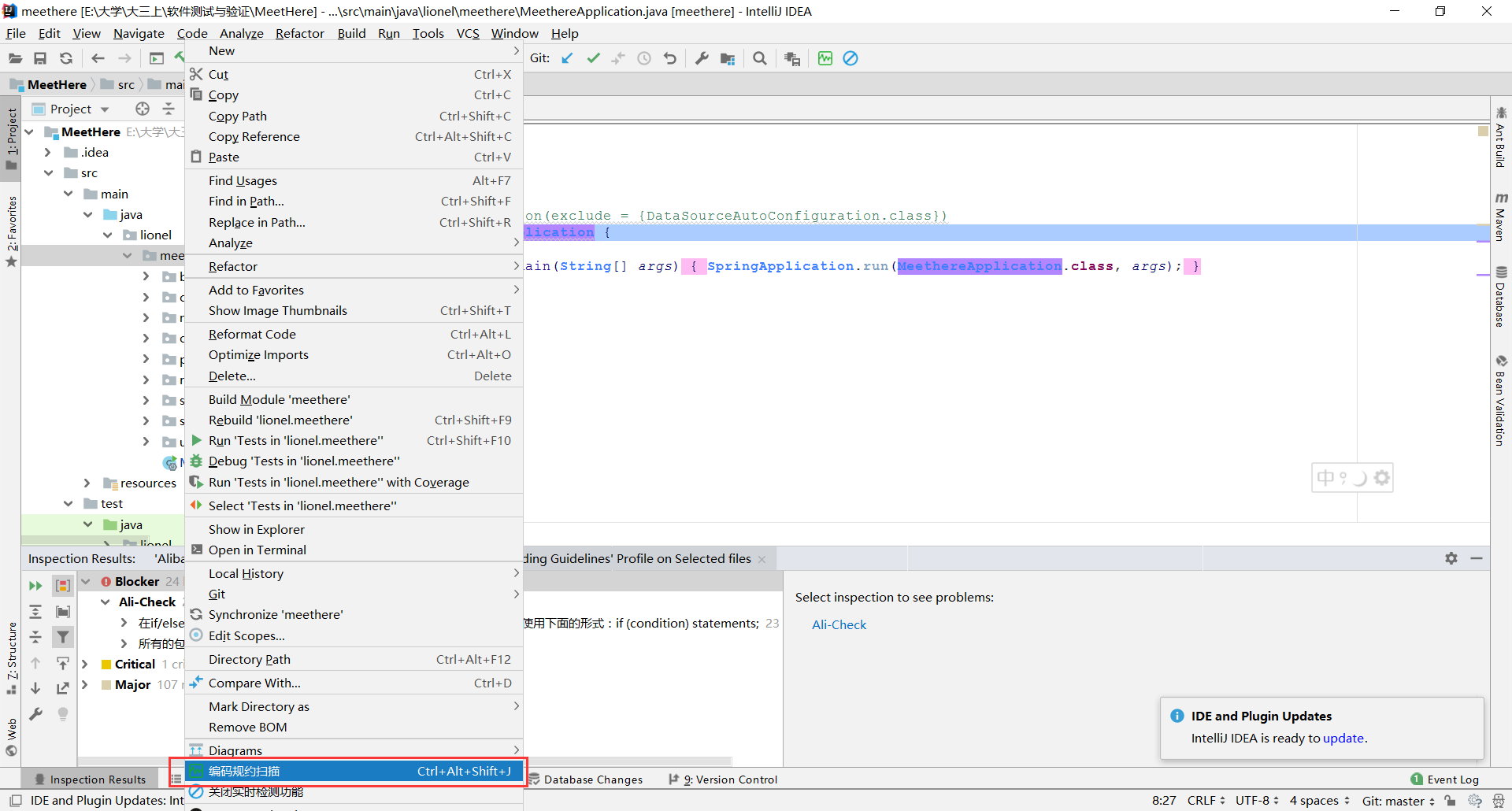
**代码静态分析**

卢宇博

1. **使用工具**
2. 阿里p3c插件，也就是阿里java规范插件，可以帮助我们检查代码中潜在的错误，使代码更加规范。我们在本次MeetHere项目的开发过程中在idea中使用了阿里p3c插件。



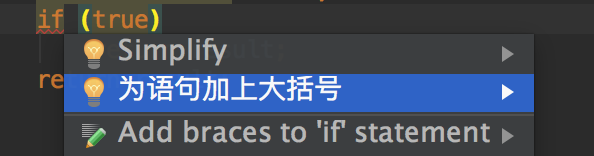
1. 在项目目录上右键点击出现这两个功能按钮，点击绿色的按钮即可开始扫描代码,或者在工程目录上右键也会出现检测的功能按钮。
2. 前面所说的方式是手动检测，插件也提供了实时监测的功能，此功能默认开启，一旦开启在编写代码时就会实时监测，一旦有不符合规范的语句就会出现错误提示。
3. **分析结果**

扫描代码后，不符合规约的代码会按Blocker/Critical/Major三个等级显示在下方面板中，**左边是扫描出的不符合规范的代码，依次点进去可以看到是代码的多少行出现了规约问题以及哪一个规约问题，右边则是规约的详细描述及实例代码。**



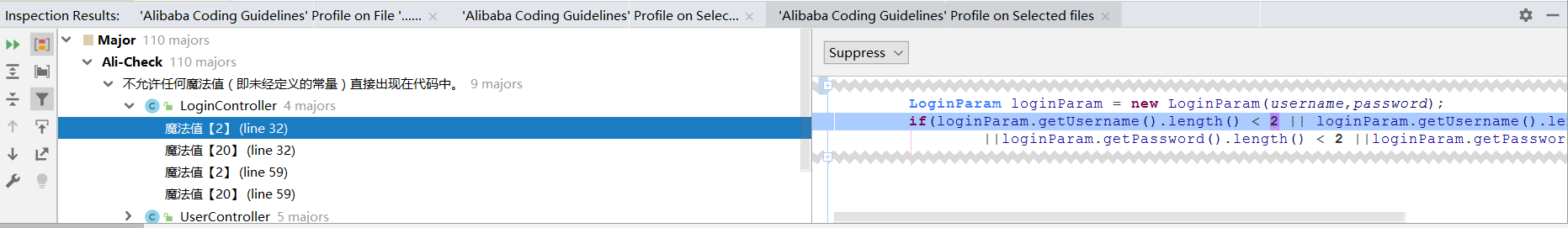
我们在扫描了代码之后，显示出现了如下的问题：

1. **大括号不规范，if下方有错误红线的提示。**

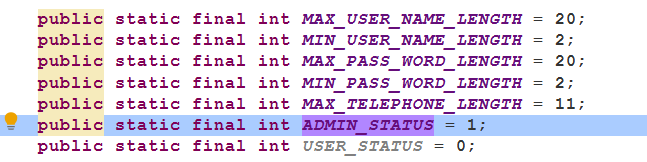


idea中有quick fix快捷键:option/alt+enter,点击后则可以自动添加大括号。

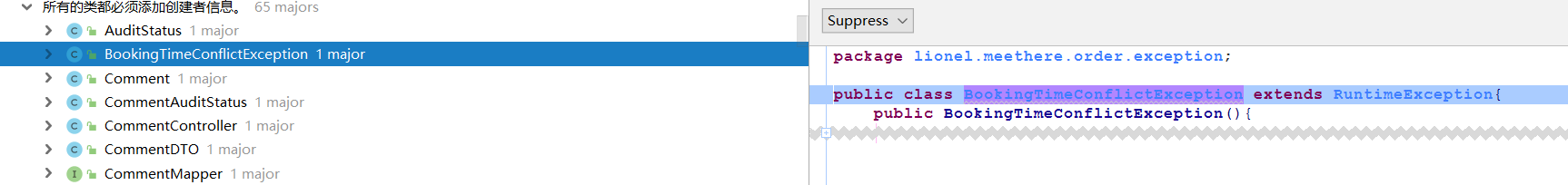
1. **不允许任何魔法值**



我们可以看出在LoginController中因为我们进行了对用户名和密码的参数检验所以使用了未经定义的常量“2”和“10”，这会影响到我们代码的可读性。为此我们引入静态常量,并且常量名全部大写并以下划线分隔。

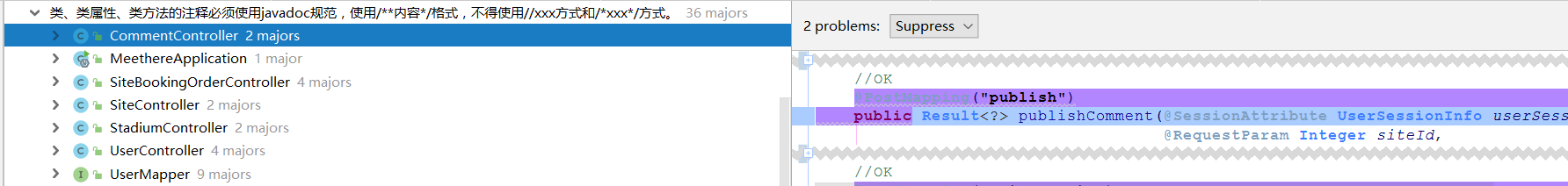


1. **所有的类都必须添加创建者信息**



根据插件提供的信息我们得知，在我们创建的所有类中，都需要加入@author注释来表明该类的开发人员，方便测试人员与其对接。

1. **类、类属性、类方法的注释必须使用javadoc规范，使用/\*\*内容\*/格式**



根据提示，我们需要将“//内容”的代码注释风格全部改为“/\*\*内容\*/”。

1. **使用心得**

p3c插件的使用非常方便，有的错误右侧还有quick fix的按钮，点击后直接可以改正代码，但是并不是所有的问题都有此按钮，有些问题还是需要手动修改。以下是在使用p3c的过程中总结的插件优点：

1. 中文提示，且按照《阿里巴巴Java开发手册》给出详细的提示和解释，看起来真的很舒服。
2. 能够检测出一些平时不在意的问题，可以提高开发编码过程中对于细枝末节的注意。
3. quick fix，检测出问题后点击一下按钮即可完成代码的修改。
4. 基本满足代码规范检测的需求。