SpringBoot

Profile配置

解决你的配置在别的电脑上显示相同的内容

Log配置

打印日志

模板引擎： 最后输出都是以流进行的。

Converter 和Formatter:

接收Json等数据格式

怎么接收参数的？

属于Controllor层

Hash是最快的。所以map为最快

Map与Tree 哪个查询快，需要根据场景

这两个集合是查询最快的

Select \* from table

星号 \* 不建议用， 具体需求具体使用，查询性能更好

查询范围对性能有影响

索引关键字

IO比较重要！

MQ异步消息队列：确保消息不丢失，性能必须降低

异步：磁盘IO、网络IO、

功能权限、数据权限

RBAC： role basic Access control

基于角色权限管理

部署：环境部署、

Session：统一会话，了解底层运行机制

HttpClient/OkHttp :java发送http请求工具包；爬虫必备

坑：底层sock有问题。无法及时释放TCP进程。CloseWait

MQ框架与SpringBoot 融合很好。

加密算法：一般用Hash 方式。

Md5

Sha：256，512

彩虹表 采宏表

SSH 是rsa 非对称加密：公钥加密，私钥解开

公钥无法解密公钥

Springboot-> mybatis-> redis 作为重点

代码拆分：

Dao包 ：存放数据库操作代码

Model ：存放entity

Exception：

Common:

Config：配置文件

使用maven :1.5.3

Maven:依赖管理、项目构建（如何清理，安装依赖，编译，打包）、测试（打包即将resource静态资源、.Class、入口文件 打包成jar包）

执行maven test 则会自动执行需要测试的位置

文件名：pom.xml

Modelversion?

配置文件必须要配置的标签：Gav:

1.Group

2.artifactId

3.version

Module标签：模块

Dependencies:依赖标签 、使用依赖

Repositories标签：仓库定义

Build标签：

**DispatherServlet :就是如何执行指定action中的方法。需要具体了解处理过程**

**配置文件描述**

**EnableAutoConfiguration ：谁来启动？谁来做这个事情**

**EnableAutoConfigurationImportSelect:做以上事情。找到什么就初始化什么**

**BootApplication.java为主函数，最好放在最外层的包。则能够扫描所有项目**

**注解：ComponentScan:扫描类。默认扫描 这个 类(jar包)中以下的所有类**

**有了component就会创建这个实例**

**Component:configuration、service 、controller**

**什么时候用get/post**

**有多少种请求方式？**

**Linux命令**

**Nohup Java ….后面加上& 后台运行**

**Ps -ef | grep 查看进程**

**Sudo kill -9 进程号**

1、

2、如果用post，接收方式与get有什么区别？

3、controller如何校验，怎么校验更合适。

今日重点：

1. 参照教程大纲，复习数据库

2. 熟悉领域模型的概念，理解领域模型驱动的设计方式，后续会用到

3. 掌握SpringBoot

3.1 启动方式

3.2 参数的接收方式，课堂上只演示了Get，自行找资料解决以下问题：

3.2.1 使用对象接收更多的请求参数

3.2.2 使用Post请求方法，接收请求参数有什么不同？

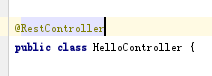
3.3 yaml文件及@Configuration的应用

3.4 使用@Resource注入bean

3.4.1 遗留问题：@Resource 和 @Autowired 有什么区别？

下节课要点：

Mybatis及其与SpringBoot的整合



这个注解 就不用使用@ResponseBody

下面直接使用 @Mapping(path=”xxx”)就可以访问了

Springboot主函数一定要放在代码最外层才能扫描到所有代码才能正常运行

Http Content-Type类型

常用注解: @RestController 、@Controller：@GetMapping,@PostMapping

@GetMapping(path=”/hello”, consumes=’application/json’)一般不用

@GetMapping(path=”/hello”)：请求参数的接收方式：属性，Model包装类，Json

请求参数校验。

@Configuration

@ConfigurationProperties(prefix=”XXX”)双引号里代表前缀名

@RequestBody:获取值放到指定的对象中

@ResponseBody:将获取的值返回

@Component:

@Resource、@Autowired：属性注入

**Get:请求参数放在header（收藏网页的时候，是会收藏里边的参数）**

**Post:请求参数放在body，不意味着不能放到header中 （网络通讯安全与否 没有跟 get有区别。安全性几乎一样） （只是post在收藏栏中无法收藏参数）Post能传递更多的数据。**

尽量用post请求 ，因为避免浏览器收藏一些信息。

Post比get实际上是比get更安全，但不代表安全

什么时候用get，什么时候用post

Get：获取资源，只去查询，不引起数据变更（数据库，硬盘，等）get更轻量，所以能用就用

Post:引起到数据变更的时候使用post（数据状态变化）(例如清除缓存，下订单，用户登录注册)（仅限于get与post）

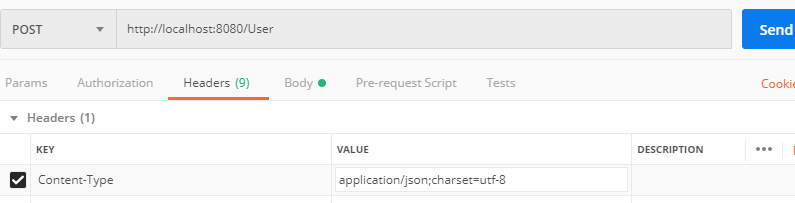
Converter:转换许多类型，json，日期等等。

Springboot自定义json解析 如何去掉null值（怎么把默认的替换）

2.model

3.model json @requestBody

4.map json @responseBody 输出的时候 null与控制字符串都输出了空字符串，有问题！



请求方式为 PostMapping

校验：

If(){throw new RuntimeException(“年龄不符合要求”)}

异常可以写在简历中

校验具体实现方式：

@Length(min = 5,max = 32)

@Email

(SpringBoot参数校验（关键字 JSR,Hibernate.validator）)

JSR:NOT ALL,NOT BLANK,DECIMAL

校验标签的操作过程，在编译前，先告诉编译器这里的值，待会跑到这个地方就跳到Validator中获取校验值，返回布尔类型 true or false 然后输出

自定义校验：

先定义标签，然后创建校验类需要实现 API

ConstrainValidator

Return 校验规则变量.matches



怎么定义一个属于自己的API 而且让人看起来舒服

{“msg”:”error”,

“code”:”0/1”,

“data”:list/存放的数据

}

0/1:101用户不存在

102密码不存在

JSR是个规范，具体有Hibernate.validator实现

Springboot+mybatis数据库

该数据库与框架用处统一参数校验，统一处理异常

今日总结：

1. 参数接收的多种方式，包括Json的接收和返回

2. 参数的校验，及如何自定义校验

3. 异常场景合理处理方式：抛异常，而不是直接return字符串或对象

4. 规范的API返回格式

问题：

1. 序列化返回Json时，如果包装类中有成员变量未设置null，怎么让它不返回？（自定义Json处理）

下节课重点：

1. 异常的统一处理

2. MyBatis的应用

关键字：

JSR： 校验规范

Hibernate Validator： 校验的实现

Springboot默认的Json序列化和反序列化的实现