

---

## Java 断言

### 1、断言的概念

- 1) `assert` 是 JDK1.4 之后引入的，属于 Java 关键字，表示断言。一般断言 `assert` 用于代码的单元测试中，编译器默认不开启断言。
- 2) 断言失败后，JVM 会抛出一个 `AssertionError` 错误，它继承自 `Error`，注意，这是一个错误，是不可恢复的，也就表示这是一个严重问题，开发者必须予以关注并解决之。
- 3) 在代码的线上运行环境中，是不会启用断言的，因此尽量避免断言代替 `if` 的逻辑。

### 2、断言的用法

- 语法格式：有两种

`assert Expression1`

`assert Expression1:Expression2`

说明：`Expression1` 应该总是一个布尔值，`Expression2` 是断言失败时输出的失败消息的字符串。如果 `Expression1` 为假，则抛出一个 `AssertionError`，这是一个错误。

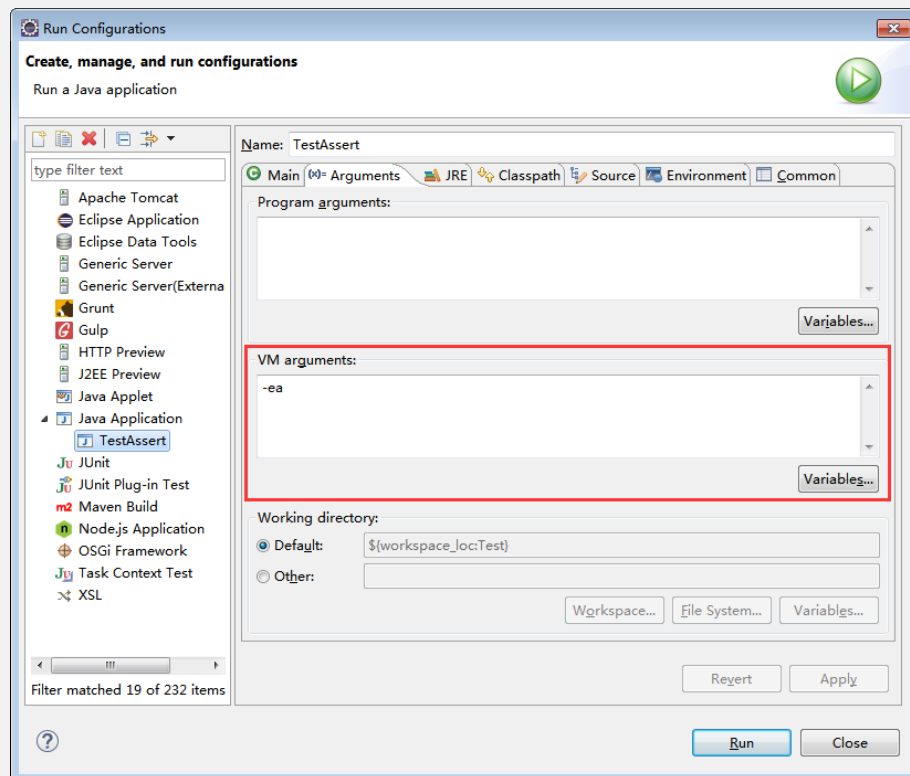
- 在使用之前应该先开启断言，Eclipse 中开启断言的方法：

右键项目——Run As——Run Configurations

选择在 Arguments 选项卡中的 VM arguments：

增加参数 `-ea`，保存，关闭窗口；这时就代表开启了断言；

如果参数是 `-da`，代表的是禁止断言。开始断言见以下图示：



# 1) 第一种使用方法示例：

```
public class Demo1 {
    public static void main(String[] args) {
        boolean isOk = (1 > 2);
        assert isOk;
        System.out.println("程序正常");
    }
}
```

```
Exception in thread "main" java.lang.AssertionError
    at Demo1.main(Demo1.java:5)
```

因为  $1 > 2$  显然是错误的，所以执行结果抛出异常：Exception in thread "main" java.lang.AssertionError。

如果把  $1 > 2$  改为  $1 < 2$  则程序能顺利执行，打印 “程序正常”

# 2) 第二种使用方法示例：

assert 后面跟个冒号表达式。如果冒号前为 true，则冒号后面的被忽略。如果冒号前为 false，则抛出 AssertionError，错误

信息内容为冒号后面的内容。以下是示例代码：

```
public class Demo2 {  
    public static void main(String[] args) {  
        boolean isOk = 1 > 2;  
        try {  
            assert isOk : "程序错误";  
            System.out.println("程序正常");  
        } catch (AssertionError err) {  
            System.out.println(err.getMessage());  
        }  
    }  
}
```

输出结果为：程序错误。从而可以根据提示，进行代码的修改。

#### 附 录：断言开启的方法

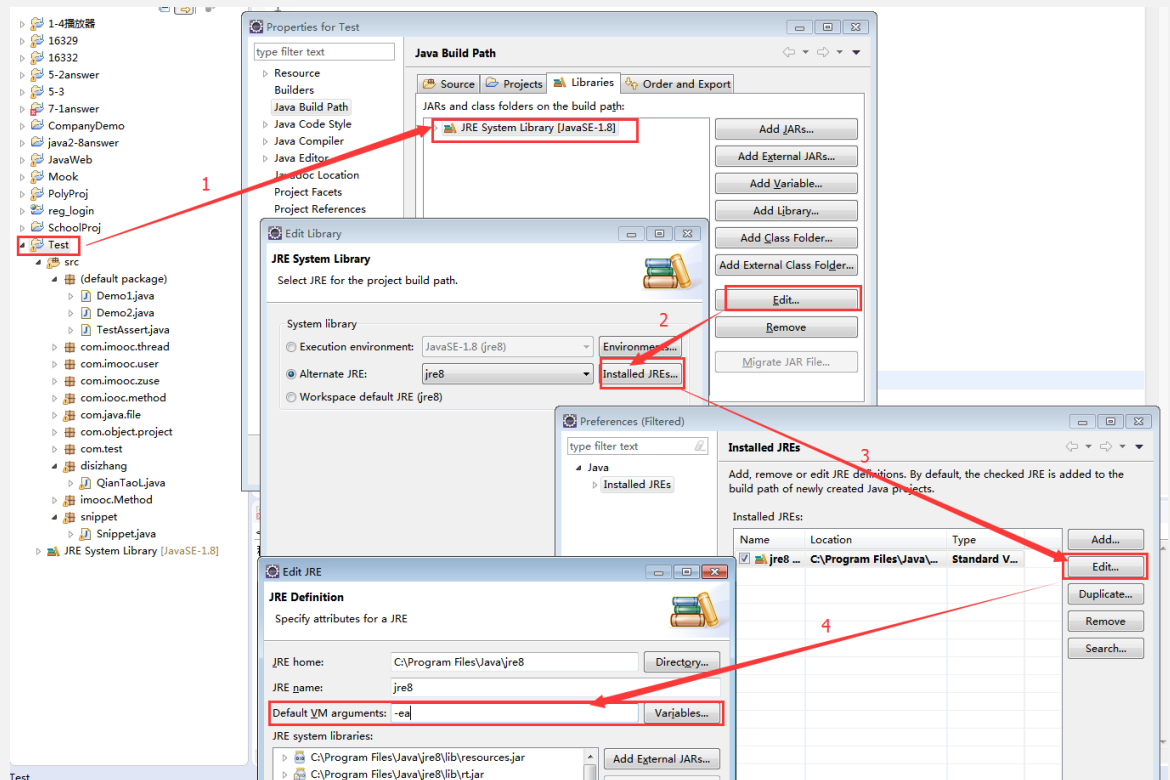
- Eclipse 开启断言

##### Eclipse 为 Tomcat 打开断言的方式：

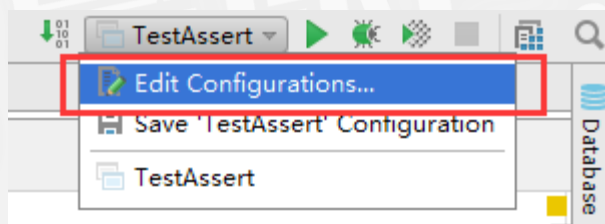
Run > Run Configurations > 左侧 Apache tomcat 选中在用的 Tomcat 服务器 > 右侧 Arguments > VM arguments 添加 -ea 需要加个空格与其他项分隔。

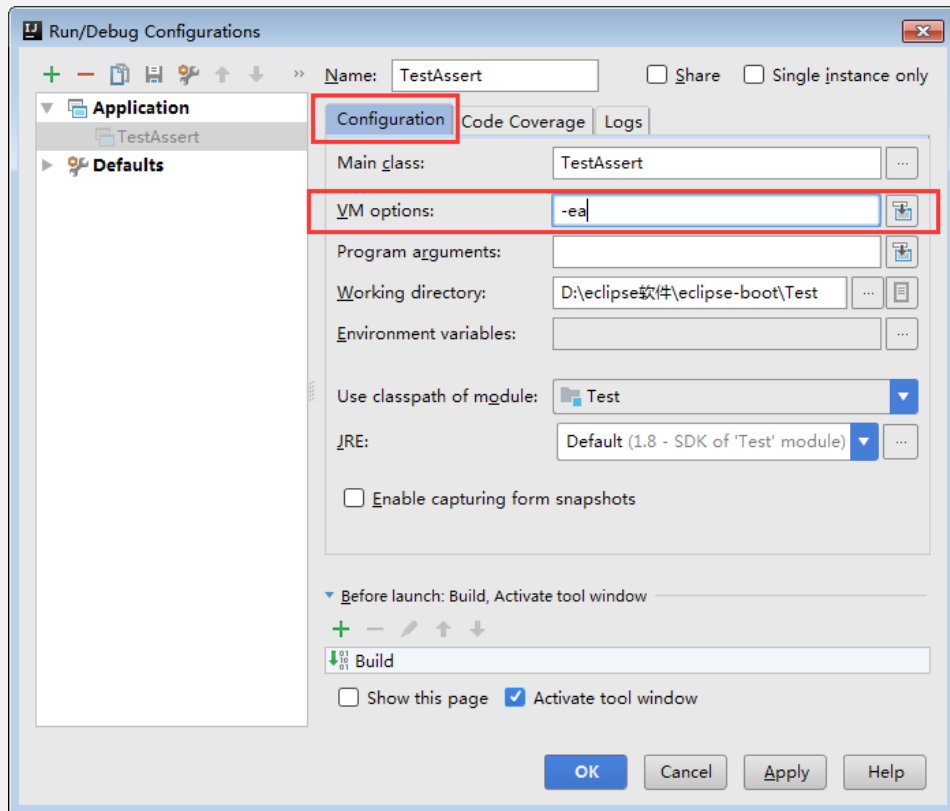
##### 整个项目开启断言：

以 Test 类为例：



- IDEA 开启断言：





同样，如果是配置 Tomcat，也是同样的做法：

