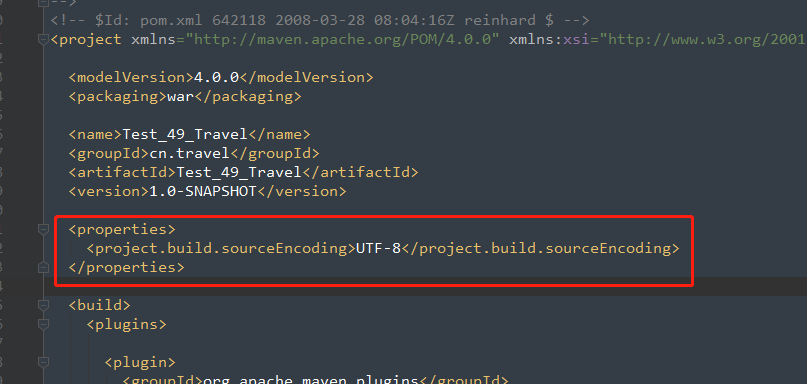
# 启动项目

## 报错：'build.plugins.plugin.version' for org.apache.maven.plugins:maven-compiler-plugin is missing

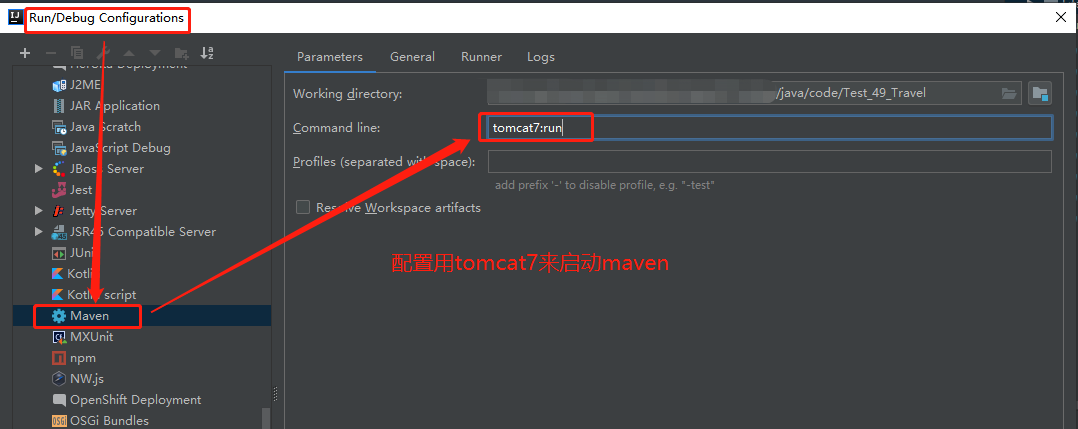


## 报错：[WARNING] Using platform encoding (UTF-8 actually) to copy filtered resources, i.e. build is platform



<properties>  
 <project.build.sourceEncoding>  
 UTF-8  
 </project.build.sourceEncoding>  
</properties>

## 配置使用tomcat7启动服务器

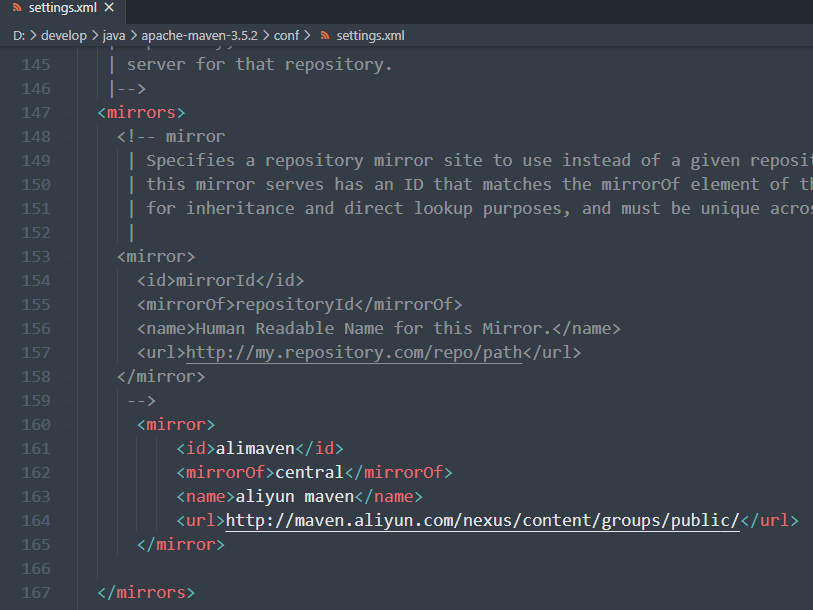


## 配置使用Maven阿里镜像

maven中央仓库下载东西贼慢。

找到Maven安装目录，/conf/setting.xml，记事本打开，搜索mirrors，

往里面加入一个mirror标签，里面写上下面几句。



<mirror>

          <id>alimaven</id>

          <mirrorOf>central</mirrorOf>

          <name>aliyun maven</name>

          <url>http://maven.aliyun.com/nexus/content/groups/public/</url>

  </mirror>

## 5、报错：java.lang.ClassNotFoundException:org.springframework.web.context.ContextLoaderListener

一般来讲是由于pom.xml中下载的jar包未被部署。

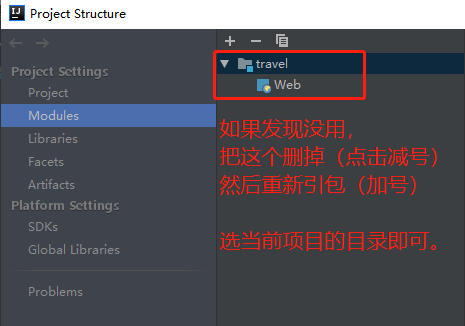
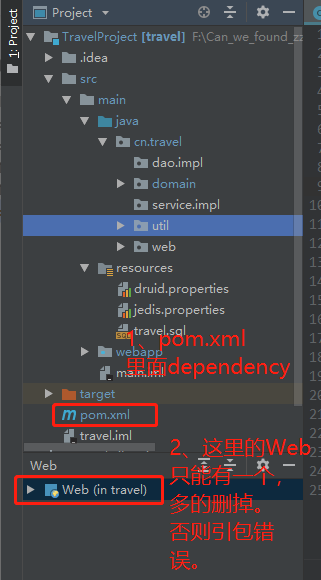
不知道怎么解决。

后面是重新导入文件解决的。

6、创建的Servlet带有Package名称，需要去掉

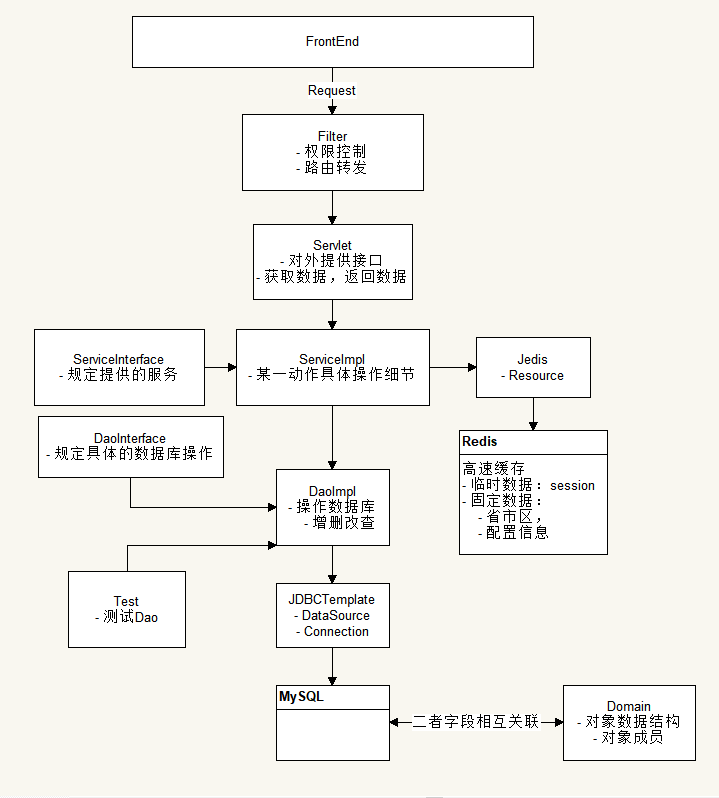


解决方法：



# 二、开发细节

## 1、工作流



Tips:

如果逻辑清晰，可以从DAO开始写，否则可以从Servlet开始写。

## 2、发送邮件

发送邮件的代码不需要掌握，只要会抄，会配置就行。

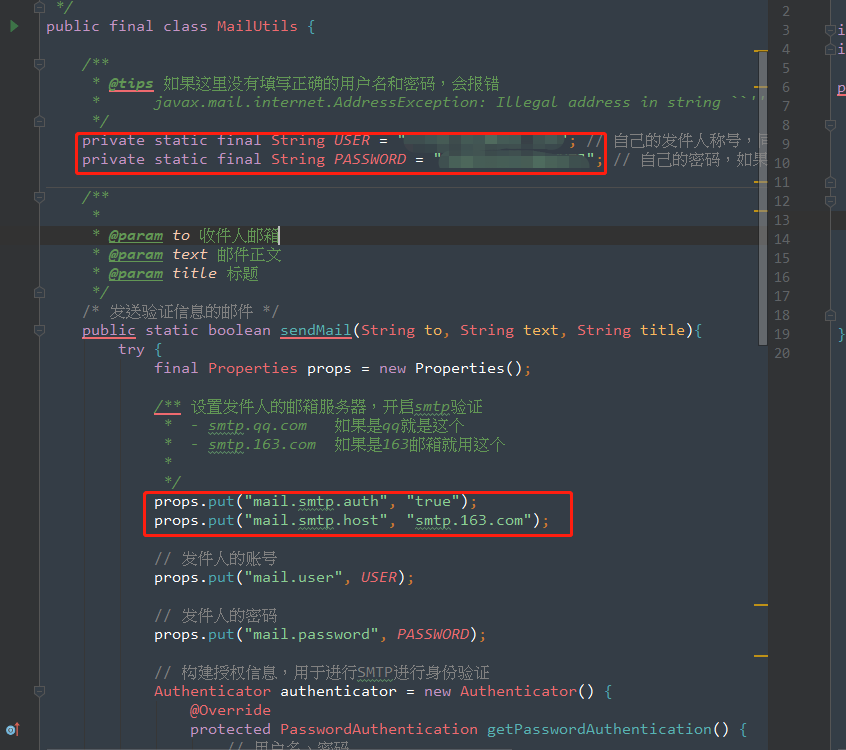
有几个配置需要注意。

发件人邮箱账号，

发件人的授权码，需要去邮箱里面获取授权码， 填进去。

发件开启smtp验证，设置smtp服务器的地址。

props.put("mail.smtp.auth", "true");  
props.put("mail.smtp.host", "smtp.163.com");



## 3、如何控制header.html中显示登陆信息？

这里用的是在header.html单独写一个js，来请求当前用户的信息。

→ 这其实不是很好的实现方式。

## 4、Idea控制台中文乱码解决：

-Dfile.encoding=gb2312

在类中，快速导入接口中的方法：ctrl+l

## 5、抽取一层BaseServlet（核心思想）

主要是，一个接口一个servlet，最后会导致servlet非常多，不要维护，

可以将同类型的servlet合并到同一个类中，原来的多个servlet现在作为同一个servlet的不同方法。

此时，创建一个BaseServlet作为其他servlet的父类，

BaseServlet继承HttpServlet，并重写service方法，当请求进来时，会默认执行service方法内部的函数。

所以可以通过解析请求的url，来分发到具体的servlet中的具体的方法中。

所以为什么需要BaseServlet：提高可维护性，分类聚合，请求分发。

## 6、使用redis对一些很少变化的数据进行缓存。

主要改动在**service**，service中，使用jedis，获取category集合，

如果集合为空，则访问数据库，并将数据库的内容存入redis。

如果集合不为空，直接将redis中获取到的数据提供给servlet。

## 7、注意方法的重载，在interface上需要补齐形参

定义interface时，形参也需要指定，

如果interface的method里面有int a，而implements里面却没有int a，

这说明这个interface的method并没有被实现。

也就是说，public int a(){} 和 public int a(int b)是两个函数，而不是一个函数。

8、Spring JDBC

JdbcTemplate template = new JdbcTemplate(datasource);

template有几种方法。

|  |
| --- |
| - **template.update**(  sql,  param  ); // 没有返回值 |
| - **template.queryForMap**(sql,param); // 返回的是一个Map |
| - **template.queryForList**(sql,param); // 返回的是一个List |
| - **template.query**(  sql,  new BeanPropertyRowMapper<User>(User.class),  参数  ); // 本方法返回的结果是 List<User> |
| - **template.queryForObject**(  sql,  new BeanPropertyRowMapper<User>(User.class),  参数  ); // 本方法返回的是一个User对象 |