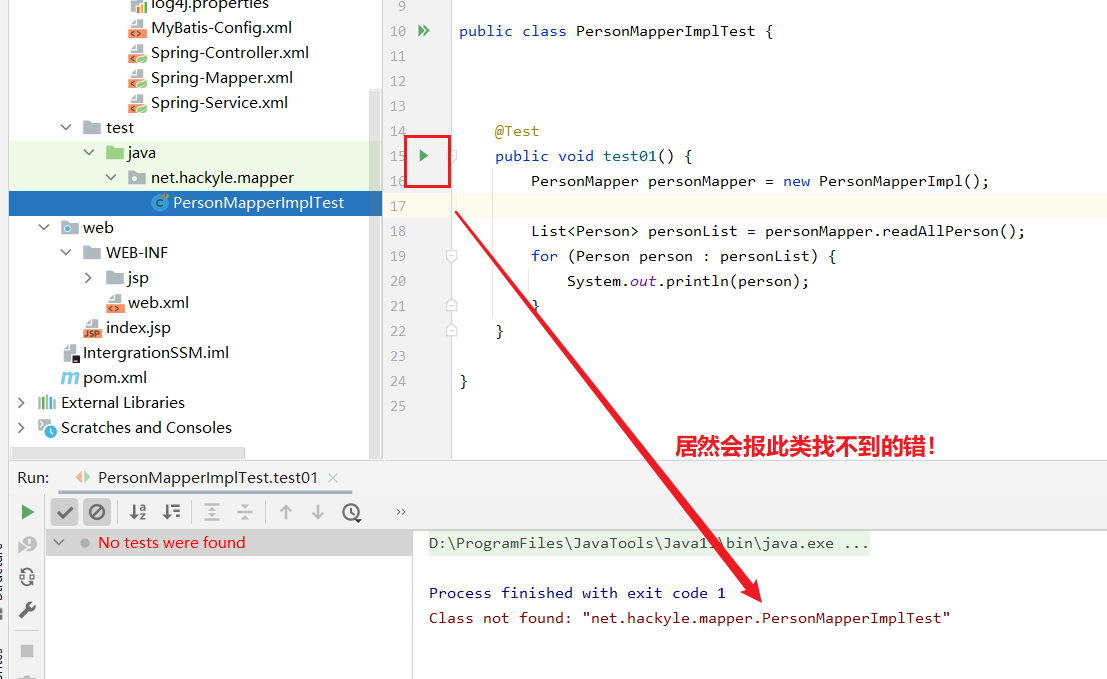
# 单元测试

在某些情况下不能使用@Test，但是依赖已经导入成功了的。这时就需要把依赖中的范围该为compile即可。

**maven中运行junit4测试类报错：class not found**

**现象：**

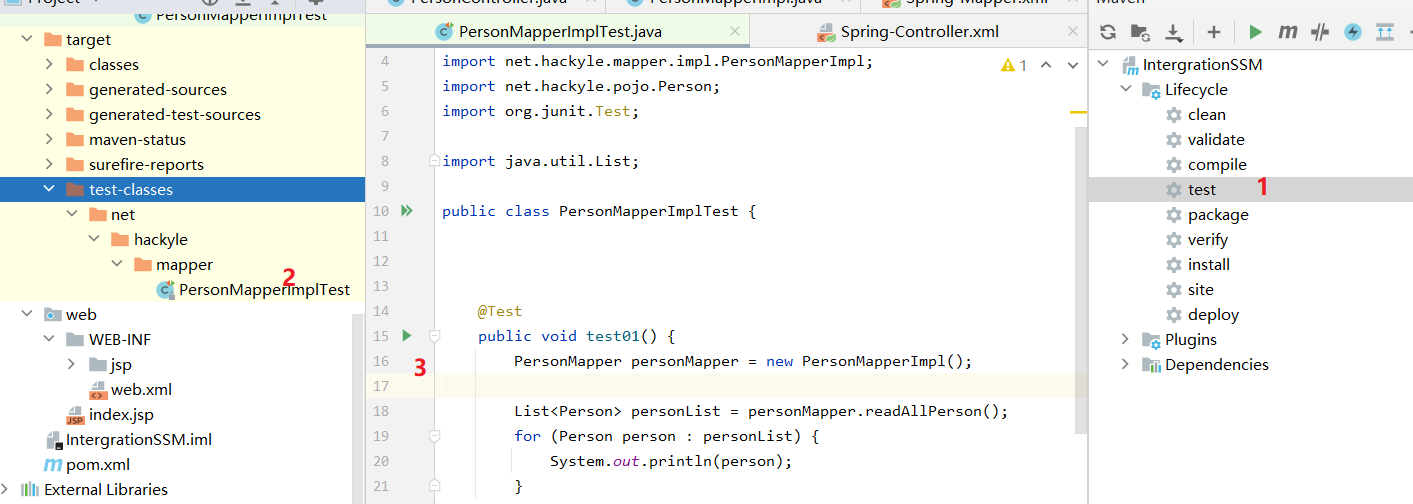


**原因：**

1. 明明编写了类，为什么会出现class not found错误呢，肯定是没有编译成功。
2. 在maven项目里要运行Junit，必须在 target/test-classes 找到对应的编译类。

**解决方案：**

1. 执行“Maven-Test”，就可以在target/test-classes能找到编译的测试类



1. 然后再运行刚刚的测试方法，就可以了

# mapper层

**报错现象：**

Property 'dataSource' threw exception; nested exception is java.lang.NoClassDefFoundError: org/springframework/jdbc/datasource/TransactionAwareDataSourceProxy

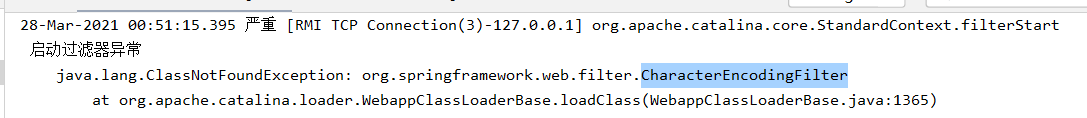
**问题原因：**

* 缺少Spring-jdbc依赖
* org.mybatis.spring.SqlSessionFactoryBean用到了spring-jdbc。

**解决：**导入该依赖。

过滤器报错

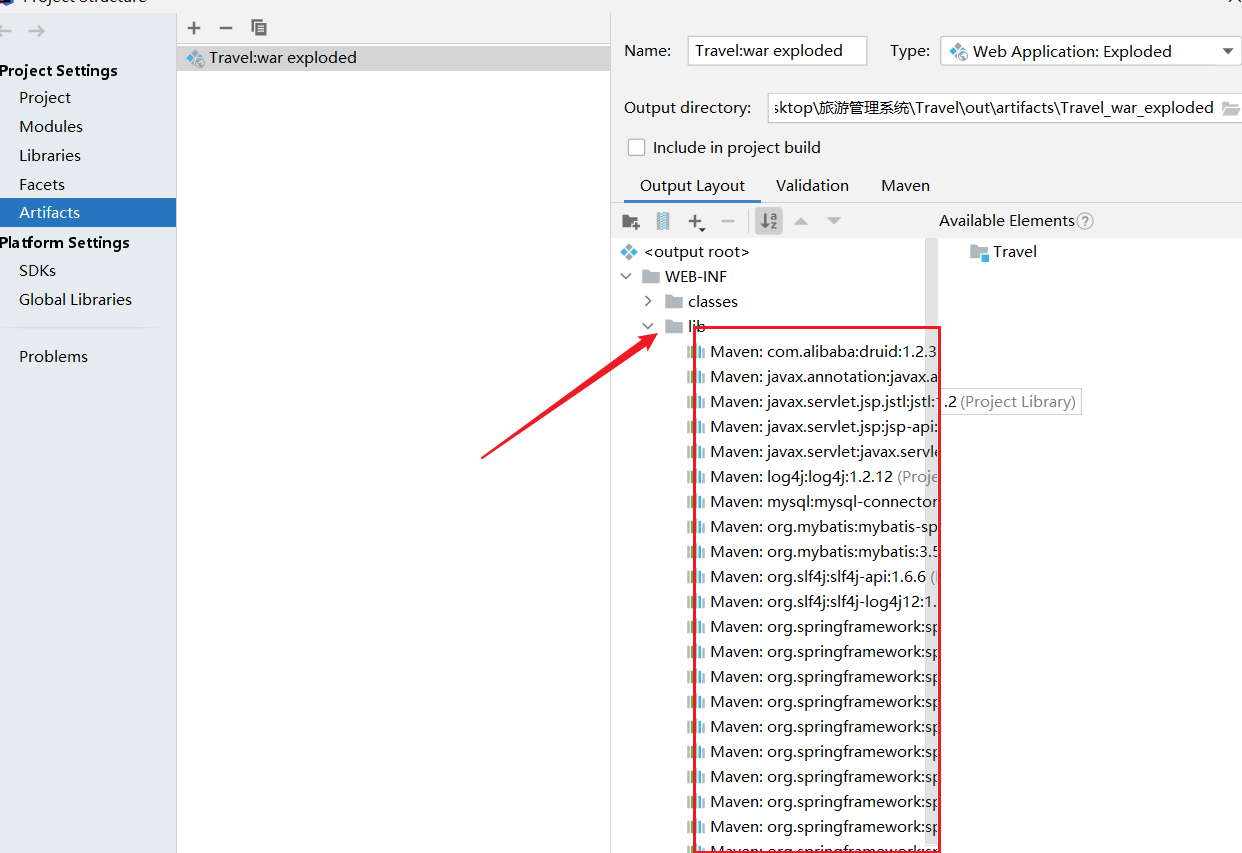
现象：



分析：过滤器是服务器一开始启动就要执行的组件，一开始就报错，说明该组件根本没有，所以检查lib包是否成功部署到服务器。

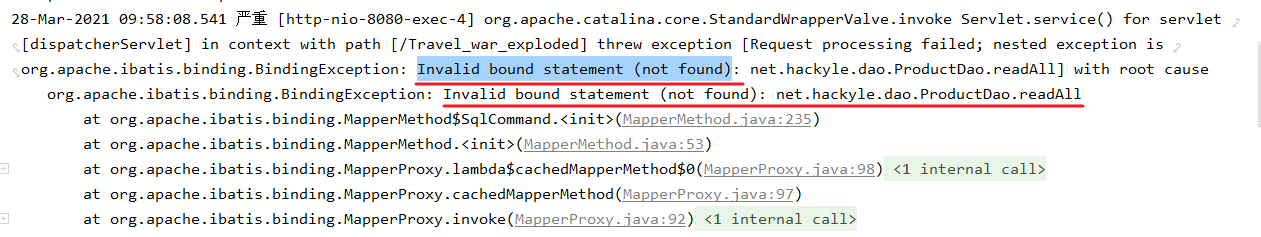
原因：部署到Tomcat时没有带上lib包

解决：



## Invalid bound statement(not found)

**问题现象：**



**解决方向一：检查Mapper接口和Mapper的XML文件中是否关联成功：**

1. 接口的**方法名**必须等于相应XML配置文件中的**id名**
2. 接口的方法**返回值**必须与XML文件中的返回**类型一致**
3. 接口方法的**入参类型**与XML中的入参类型一致
4. XML中的**命名空间**必须绑定此接口

**检查Spring-Mapper.xml是否导入了MyBatis主配置文件、映射文件：**



**检查MyBatis-Config.xml是否导入了映射XML文件：**

