## 自我介绍

您好：

我叫秦丛，来自河北石家庄，16年毕业于河北北方学院。加入Java这个行业有两年时间了，上一家公司是**北京知达客信息技术有限公司，**担任java开发工程师，在这期间我参与开发了四个项目，在这些项目中主要使用的框架有SSH SSM一些主流框架。近期做的一个项目是以销售鞋业为主的网络平台，由线下实体经营和线上购买商品的一个中心平台，为客户提供个性化服务，也为企业创建一个良好的收入渠道。在选择服务架构的时候我们考虑到高并发、高可用、系统安全和处理海量数据这方面的问题，服务器整体结构上我们选取常用的分布式+集群形式。软件架构使用SOA架构，实现控制层和服务层的分离。后台框架使用的是spring+springMVC+myBatis+dubbo实现，在前端使用AngularJS+Bootstrap，还有就是使用SpringSecurity来控制用户的登陆和权限，防止Sql注入等系统安全问题。这个网络平台前台主要有登陆注册、站内搜索、商品详情页、用户管理等模块；后台主要有客户管理、订单管理、商品管理、销售管理、客服管理等模块。

## **责任描述**

我负责该网络平台前台的登陆注册、站内搜索、商品详情页等模块，在商家后台负责商品管理代码的编写；根据项目需求完成程序开发、代码调试、单元测试等工作，配合团队完成开发项目。

### 用户注册

#### 密码加密

在用户注册的时候我们使用BCrypt算法实现密码加密。当我们添加用户信息的时候，创建BCrypt密码加密的对象，然后调用加密encode方法，实现加密，放到数据库。还需要在springsecurity的配置文件中指定加密算法。

### 用户登录

用户的登录要用数据库数据来实现登录，首先我们需要创建一个实现了UserDetailsService接口的实现类，并在后端定义 loadUserByUsername的方法。在该方法中，首先获取登录的用户名，然后根据用户名去查询用户的密码，最后，把用户名和密码交给SpringSecurity与我们输入的用户名和密码进行匹配。如果匹配成功，跳转到用户的当前页。

#### 什么是单点登录（SSO）

这里涉及到单点登录sso，（在分布式系统当中，只要用户在任何一个系统中登录之后，都可以访问其他的系统，不需要二次登录。我们在近期的项目中使用的CAS来解决单点登录。）

#### SSO单点登录流程

实现单点登录，首先SSO客户端把用户请求重定向到SSO服务端，SSO服务端实现用户身份认证，如果认证通过，SSO服务端会产生一个随机的Service Ticket,发送给用户。当用户访问其他系统的时候，去SSO客户端验证票据，如果验证通过，允许用户访问。

#### CAS实现单点登陆

我们使用的是cas和springsecurity的整合，cas是用来实现登陆，SpringSecurity是用来授权的，

在实现用户登录这个模块当中，我首先将cas的服务上传到服务器当中，默认访问的用户名casuser密码mellen,将SSL安全证书放入到cas的服务上，修改我们的配置文件，指定数据源，指定数据库，密码加密的方式(MD5),将cas和springsecurity的依赖导入到项目当中，在spring-security的配置文件中，指定登录的url，认证过滤器，认证管理器，认证提供者，认证类，票据验证器，登出的监听器和地址。其中，认证类实现了UserDetailsService接口，并重写了loadUserByUsername方法，这里我们只是实现用户的权限，而并没有实现用户的认证。

先登录 后认证

### 站内搜索

#### 搜索实现的原理

网络爬虫（网络蜘蛛）去各个网站上抓取网页，然后保存到服务器上，通过分词器,分析网页中的出现最多的关键词，按照关键词出现的次数进行排名。当我们搜索，实际上就是关键词出现的次数最多对应的网页的排名。

#### 配置Sorl环境

我们网站实现搜索，是使用了solr。Solr是一个实现搜索的开源服务。它本身是一个war包，可以放到tomcat中。

我们配置中文分词器IK Analyzer，首先添加分词器的jar包，然后把扩展字典、停用字典、分词器的配置IKAnalyzer.cfg.xml复制到WEB—INF/class目录下。最后，修改 Solrhome下的 schema.xml 文件，配置一个 FieldType，使用 IKAnalyzer。

我们根据搜索的业务需要，指定了业务域，比如标题，图片，价格，规格等。首先业务域的定义，常用属性有域的名字（索引库的字段名），域的类型type，是否索引index，是否储存store，是否必须require，是否多值。

定义域分为一般域、复制域（通过复制域可以实现搜索多个域中的关键字）、动态域（把类似域名的域，设置为一个独立的域）

#### 实现Solr的数据导入

首先，去指定pojo类中属性和索引库域的对应关系。然后查询数据库中所有需要导入的数据。最终通过solrTemplate对象的saveBeans（）把数据导入到索引库中。

#### 使用SpringDataSorl实现简单查询

首先创建一个搜索对象Query，然后创建一个条件对象Criteria，设定查询的域以及查询的关键字。并添加查询对象中。最终调用sorlTemplate的queryForPage（）执行查询。

### 商品详情页

这里就用到了网页静态化FreeMarker，FreeMarker是一款[模板引擎](https://baike.baidu.com/item/%E6%A8%A1%E6%9D%BF%E5%BC%95%E6%93%8E/907667" \t "_blank)：需要数据和模板，把数据填充到模板中，生成静态页面。模板主要有四个元素：文本，注释（#），插值（$），FTL指令。

freemarker大致原理是：将页面中所需要的样式放入FreeMarker文件中，然后将页面所需要的数据动态绑定，并放入Map中，通过调用FreeMarker模板文件解析类process()方法完成静态页面的生成。

步骤：

创建一个 Configuration 对象,设置模板文件所在的路径。设置模板的字符集。一般就是 utf-8.

创建一个模板对象,创建一个模板使用的数据集,一般是 Map。创建一个 Writer 对象，一般创建一 FileWriter 对象，指定生成的文件名。调用模板对象的 process 方法输出文件。

关闭流

### 商品管理

商品管理里面包括商品的新增，商品 修改、删除、上下架

#### 商品的三级分类显示

商品三级分类的显示，我们会根据分类的父id（parent\_id）查询分类。

默认情况，分类页面显示顶级分类，也就是parent\_id=0的分类。当我们

点击某个顶级分类的时候，会根据这个顶级分类的id，去查询这个顶级分类

的二级分类。当我们点击某个二级分类，根据这个二级分类的id，去查询它的三级分类。

首先在后台定义一个根据parent\_ id查询分类列表的方法，然后在前端的js页面的controller层，接收传递过来的parentId，调用后台的方法。

当进入三级分类页面的时候，默认显示顶级列表，使用ng-init在页面初始化的时候，调用js的controller层，通过service获得后台查询到的parent\_id=0的数据的集合。如果成功获得就将数据添加到$scope的变量中，使用ng-repeat循环集合，最终使用angularjs的表达式显示。

二级分裂列表显示，通过获得点击分类的id，这个id就是二级分类的parentid，同样道理，查询二级分类内容。三级依然。

#### 面包屑导航

根据商品的三级分类，面包屑导航也分为三级。在进入分类页面的时候，页面中

默认显示顶级分类，面包屑导航显示固定字符--“顶级分类”。当我们查看二级分类的时候，

面包屑导航显示我们选中的顶级分类的名称。当我们查看三级分类的时候，面包屑导航显示的是我们选中的二级分类的名称。

实现面包屑导航三级分类，首先定义一个分级的级别，默认是1。当我们点击”查看下级分类”的按钮时候，能够获取到当前分类对象，并把当前分类对象的名称显示在面包屑导航中。

当我们点击某个面包屑导航，首先设置分类级别，然后根据不同的分类级别，给不同的分类对象设置内容。

使用ng-if判断当前的分类级别，如果当前的级别不等于3的时候才显示“查询下级”按钮。

#### 新增商品分类

我们在新增商品分类时首先要注意当前显示的是，哪一级分类的列表。我们新增的商品分类就会添加到这个分类下。所以，我们需要一个新增分类的时候，使用变量去记住上级分类的的id。

首先我需要在js的界面中定义一个保存Id的变量。然后在查询三级分类的时候将查询的parentId保存到这个变量中。在添加分类时，将这个变量值设置到新增分类对象的parentid属性中。

#### 修改商品分类

修改分为两步:先查询，在修改。

首先，我们点击修改的时候，把需要修改的ID传入到后台进行查询。如果成功获得对象，把对象添加到$scope中。最后，在页面中使用ng-repeat循环$scope中的变量，并使用angularJS的表达式显示。

然后对数据修改，将所修改的信息以对象形式传入后台，修改成功之后进行页面刷新。

#### 删除商品分类

在删除商品分类时，首先要判断选中删除的商品分类是否含有子分类。如果有，就不能删除此分类。如果没有子分类，就删除此商品分类。

在可以删除分类的时候，我们使用的是批量删除。首先我们先定义一个存放删除ID的集合，并定义一个添加ID的方法，当我点击单选框的时候将ID添加到集合中，当我取消选中，将取消的ID移除。这里使用到了$event事件对象。

当我们点击删除的时候，将选中的ID集合传入到后台，循环遍历，调用删除方法。当删除成功时，返回结果对象Result，进行页面刷新。

#### 新增商品信息

新增商品我们要同时操作两张表，tb\_goods商品表和tb\_goods\_desc商品描述表。

首先创建一个包装类，里面包含两个属性goods商品的基本信息，goodsDesc商品的扩展信息。

当我们输入内容，点击保存的时候，将商品信息和商品的描述信息一起添加包装类中。我们将商品的名称，价格，副标题等一些基本信息插入到tb\_goods商品表中，并通过返回插入数据的id，这个id是通过mybatis的sql映射文件，里面insert标签中包含selectKey标签调用LAST\_INSERT\_ID的方法获得的。获得这个id之后，以这个id为外键向tb\_goods\_desc表中插入商品的描述信息。这里需要注意的是，商品介绍的内容是通过富文本编辑器获得的。要使用富文本编辑器，在页面中导入富文本编辑器的css和js，然后在页面中添加js的代码，初始化富文本编辑器。然后在页面中通过js的edit.Html()用来获取编辑器的内容。

#### 商品图片上传

我们的图片是通过专门的文件服务器FastDFS来管理的。FastDFS是可以放置图片和文件的服务器，我们录入的所有的图片信息全部放在这个文件服务器上。图片上传的第一步，我们要在工程中添加文件上传的依赖fastdfs和fileupload。并添加工具类FastDFSClient。然后在商家后台添加配置文件application.properties（服务器地址），fdfs\_client.conf(图片上传路径)。并在springMVC的配置文件中配置图片上传解析器。

第二步、在前端的js页面，我们通过AngularJS实现上传，这里面需要特别设置两个属性：Content-Type header属性，这个属性的默认值是application/json。我们如果要上传图片，需要把这个属性设置为null，这样就会使用 multipart/form-data值。另外需要

设置transformRequest: angular.identity，这个属性主要是上传对象序列化。

当点击上传按钮的时候，我们在后端通过FastDFS的客户端上传图片，并获得上传地址。最后在添加商品描述对象的时候，把图片地址插入到tb\_goods\_desc表中的item\_images字段中。

#### 一级分类下拉选择框

页面启动时候，查询一级分类，也是parentId=0的分类，使用ng-options循环展示分类的id和名称。

#### 二级分类下拉选择框

当一级分类选中的时候，使用angularJS的内置对象$wathch获得选中的一级分类的id值，

以这个id值作为二级分类的parent\_id值，查询出二级分类。然后使用ng-options循环展示分类的id和名称。

#### 三级分类下拉选择框

当二级分类选中的时候，使用angularJS的内置对象$wathch获得选中的二级分类的id值，

以这个id值作为三级分类的parent\_id值，查询出三级分类。然后使用ng-options循环展示分类的id和名称。

#### 读取模板ID

当三级分类选中的时候，使用angularJS的内置对象$wathch获得选中的三级分类的id值，

以这个id值查询当前三级分类对应的模板id。然后使用angularjs的双括号表达式展示

#### 品牌选择

当模板id获得值之后，查询模板对象，获得模板对象中保存的品牌字段的json格式数据，

然后把json格式数据转化成js对象。然后使用ng-options循环展示品牌的id和名称。

#### 扩展属性

当模板id获得值之后，查询模板对象，获得模板对象中保存的扩展属性字段的json格式数据，然后把json格式数据转化成js对象。使用ng-repeat循环展示。

#### 显示规格及规格选型

[{"key":value,"array":[{"key":value},...]},...]

要实现前端效果，首先

需要考虑前端页面显示的数据结构。当数据结构是json的对象集合，其中的

对象属性又是对象集合。也就是集合对象中对象属性包含集合对象。这种形式，对应着Java的

List<Map>形式。

通过Angularjs的$watch的内置对象，监控模板id的变化。当我们在前台选中模板id的时候，

通过模板id找到规格id，然后通过规格的id找到对应的规格选项，然后把规格选项集合put到

规格map中，就拼装成了我们需要的对象属性包含集合对象形式。

最终，用ng-repeat进行循环展示。

#### 保存选中规格选项到变量中

因为规格选项保存在tb\_goods\_desc表的specification\_items字段中，这个字段的格式是[{“attributeName”:”规格名称”,”attributeValue”:[“规格选项1”,... ] } , ...]

这个格式json数组中包含对象集合，单个对象中又包含对象数组。

当我们点击某一个规格选项的时候，首先判断数组中是否这个规格对象，如果包含这个规格对象，我们就获取这个规格对象并添加新的规格选项。否则，如果不包含这个规格对象，我们就构造一个新的规格对象。

当我们取消选中规格选项的时候，首先判断这个规格选项是否是该规格对象中最后一个规格选项，如果是，就移除这个规格对象。如果不是，就从规格对象中移除这个规格选项。

#### SKU商品信息

要实现前端效果，首先需要考虑前端页面单条记录显示的数据结构。

[{spec:{},price:0,num:99999,status:'0',isDefault:'0' } ,...]

这个json格式数组中包含对象集合，单个对象中又包含对象。

根据刚才保存的规格选项，循环每个规格对象，里面包含规格名称和规格选项值。

循环SKU集合中的每个对象，通过js的深克隆，克隆新的SKU对象，把循环的每个规格名

和它下面的每一个规格选项循环添加到克隆的SKU对象中。最终，把克隆的每个SUK对象添加

到一个集合中。最后，对SKU集合对象循环展示。

当我们点击提交的时候，会获得每一个SKU对象，进行循环添加到tb\_item表中。其中，SUK的标题是

商品名称+规格名称

#### 启用规格

首先，设置启动规格的单选框的值，当选中的时候使用ng-true-value设置为1，未选中的时候使用ng-false-value设置为0。当我们启用规格的时候，使用ng-if来判断是单选框是否被选中，如果选中显示规格选项，否则隐藏。

当我们未启用规格，就保存单个SKU信息。当启用规格，才按照SKU的条数进行保存。

#### 商品上下架

有上架和下架两种状态。上架代表可以销售，下架代表不能销售。

修改商品表的is\_marketable字段。当设置1表示上架、0表示下架。

## Spring

### 简介：

是为了解决企业应用开发的复杂性而创建的，使用基本的JavaBean来完成以前只可能由EJB完成的事情。然而，Spring的用途不仅限于服务器端的开发。从简单性、可测试性和松耦合的角度而言，任何Java应用都可以从Spring中受益。Spring的核心是控制反转（IoC）和面向切面（AOP）。简单来说，Spring是一个分层的JavaSE/EE 一站式轻量级开源框架。

### Spring优点：

1. 方便解耦，简化开发

Spring就是一个大工厂（或者说容器），可以将所有对象创建和依赖关系维护，交给spring进行管理。Spring工厂主要用来生成bean

1. AOP编程的支持

Spring提供面向切面编程，可以方便的实现对程序进行权限拦截、运行监控等功能

1. 声明式事务的支持

只需要通过配置就可以完成对事务的管理，就不需要手动编程了。

1. 方便程序的测试

支持对Junit4支持，可以通过注解方便的测试spring程序

1. 方便集成各种优秀的框架

Spring不排斥各类优秀的开源框架，其内部提供了对各种优秀框架（如：Struts、Hibernate、Mybatis等）的直接支持

1. 降低JavaEE API的使用难度

对JavaEE开发中非常难用的一些API（JDBC、JavaMail、远程调用），都提供了封装，是这些API应用难度大大降低。

## Struts2

### Struts2简介

Struts2是一个优秀的MVC框架，基于Model2设计模式，由传统的Struts1和WebWork两个经典框架发展而来。

### Struts2核心功能

1. 允许POJO对象作为Action
2. Action的execute方法不再与ServletAPI耦合，更易测试
3. 支持更多视图技术（JSP、freemaker等）
4. 基于Spring AOP思想的拦截器机制，更以扩展
5. 更强大、更易用输入校验功能

### Struts2工作原理

1. 客户端初始化一个指向Servlet容器的请求
2. 这些请求经过一系列的过滤器，比如一个叫做ActionContextCleanUp的可选过滤器
3. 调用FilterDispatcher并询问ActionMapper是否需要调用某个Action
4. 如果Action决定调用某个Action，FilterDispatcher把请求的处理交给ActionProxy
5. ActionProxy通过ConfiguartionManager询问框架的配置文件，找到需要调用的Action类
6. ActionProxy创建一个ActionInvocation的实例
7. ActionInvocation实例使用命名模式来调用，在调用Action的过程前后，涉及到相关拦截器（Intercepter）的调用
8. Action执行完毕，ActionInvocation根据struts.xml中的配置找到对应的返回结果，返回结果通常是JSP或者freemaker模版

### Struts2工作流程

1. 客户端浏览器发出HTTP请求
2. 根据web.xml的配置，该请求被FilterDispatcher接收
3. 根据struts.xml配置，找到需要调用的Action类和方法，并通过IOC方式，将值注入给Action
4. Action调用业务逻辑组件处理业务逻辑，这一步包含表单验证
5. Action执行完毕，根据struts.xml中的配置找到对应的返回结果Result，并跳到相关页面
6. 返回HTTP响应给客户端浏览器

## springMVC

### Springmvc工作流程：

1. 用户请求发送到前端控制器DispatcherServlet
2. DispatcherServlet收到请求调用HandlerMapping处理器映射器
3. 处理器映射器找到具体的处理器（根据xml配置、注解进行查找），生成处理器对象及处理器拦截器一并返回给DispatcherServlet
4. DispatcherServlet调用HandlerAdapter处理器适配器
5. HandlerAdapter经过适配调用具体的处理器
6. Controller执行完成之后返回ModelAndView
7. HandlerAdapter将Controller执行结果ModelAndView返回给DispatcherServlet
8. DispatcherServlet将ModelAndView传给ViewReslover试图解析器
9. ViewReslover解析后返回具体的视图view
10. DispatcherServlet根据view进行渲染视图
11. DispatcherServlet响应用户

### Springmvc常用注解：

@controller

@RequestMapping

@Rrsource 和@Autowired

@ModelAttribute和SesionAttribute

@PathVariable

@RequestParam

@ResponseBody

@Component

@Repository

## Struts2和springmvc架构对比

### struts2的漏洞问题

早些的项目一般都用spring，struts2，hibernate架构。因为hibernate的开发速度比较快，struts2大家也用的比较熟。

但是后来struts2的OGNL慢慢报出有安全漏洞，我当时还专门查看了是什么漏洞，好像这些漏洞可以使黑客获得网站服务器的“最高权限”，这些服务器可以变成“肉鸡”。

而且大型项目的话，ssh架构响应速度会有点慢，所以后来慢慢开始采用ssm的框架组合，使用springmvc替换了struts2，大大提高了响应速度。

### Struts2的多线程问题

我说说struts2和springmvc的区别吧。

struts2的开发是基于类的，前台传到后台的参数绑定都是基于成员变量的，成员变量属于共享变量，会出现多线程的安全问题。所以struts2就不能采用单例模式开发，必须每一个请求，都创建一个新的action实例来处理。

我们都知道对象的创建和销毁都需要消耗时间和系统资源的，所以在大型项目中访问量过大就会创建大量的action，引起响应速度变慢的问题。

### Struts2造成的堆内存和栈内存溢出问题

如果jvm的堆内存设置的太小，有可能造成OutOfMemery堆内存溢出问题，解决办法的话可以使用-Xmx把最大堆内存调大(一般设置为2048M)；

如果为了在大并发情况下能创建更多的线程，而把jvm的线程栈内存设置太小，栈内存被用完后会造成StackOverFlowOfMemery栈内存溢出问题。

当然，也可以使用-Xss调整每个线程的栈内存大小(一般设置为512K)。

### 成员变量管理问题

还有一个问题是，我记得当时我们的项目开发越来越大，后来成员变量真是太多了，哪些方法用了哪些变量，查找和管理起来太难了。

### 介绍springmvc的优点

所以当我后来的项目采用springmvc开发的时候发现基于方法的参数绑定，结构真是太清晰了。

而且springmvc是单例模式的，controller实例只有一个，不需要多次创建，响应速度会很快。

Springmvc的参数绑定种类也特别多，比如常规的HttpServletRequest，HttpServletResponse，HttpSession，基本数据类型，pojo，数组，list，set，map都可以。

而且springmvc支持restful风格的编程，使用路径来传递参数，在方法形参中使用注解@PathVariable来接收参数。

说一下springmvc的基本原理吧

### Springmvc成员变量问题

不过参数绑定的时候要注意一个问题，就是成员变量的处理。

我记得第一次使用springmvc开发的时候遇到过一个成员变量的问题。当时我们设计了一个baseController类，作为所有controller的基类。我们在基类里面设置了成员变量HttpServletRequest，并且创建了获取HttpServletRequest参数的方法。

导致我们并发测试的时候总有些结果是重复的，后来定位到问题是多线程的成员变量共享的问题，最后使用了ThreadLocal来专门存放这些共享变量，让变量在每一个线程空间里都有一个副本，而且线程间互斥。

### ThreadLocal底层原理

当时我还不了解ThreadLocal这个类，所以还专门学习了一下它的底层，原来它底层有一个内部类叫ThreadLocalMap，它不对外使用。

当使用ThreadLocal存值时，首先是获取到当前线程对象，然后获取到当前线程本地变量Map，最后将当前使用的ThreadLocal和传入的值放到Map中，也就是说ThreadLocalMap中存的值是[ThreadLocal对象, 存放的值]，这样做的好处是，每个线程都对应一个本地变量的Map，所以**一个线程可以存在多个线程本地变量**。

## Hibernate

Hibernate简介：hibernate是一个开源框架，是对象关联关系映射的框架，对JDBC做了轻量级的封装，我们就可以使用面向对象的思想来操作数据库。

### Hibernate工作原理：

1.通过Configuration config = new Configuration().configure();//读取并解析hibernate.cfg.xml配置文件

2.由hibernate.cfg.xml中的<mapping resource="com/xx/User.hbm.xml"/>读取并解析映射信息

3.通过SessionFactory sf = config.buildSessionFactory();//创建SessionFactory

4.Session session = sf.openSession();//打开Sesssion

5.Transaction tx = session.beginTransaction();//创建并启动事务Transation

6.persistent operate操作数据，持久化操作

7.tx.commit();//提交事务

8.关闭Session

9.关闭SesstionFactory

### Hibernate中get和load的区别

Hibernate对于load方法认为该数据在数据库中一定存在，可以放心的使用代理来延迟加载，如果在使用过程中出现问题，就会抛出异常；而对于get方法，hibernate一定要获得真实的数据，否则就返回空。Load支持延迟加载，get不支持延迟加载。

### Hibernate怎样实现类之间的关系？

类与类之间的关系主要还是体现在表与表之间的关系进行操作，它们都是对对象进行操作，我们在程序中把所有表与类都映射在一起，通过配置文件中的many-to-one、ont-to-many、many-to-many。

### Hibernate查询数据的几种方式：

1. 导航对象图查询
2. OID查询
3. HQL查询
4. QBC
5. 本地SQL

## MyBatis

MyBatis是一款优秀的持久层框架，它支持定制化SQL、存储过程以及高级映射。MyBatis避免了几乎所有的JDBC代码和手动设置参数以及获取结果集。MyBatis可以使用简单的XML或注解来配置和映射原生信息，将接口和Java的POJOs(Plain Old Java Objects,普通的Java对象)映射成数据库中的记录。

### MyBatis工作流程

通过Reader对象读取MyBatis映射文件

通过SqlSessionFactoryBuilder对象创建SqlSessionFactory对象

获取当前线程的SQLSession

事务默认开启

通过SQLSession读取映射文件中的操作编号，从而读取SQL语句

提交事务

关闭资源

**Mybatis中的事务是默认开启的，因此我们在完成操作以后，需要我们手动去提交事务！**

#### #{}与${}的区别

* #{}解析传递进来的参数数据。
* ${}对传递进来的参数原样拼接在SQL中。
* #{}是预编译处理，${}是字符串替换。
* 使用#{}可以有效的防止SQL注入，提高系统安全性。

#### Mybatis一级缓存

##### **mybatis默认是支持一级缓存的。**

mybatis提供查询缓存，如果缓存中有数据，就不用从数据库中获取，用于减轻数据压力，提高系统性能

一级缓存是sqlSession级别的缓存，在操作数据库的时候，需要构造sqlSession对象，在对象中有一个数据结构（HashMap）用于存储缓存数据。不同的SqlSession的缓存区域（HashMap）是互相不受影响的。

##### 那当什么时候一级缓存会被清空呢？

通过测试发现，当在第二次查询数据库之前对数据库进行了写的操作后，第二次查询就不会从缓存中查询结果，而是直接从数据中查询结果。另一种就是在第一查询结果后，手动的方式清空一级缓存（使用sqlSession.clearCache();方法。）

#### Mybatis二级缓存

二级缓存的作用域是mapper的同一个namespace。不同的sqlSession两次执行相同的namespace下的sql语句，且向sql中传递的参数也相同，即最终执行相同的sql语句，则第一次执行完毕会将数据库中查询的数据写到缓存，第二次查询会从缓存中获取数据，不再去底层数据库查询，从而提高效率。

在MyBatis配置文件（mybatis-config.xml）中开启二级缓存(详细过程自己百度搜索开启)

//value属性默认为false

在\*\*Mapper.xml中开启当前mapper的namespace下的二级缓存

二级缓存是Mapper级别的缓存。多个SqlSession去操作同一个Mapper的sql语句，多个SqlSession可以共用二级缓存，二级缓存是夸SqlSession的。（二级缓存Mybatis默认是关闭的，需要自己去手动配置开启或可以自己选择用哪个厂家的缓存来作为二级缓存）

## Hibernate和mybatis框架对比

### Hibernate优缺点

优点：

我再说一下早期用的hibernate吧，hibernate是一个完全的ORM对象关系映射的dao框架。

如果单表操作多的话，几乎可以完全避免写sql语句，由hibernate帮我们封装好的方法来操作数据库。不需要关心具体的数据库，只需要配置一个数据库方言，指定数据库类型。

所以hibernate开发很快，而且便于数据库的移植。

缺点：

但是缺陷是查询和更新都是针对所有字段的，这样使得执行效率很低。并且hibernate启动时需要检查数据库，如果没有还需要创建表结构，所以启动慢。

并且hibernate有一级和二级缓存，如果缓存多的话这样占内存就很高。

对于批量的操作支持不够。对象之间的对应关系如果很错综复杂，也不适合OR映射

1. 由于对持久层封装过于完整，导致开发人员无法对SQL进行优化，无法灵活使用JDBC的原生SQL，

Hibernate封装了JDBC，所以没有JDBC直接访问数据库效率高。要使用数据库的特定优化机制的时候，不适合用Hibernate   
二、框架中使用ORM原则，导致配置过于复杂，一旦遇到大型项目，比如300张表以上，配置文件和内容是非常庞大的，另外，DTO满天飞，性能和维护问题随之而来  
三、如果项目中各个表中关系复杂，表之间的关系很多，在很多地方把lazy都设置false，会导致数据查询和加载很慢，尤其是级联查询的时候。  
四、Hibernate在批量数据处理时有弱势，对于批量的修改，删除，不适合用Hibernate,这也是ORM框架的弱点

### Mybatis优点

易于上手和掌握

sql写在xml里，便于统一管理和优化。

查询可以指定返回某几个字段

提供xml标签，支持编写动态sql。

Mybatis的话就可以很好的规避掉hibernate的缺点，这也是mybatis使用越来越流行的原因。

特别是在sql优化方面。

我说下sql优化吧。

## Oralce和MYSQL区别

|  |  |
| --- | --- |
| Oracle | Mysql |
| Oracle要求硬件性能要高 | mysql要求低 |
|  | Mysql有多种存储引擎, 比如Myisam,不支持事务，查询快，innodb支持事务 |
| Oracle使用序列实现主键自增长 | mysql自带自增长功能 |
| oracle支持行级锁，不依赖索引 | mysql多数采用表级锁，  innodb引擎支持 行级锁，但是需要索引支持  如果表没有索引，或者sql语句没有使用索引，那么仍然使用表级锁 |
| Oracle支持事务 | mysql的innodb引擎支持事务,myisam不支持事务 |
| Oracle 支持闪回功能  从回滚段中查看一定时间内操作过的数据  比如: 修改或者删除了一条数据，可以查看5分钟前该条数据的情况 | mysql不支持闪回功能 |
| Oracle 字符串用单引号 | mysql可以使用单引号或双引号 |
| Oracle 分页使用rownum | mysql分页使用limit |

Oracle使用rownum分页

Mysql使用limit分页Redis

## MyBatis,JDBC,Hibernate的区别是什么? （客观面试题）

1）从层次上看，JDBC是较底层的持久层操作方式，而Hibernate和MyBatis都是在JDBC的基础上进行了封装使其更加方便程序员对持久层的操作。

2）从功能上看，JDBC就是简单的建立数据库连接，然后创建statement，将sql语句传给statement去执行，如果是有返回结果的查询语句，会将 查询结果放到ResultSet对象中，通过对ResultSet对象的遍历操作来获取数据；Hibernate是将数据库中的数据表映射为持久层的java对象，实现数据表的完整性控制；MyBatis是将sql语句中的输入参数和输出参数映射为java对象，放弃了对数据表的完整性控制，但是获得了更灵活和响应性能更快的优势。

3）从使用上看，如果进行底层编程，而且对性能要求极高的话，应该采用JDBC的方式；如果要对数据库进行完整性控制的话建议使用Hibernate； 如果要灵活使用sql语句的话建议采用MyBatis框架。

## AngularJS

AngularJS是一个开源Web应用程序框架。最初是由MISKO Hevery和Adam Abrons于2009年开发，现在是由谷歌维护。

angularJS是一款前端JS框架，扩展了HTML，当一个网页加载完毕，angularJS就自动开启。它可以通过<script>标签添加到HTML页面

### angularJS四大特性

#### MVC

angularJS的MVC是借助于$scope实现的。MVC只是一种手段，最终的目的是代码的模块化和重用。

#### 模块化设计

模块用于单独的逻辑表示服务，控制器，应用程序等，并保持代码的整洁性、可重用性。

#### 自动化双向数据绑定

所谓的双向绑定，简单说就是：前台界面的操作能实时的反映到数据，后台数据的变更能实时的展示到前台页面。 n

#### 依赖注入（DI）

**依赖注入(Dependency Injection,简称DI)是一种设计模式**，就是说某个对象依赖其他对象无需手工创建，在创建对象的时候，其依赖的对象由框架自动创建并入。

### angularJS的指令与表达式

angularJS通过指令扩展了HTML，并通过表达式绑定数据到HTML。

##### angularJS指令

angularJS指令是用来扩展HTML属性，前缀带有ng-.

常用的指令有ng-app：声明angularJS所管辖的区域，一般写在HTML标签或者body标签上。

ng-model：把输入域的值绑定到应用程序的变量中。

ng-init：初始化应用程序中的变量值。

ng-click：点击事件

##### 表达式

angularJS中使用{{}}绑定表达式，用于将表达式的内容输出到页面中。表达式中可以是文字、运算符、变量。

### angularJS scope

Scope(作用域)是应用在HTML(视图)和JavaScript(控制器)之间的纽带。在创建控制器的时候，可以把scope当作一个对象传递。

## Redis

1、Redis是一个基于内存的高性能key-value形式的nosql数据库，使用c语言开发的。

2、Redis的特点：支持保存多种数据结构，单个value的最大限制是1GB，不像memcached 只能保存1MB，支持数据持久化。

3、Redis支持的数据类型：String、Lists、Sets、SortedSet、hashes

## Redis穿透和雪崩的预防及解决

**缓存穿透**：是指按照key值去缓存查询，如果不存在对应的value，就回去DB数据库进行查询，这个时候，如果请求的并发量很大，就会对DB数据库造成很大的压力，这就叫做缓存穿透。

**解决方法**：1、设置布隆过滤器

1. 如果一个查询返回的数据为空，不管数据不存在还是系统故障，我们仍然把这个结果进行缓存，但是它的过期时间会很短。

**雪崩**：缓存集中在一段时间内失效，所有的查询都落在数据库上，造成缓存雪崩。

**解决方案**：1、缓存失效后，加锁或者队列来控制读数据库写缓存的线程数量。

2、通过缓存reload机制，预先更新缓存，在即将发生大并发访问前手动触发加载缓存。

1. 不同的key设置不同的过期时间。
2. 做二级缓存或者双缓存策略。

## [ActiveMQ](https://www.cnblogs.com/Survivalist/p/8094069.html)

MQ全称为Message Queue, 消息队列（MQ）是一种应用程序对应用程序的通信方法。

### JMS简介：

是Apache公司旗下的一个消息总线

[ActiveMQ](http://activemq.apache.org/)是一个开源兼容[Java Message  Service](http://www.baike.com/wiki/Java+Message+Service)  (JMS) 1.1面向消息的中件间. 来自[Apache Software Foundation](http://www.apache.org/). ActiveMQ提供松耦合的应用程序架构.

### [ActiveMQ](http://activemq.apache.org/)能干什么？

　　用来在服务与服务之间进行异步通信的

### [ActiveMQ](http://activemq.apache.org/)的优势

1.流量肖锋

2.任务异步处理

特点：可以解耦合

### 优点：

　　是一个快速的开源消息组件(框架)，支持集群，同等网络，自动检测，TCP，SSL，广播，持久化，XA，和J2EE1.4容器无缝结合，并且支持轻量级容器和大多数跨语言客户端上的Java虚拟机。消息异步接受，减少软件多系统集成的耦合度。消息可靠接收，确保消息在中间件可靠保存，多个消息也可以组成原子事务。

### 缺点：

　　ActiveMQ默认的配置性能偏低，需要优化配置，但是配置文件复杂，ActiveMQ本身不提供管理工具；示例代码少；主页上的文档看上去比较全面，但是缺乏一种有效的组织方式，文档只有片段，用户很难由浅入深进行了解，二、文档整体的专业性太强。在研究阶段可以通过查maillist、看Javadoc、分析源代码来了解。

### 通信模式：

#### 1.点对点(queue)

》一个消息只能被一个服务接收

》消息一旦被消费，就会消失

》如果没有被消费，就会一直等待，直到被消费

》多个服务监听同一个消费空间，先到先得

详解：这个特点的原理是这样的，在activeMQ

#### 2.发布/订阅模式(topic)

 　　》一个消息可以被多个服务接收

　　》订阅一个主题的消费者，只能消费自它订阅之后发布的消息。

　　》消费端如果在生产端发送消息之后启动，是接收不到消息的，除非生产端对消息进行了持久化(例如广播，只有当时听到的人能听到信息)

## SpringSecurity

Spring Security是一款强大的安全认证服务框架，它的原理就是在访问我们的系统前加了一系列的过滤器，可以称为过滤器链。它的两大核心就是认证和授权。

### Spring Security主要是从两个方面解决安全性问题：

#### web请求级别：

使用servlet过滤器保护web请求并限制URL级别的访问

#### 方法调用级别：

使用Spring AOP保护方法调用，确保具有适当权限的用户采用访问安全保护的方法。

## cookie和session的区别和用法

  1、Cookie和Session都是会话技术，Cookie是运行在客户端，Session是运行在服务器端。  
     
   2、Cookie有大小限制以及浏览器在存cookie的个数也有限制，Session是没有大小限制和服务器的内存大小有关。

   3、Cookie有安全隐患，通过拦截或本地文件找得到你的cookie后可以进行攻击。

   4、Session是保存在服务器端上会存在一段时间才会消失，如果session过多会增加服务器的压力。

## Nginx

Nginx简介Nginx("enginex")是一个高性能HTTP和反向代理服务器,也是一个IMAP/POP3/SMTP代理服务器。其特点是占有内存少,并发能力强,能够支持高达50000个并发连接响应。

(1)Nginx是一个高性能的HTTP和方向代理服务器  
(2)采用C语言编写  
(3)支持的操作系统众多,windows、linux、MacOS X  
(4)安全性高,外界只能访问nginx所在服务器,nginx将请求转发内部服务器。调用后,返回调用的结果  
(5)可实现负载均衡  
(6)Rewrite功能强大

## 为什么会开发这个平台？

**考虑到在第三方平台**（天猫、京东、拍拍、一号店等），平台大、费用昂贵、依赖性强、处罚多、要求高。有线下业务、品牌、渠道、顾客等多方面资源的支持，由实体经营延伸到**自建独立网络平台**自主拥有、域名独立、不依赖任何平台、费用低廉。

## 公司信息

第一家公司（石家庄雄宇科技有限公司）

公司地址：石家庄裕华区翟营南大街71号

实习：住的地方：石家庄裕华区东岗路75号[世纪花园二期](http://sjz.zu.fang.com/house-xm1310008344/) ([裕华](http://sjz.zu.fang.com/house-a0359/)/[世纪公园](http://sjz.zu.fang.com/house-a0359-b05146/))



**乘车路线：步行-公交**

**入职时间：**2015/11 （一年零3个月）

**离职时间：**2017/02

**职位：Java工程师**

**项目经理名字：李子寒（15030148418）**

**薪资:2000-4000 (实习)**

第二家公司（**北京知达客信息技术有限公司**）

老板 ：**林炳bing漂**

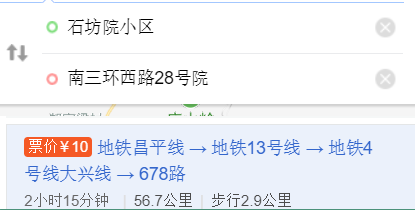
|  |  |
| --- | --- |
| 公司性质： | 民营 |
| 公司规模： | 20-99人 |
| 公司行业： | 计算机软件、IT服务(系统、数据、维护)、互联网、电子商务、外包服务 |
|  |  |

公司规模：60多人，我们项目组8人，一个项目经理，5名开发人员，2名前台，一个运维

分成多个开发小组（3-5），产品经理（3），项目经理（1-2），前台（7），后台（10），运维（5）

公司地址：北京市丰台区南三环西路28号院中林置业大厦B座301号

住的地方：北京市昌平区东关石坊院小区（鼓楼南街-永安巷）（附近昌平公园、昌平妇幼保健站）



**乘车路线：步行-地铁-公交**

**入职时间：**2017/02（一年零9个月）

**离职时间：**2018/11

**职位：Java工程师**

**项目经理名字：曹仓仓（176-0294-3287）**

**薪资:6000-8000**

## 学校信息

**学校地址：河北北方学院**

设有17个二级学院、1个系和2个教学部（基础医学院、医学检验学院、文学院、理学院、信息工程学院、法政学院、艺术学院、中医学院、经济管理学院、动物科技学院、农林科技学院、外国语学院、演艺学院、国际教育学院、第一临床医学院、继续教育学院、研究生学院、药学系、公共体育部和马克思主义理论教学部）。**乘车路线**：从张家口火车南站到河北北方学院(西校区)坐1路车

**学校周围有：**旁边有条清水河 张家口市第二医院 张家口市市政府

河北（地区分数线2012年 最高分587分，平均分527分，省控线：509） 理科 二批

分数线：电子信息科学与技术 2012 年 最高分545 平均分526 我的分数510

**年龄-----------------------------------------------------**

**今年 24岁（虚岁） 属猪**

**出生日期：1995年12月27日**

**上小学日期：2001年9月 7岁六年**

**上初中日期：2006年9月 12岁三年**

**上高中日期：2009年9月 15岁三年**

**上大学日期：2012年9月 18岁四年**

**毕业时间：2016年7月 22岁**

**年龄-----------------------------------------------------**

**今年 22岁（周岁） 属猪**

**出生日期：1995年12月27日**

**上小学日期：2001年9月 5岁六年**

**上初中日期：2006年9月 10岁三年**

**上高中日期：2009年9月 13岁三年**

**上大学日期：2012年9月 16岁四年**

**毕业时间：2016年7月 20岁**

**校长名字：**宋鸿儒

校训：厚德、远志、笃学、创新

校徽

含义：

“厚德”就是明辨曲直、是非，知荣知耻，去恶从德善。

“远志”就是要树立远大的理想和抱负，这是人的行为指南。

“笃学”就是要有刻苦学习，沉淀知识的精神。

“创新”就是具有积极进取的开拓精神和求新、创造的意识。

**专业名称：计算机科学与技术**

计算机科学与技术专业导论、线性代数、高等数学、C程序设计、数字逻辑电路、离散数学、数据结构、计算机网络、数据库原理与SQL server开发、.net程序设计、面向对象程序设计、计算机组成原理、算法分析与设计、Java应用程序开发 、操作系统、人工智能、计算机图形学、编译原理

**学历证书：本科**

## 会问的问题

### 没有上五险一金的原因：

**（没有在北京买房的目的，交了五险一金还不如每个月多发一些现金比较来的实际，北京好多公司其实都会这样，你也应该会明白）**

**在上家公司的时候，人事说咱们公司的五险一金如果要上的话都是从自己工资里面扣的，当时感觉没啥必要也就没上。**

**五险一金的名称：养老保险，医疗保险，失业保险，工伤保险，生育险，住房公积金**

### 为什么年龄和身份证上不符：

**那几年我们那计划生育抓的比较紧，所以晚上了几年。**

### 三个词形容自己：

认真细致，团队协作能力强，抗压能力强，爱笑，幽默

### 谈谈你的对软件行业发展前景的理解：

现在的生活越来越离不开网络和便捷的生活方式，软件的行业就是顺应了这一特点，通过开发的软件简化了人们一些生活杂度，所以在不久的将来必将在以后成为科技的前端和人们生活的密切相关。

### 个人优点：

喜欢钻研技术，我们的项目用到了定时器我们项目经理让我去看一些相关的资料然后把这个工作交给了我，让我三天搞定。我用了一天半的时间就搞定了并且运用到了项目中。

### 个人缺点：

我这个人做事比较直，有时候对于遇到的问题太较真，可能会耽误一些项目的进展，之前项目经理找我聊过这方面的问题，现在我对这方面已经改进，以项目为主，以公司利益为主

### 为什么离职：

我在上家公司已经待了两年了，合同也到期了。我就想换一份工作。我们的同事和项目经理也挽留我。我想换一个新的环境进行自我挑战，让自己有一个更好的前景，也让自己不在一个安逸的环境下使自己失去了自己的目标。我觉得咱们公司有自己的企业文化和实力（具体到时候自己网上查），能够给我提供一个展现自己能力的价值的平台，所以我给咱们公司投递了简历。

### 如何看待你上家公司：

每个公司都有自己的一个企业文化，从上一家公司我有深刻的体会，就是遇事要向内归因，外界的给我们带来的压力只能是客观因素，从本身去找解决办法才是我们提高的一个重要标志，逐步让我成为一个有责任心，有目标的人。和之前的同事的交流也让我学到非常多的知识和扩展交际圈，明白更多的沟通方式。

### 你凭什么要这么高的薪资?

在上个公司，项目经理给我分配了任务，本来给我规定

的时间是4天左右的时间，我在第二天的时候就基本完成了全部任务。

最近的项目是我带三个大学生采用springMvc框架进行电商产品模块的开发，并解

### 你的五年规划

前两年就是增加自己的知识量和扩展自己的知识面，后面的时间往项目经理的方向发展，为公司带来更多的利益。

### 谈谈你的学习能力与抗压能力：

最近的项目是我带三个大学生采用springMvc框架进行电商产品模块的开发，并解决他们遇到的一些问题，原本需要三个月的时间我们用了两个半月时间就完成了，为公司创造更多地价值

### 谈谈你对加班的看法：

加班其实是每个公司遇到的问题，之前我也提过我是一个能够以大局为重的人，为了公司的利益和项目组的进度我完全可以加班来完成，即使在我的任务完成的前提下，我也会和项目组待在一起帮助他们完成项目任务。

### 人事问题反问：

#### 你对我们公司有什么想了解的吗：（三选二）

在几方面 我想了解咱们公司的晋升机制

咱们公司会不会进行一些技术培训

咱们公司的五险一金是怎么上的

#### 你最快什么时候能入职

上个公司已经提前申请了离职，一个星期可以办理入职

#### 理由：

我得找房子

虽然我已经离职了 但是我的那个岗位已经有人替了,是个新人对业务还有不熟悉 我们项目经理让我帮助他梳理一下 毕竟和项目经理关系不错也是在一个圈里混的互帮互助嘛

