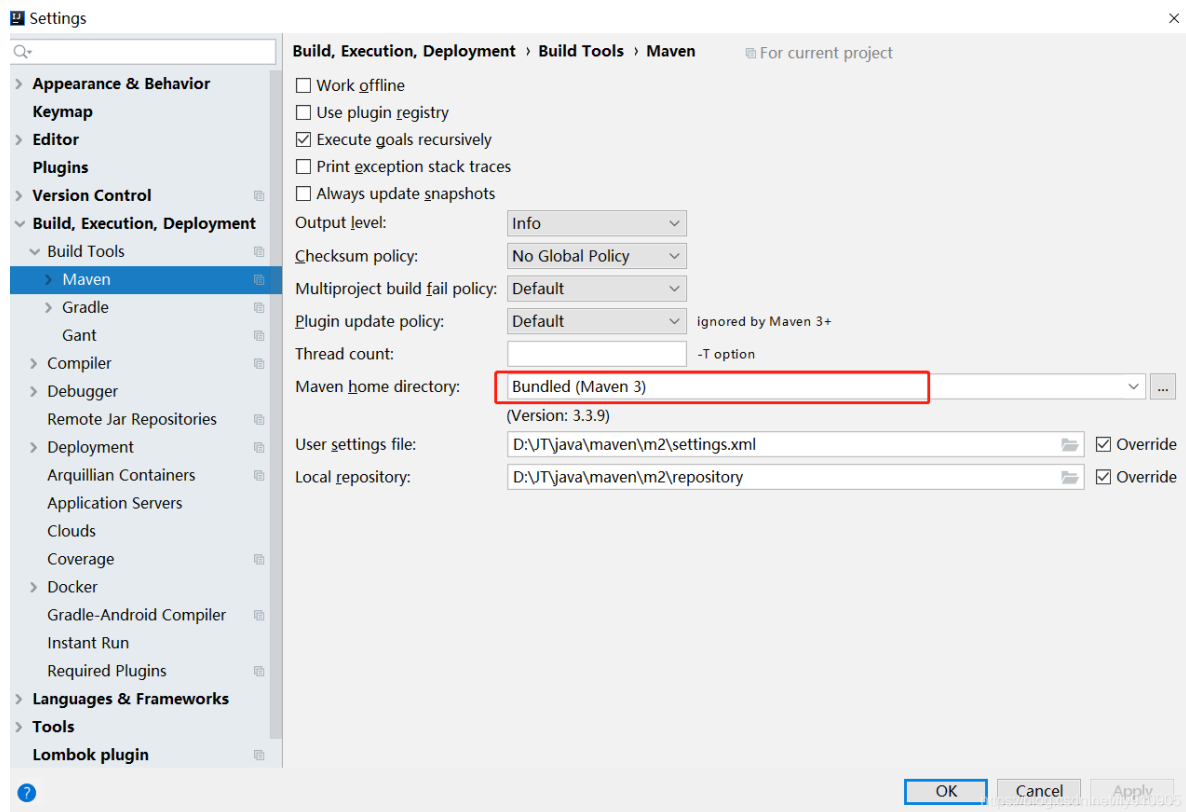
将enable javadoc formatting取消掉



*34. import导入报错，更新maven提示 *Unable to import maven project: See logs for details

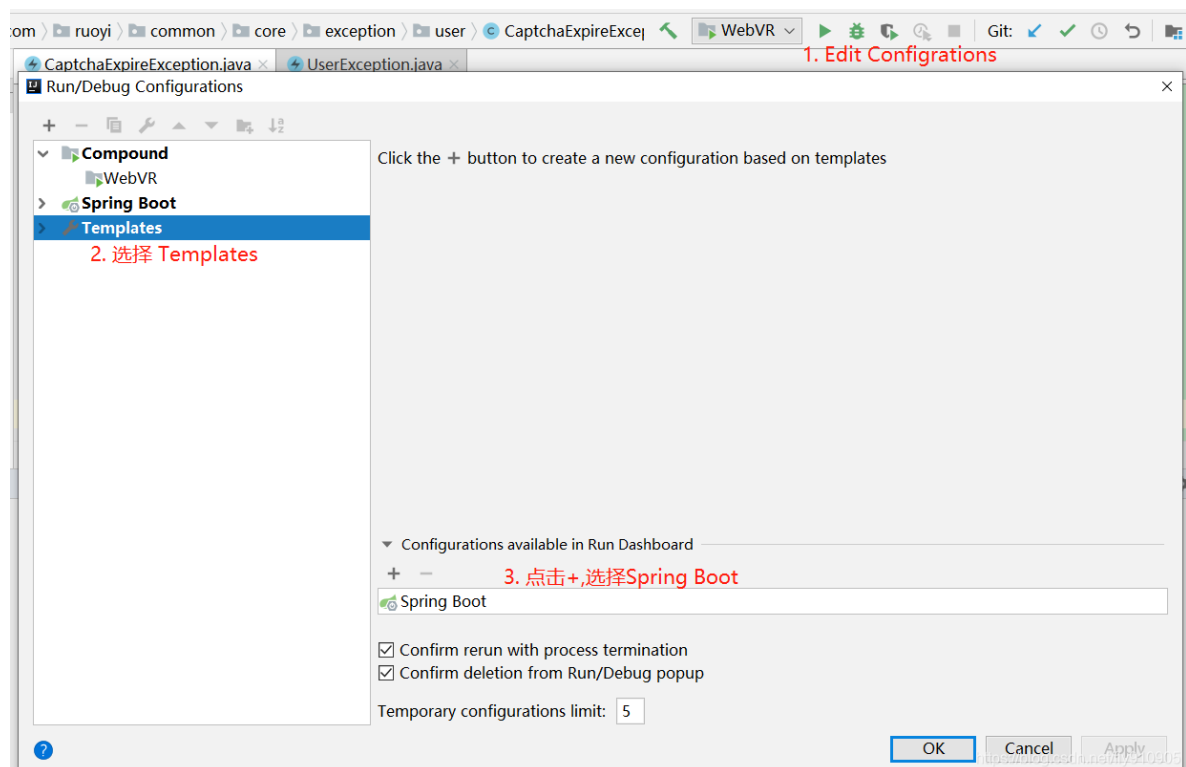
IDEA2019和Maven3.6.2不兼容导致的，需要把Maven降级到3.6.1版本

或者使用idea自带的maven组件



35. IDEA2019版显示出Run Dashboard

1. 启动按钮，点击“Edit Configurations”
2. 左侧，选择Templates
3. 右侧，选择Configurations available in Run Dashboard
4. 点击“+”，选择Spring Boot



36. 切换大小写

- idea的切换大小写的默认快捷键是ctrl+shift+u

- 如果默认快捷键冲突，可以双击shift，输入Toggle Case