# Web开发

## json 接口开发

* 1. Spring开发步骤

1. 添加 jackjson 等相关jar包
2. 配置spring controller扫描
3. 配置spring controller扫描
   1. SpringBoot开发步骤
4. 类添加注解 @RestController



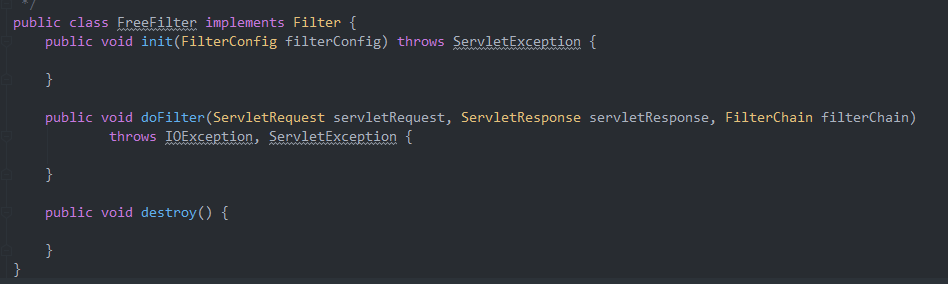
1. 需要使用页面开发只要使用 @Controller

## 自定义Filter

* 1. 使用步骤

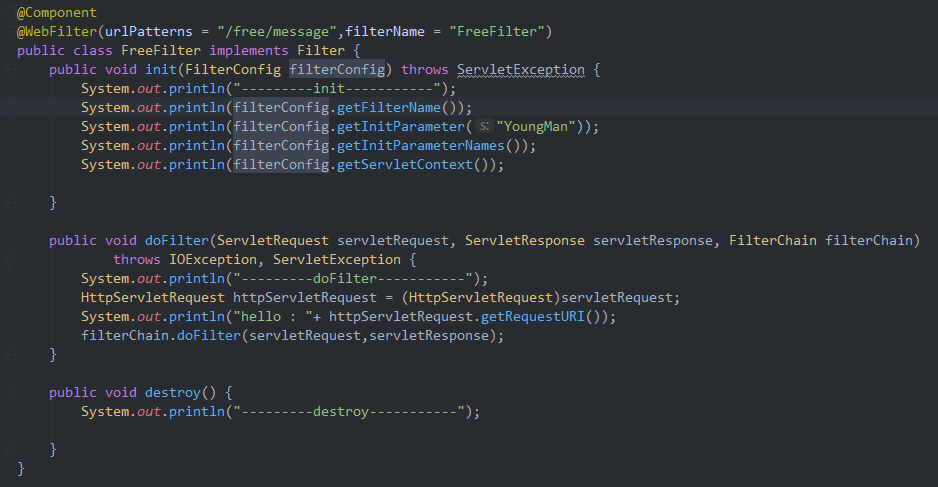
1. 编写自己的过滤器，实现Filter接口
2. 配置过滤器，可以设置过滤器顺序

2.1 编写过滤器



* 1. 配置过滤器

1. 注解@Component @WebFilter配置



(经测试，注解@WebFilter无法生效)

1. 配置注解@Configuration,利用Bean配置

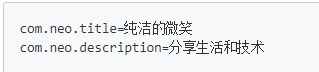


registration.setOrder(1);该方法可以设置过滤的顺序

## 自定义Property

* 1. 使用步骤

1. 将参数配置在application.properties中

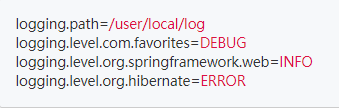


1. 使用注解@Value进行引用



## log配置

* 1. 配置自带的logback日志



1. Path为本地的log地址
2. logging.level是根据包路径的不同配置不同级别的log

# SpringData JPA

## 简介

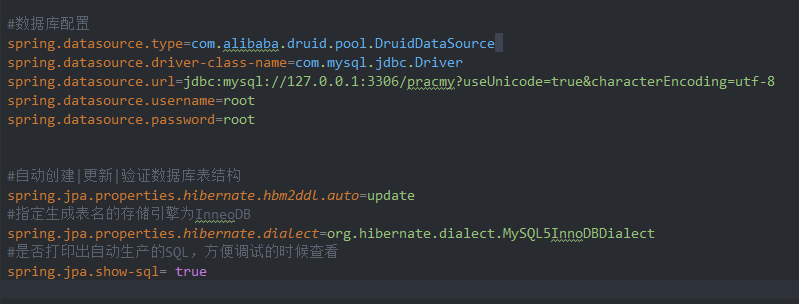
1. Spring Data是Spring下的一个子项目，用于简化数据库访问，并支持云服务的开源框架。Spring Data支持NoSQL和 关系数据存储，其主要目标是使得数据库的访问变得方便快捷。并支持map-reduce框架和云计算数据服务。对于拥有海量数据的项目，可以用Spring Data来简化项目的开发。 然而针对不同的数据储存访问使用相对的类库来操作访问。
2. 相同的ORM框架有Mybatis（增强MybatisPlus）,Hibernate（集成到了JPA）

## JPA的基本使用

* 1. 导入依赖包



* 1. 配置参数

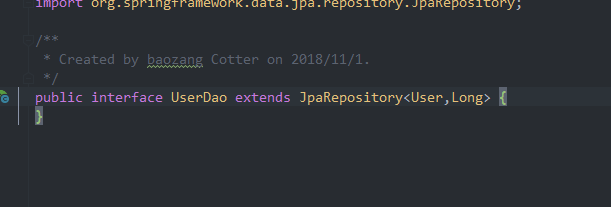


hibernate.hbm2ddl.auto参数四个值：

1. create： **每次加载hibernate时都会删除上一次的生成的表**，然后根据你的model类再重新来生成新表，哪怕两次没有任何改变也要这样执行，这就是导致数据库表数据丢失的一个重要原因。
2. create-drop ：每次加载hibernate时根据model类生成表，但是sessionFactory一关闭,表就自动删除。
3. update：最常用的属性，第一次加载hibernate时根据model类会自动建立起表的结构（前提是先建立好数据库），以后加载hibernate时根据 model类自动更新表结构，即**使表结构改变了但表中的行仍然存在不会删除以前的行**。要注意的是当部署到服务器后，表结构是不会被马上建立起来的，是要等 应用第一次运行起来后才会。
4. validate ：每次加载hibernate时，验证创建数据库表结构，只会和数据库中的表进行比较，不会创建新表，但是会插入新值。
   1. 配置实体类



1. JPA根据实体类自动生成数据库表，并且生成相应的sql语句。
   1. 自动生成操作数据库语句



1. 实现接口JpaRepository

# Webjars（简单了解）

## 简介

1. WebJars是将客户端（浏览器）资源（JavaScript，Css等）打成jar包文件，以对资源进行统一依赖管理。WebJars的jar包部署在Maven中央仓库上。
2. 在开发Java web项目的时候会使用像Maven，Gradle等构建工具以实现对jar包版本依赖管理，以及项目的自动化管理，但是对于JavaScript，Css等前端资源包，我们只能采用拷贝到webapp下的方式，这样做就无法对这些资源进行依赖管理。那么WebJars就提供给我们这些前端资源的jar包形势，我们就可以进行依赖管理。

## 简单使用

* 1. 引入依赖

WebJars主官网 查找对于的组件，比如Vuejs



* 1. 前端开发页面引入

C:\Users\MONSTE~1\AppData\Local\Temp\1541064125(1).png