# 项目技术解析

## springboot aop

## logback

<https://www.cnblogs.com/zhucww/p/9319947.html>

## springboot注解详解

### 1.@Autowired

https://www.jianshu.com/p/3942cce05f71

2.@InitBinder

### 3.@Resource

## 4.springboot多模块构建

ok，所有的创建都使用quickstart

教程参考

<https://ke.qq.com/course/282793>

* 1. 4课程

1. mybatis配置文件的集成

## springboot Mybatis的xml注解实现

**参考教程**[**https://ke.qq.com/course/179440**](https://ke.qq.com/course/179440)

**第35课，和第42课资料springboot\_demo2.rar**

### 使用Mybatis自动生成工具生成

#### 实战:如何使用<https://segmentfault.com/a/1190000009058867>

#### bug1 : 对实体 "characterEncoding" 的引用必须以 ';' 分隔符结尾

解决方案<http://huangyunbin.iteye.com/blog/2172155>

#### bug2：Could not create connection to database server.

解决方案 <https://blog.csdn.net/qq_22076345/article/details/81952035>

## nginx搭建

课程60: <https://ke.qq.com/course/179440>

详细使用教程：https://ke.qq.com/course/170989

命令：ps -ef|grep nginx

安装：brew search nginx

brew install nginx

路径：

/usr/local/etc/nginx/nginx.conf（配置文件路径）

/usr/local/var/www（服务器默认路径）

/usr/local/Cellar/nginx/1.15.6/bin（安装路径）

/usr/local/var/log/nginx (日志路径)

./nginx –t

### nginx的搭建

注意不能使用中文名，最好放在 c盘program目录中

教程：<https://my.oschina.net/u/1760791/blog/1575808>

### nginx的发布

设置assetsPublicPath: './',

执行 npm run build

安装nginx：<https://www.cnblogs.com/jerrypro/p/7062101.html>

启动nginx：进入nginx的sbin目录,./nginx就可以启动。

关闭：./nginx-s stop

重启配置：./nginx -s reload

## shiro权限认证

shiro教程:https://ke.qq.com/course/289923

shiro文章：<https://blog.wuwii.com/springboot-13.html>

自定义认证：参照课程5

1. 自定义Realm继承自AuthorizingReam类重它的2个方法
2. 配置Ini文件，指定使用自定义的realm
3. 加载配置文件

自定义认证加密：参照课程6

MD5（密码，用户名，散列次数）

角色判断：用户是否拥有角色hasrole role1，role2

hasRoles，返回boolean数组

has AllRoles 是否拥有所有的权限

checkRole返回异常

权限判断：用户权限操作 ispermitted user：create，user：delete

ispermittedAll返回是否拥有所有的权限

ispermitted返回boolean数组

checkPermission 返回异常

## 添加所有的common类

1.sl4j的使用： <https://www.2cto.com/kf/201712/702543.html>

配合logback使用

依赖spring-boot-start-web组件

1. 多环境配置
2. mysql取消外键<https://blog.csdn.net/lcathm/article/details/47151615>
3. druid使用：<https://blog.csdn.net/ikownyou/article/details/79022560>

5多表查询：<https://blog.csdn.net/qq_21050291/article/details/72629595>

参考文章:https://my.oschina.net/u/1157906/blog/205122

Mybatis 示例之 foreach: <https://blog.csdn.net/isea533/article/details/21237175\>

6 pagerhelper使用：<https://blog.csdn.net/gnail_oug/article/details/80229542>

7，高并发下库存：<https://www.bilibili.com/video/av35677404/>

<https://juejin.im/post/5b8737cdf265da43737ea13a>

<https://blog.csdn.net/u013815546/article/details/53928912>

## Vue框架的学习

参考文献：<https://ke.qq.com/course/279700>

参考文献：<https://cn.vuejs.org/v2/guide/>

参考文献: <https://www.jianshu.com/p/3b82a7a475d0>

### 使用

1，引入vue.js文件

2，创建Vue数据

var app=new Vue({

//指定一个id为app的元素

el:’#app’

//指定data

data：{

message:”hello vue!”

},

methods;{

reverseMessage;function(){

}，

computed: {  
 // 计算属性的 getter  
 reversedMessage: function () {  
 // `this` 指向 vm 实例  
 return this.message.split('').reverse().join('')  
 }  
 }

}

})

界面直接调用

{{message}}

复杂的逻辑用计算属性（有缓存）

如果不想要缓存的话 用方法

### 语法

<span v-bind:title=”message”></span>

意思是将这个元素的title属性与vue的message属性绑定

<p v-if=”seen”></p>控制是否显示

<li v-for=”item in list”> 绑定数组数据

{{item.text}}

</li>

v-on添加一个事件监听器

<button v-on:click=”reverseMessage”/>

<input v-model=”message”/>双向数据绑定

<span v-once>这个奖不会改变 ：{{msg}}</span>

输出html

<p>Using v-html directive: <span v-html="rawHtml"></span></p>

### 组件化构建

自定义组件1

Vue.component(“to-item”,{

Template:”<li>这是一个代办项</li>”

})

自定义组件2

Vue.component(“to-item”,{

Props:[‘todo’]

Template:”<li>{{todo.text}} </li>”

})

<todo-item

v-for=”item in list”

v-bind:todo=”item”

v-bind:key=”item.id/>

### 计算属性和侦听器

watch 选项允许我们执行异步操作 (访问一个 API)，限制我们执行该操作的频率，并在我们得到最终结果前，设置中间状态。这些都是计算属性无法做到的。

除了 watch 选项之外，您还可以使用命令式的 [**vm.$watch API**](https://cn.vuejs.org/v2/api/#vm-watch)。

### 样式

v-bind:class 一个对象，以动态地切换 class：

|  |
| --- |
| <div v-bind:class="{ active: isActive }"></div> |

绑定的数据对象不必内联定义在模板里：

|  |
| --- |
| <div v-bind:class="classObject"></div> |

|  |
| --- |
| data: {  classObject: {  active: true,  'text-danger': false  } } |

|  |
| --- |
| <div v-bind:class="classObject"></div> |

|  |
| --- |
| data: {  isActive: true,  error: null }, computed: {  classObject: function () {  return {  active: this.isActive && !this.error,  'text-danger': this.error && this.error.type === 'fatal'  }  } } |

我们可以把一个数组传给 v-bind:class，以应用一个 class 列表：

|  |
| --- |
| <div v-bind:class="[activeClass, errorClass]"></div> |

|  |
| --- |
| data: {  activeClass: 'active',  errorClass: 'text-danger' } |

渲染为：

|  |
| --- |
| <div class="active text-danger"></div> |

## 10 阿里云系统搭建

宝塔：sangxiang 89082800

1，购买阿里云服务器

2，安装宝塔服务，

安装宝塔最新版：yum install -y wget && wget -O install.sh http://download.bt.cn/install/install.sh && sh install.sh

查看宝塔密码：sudo su root

bt default

3，安装mysql

安装ftp服务

安装filezilla并设置传输模式为主动，加密方式为只使用明文

Wget -c -t 0 下载地址

Linux下tar.xz结尾的文件的解压方法:

查找文件：find -name mysql\* -o -name \*history

du -sh mysql-8.0.13-linux-glibc2.12-x86\_64.tar.xz

查看初始密码：grep 'temporary password' /var/log/mysqld.log

ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql\_native\_password BY 'sx890828';

安装：<https://blog.csdn.net/qq_25340149/article/details/84307547>

链接时候出现is not allowed to connect mysql server

查看mysql是否启动： service mysqld status

cd /www/server/mysql/bin

登录：./mysql -u root -p

退出：exit;

解决方案：<https://blog.csdn.net/e_wsq/article/details/79695709>

4，在宝塔中修改mysql的密码，并

添加mysql的端口：宝塔中，阿里云上

1. 安装jdk

查看linux多少位：getconf LONG\_BIT

下载linux版本的jdk，上传到服务器

wget --no-check-certificate --no-cookies --header "Cookie: oraclelicense=accept-securebackup-cookie" https://download.oracle.com/otn-pub/java/jdk/8u201-b09/42970487e3af4f5aa5bca3f542482c60/jdk-8u201-linux-x64.tar.gz

Tar -zxvf jdk-8u201-linux-x64.tar.gz

添加系统环境变量

vi /etc/environment

PATH="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games"

export CLASSPATH=.:$JAVA\_HOME/lib:$JAVA\_HOME/jre/lib

export JAVA\_HOME=/sangxiang/jdk1.8.0\_201

source /etc/environment

添加用户环境变量

vi /etc/profile

export JAVA\_HOME=/sangxiang/jdk1.8.0\_201

export JRE\_HOME=$JAVA\_HOME/jre

export CLASSPATH=.:$JAVA\_HOME/lib:$JRE\_HOME/lib:$CLASSPATH

export PATH=$JAVA\_HOME/bin:$JRE\_HOME/bin:$PATH

source /etc/profile

验证安装

java -version

1. 购买腾讯云域名

ssl证书

Dns解析

1. springboot打包发布运行
2. windows下的mysql
3. 下载mysql版本5.x版本

2，停止mysql服务：net stop mysql

3，

## 11.springboot发布

1. 在需要的app添加maven
2. 在父目录中package
3. 执行java -jar 运行
4. nohup java -jar app-1.0-SNAPSHOT.jar &

所有日志输出到nohup.out中

查看java占用进程：ps -aux | grep java

杀死进程：Kill -9 端口号

## 12．Redies集群

mac下安装集群

教程：https://www.jianshu.com/p/c869feb5581d

1，mac下安装brew

ruby -e"$(curl -fsSL <https://raw.githubusercontent.com/Homebrew/install/master/install>)"

2，安装ruby

yum install ruby

1. 安装redies依赖

教程：https://www.jianshu.com/p/6e7c627703a5

gem install redis -- 安装redis依赖

因为墙的问题，gem install redis 可能会失败，改用国内的源

gem sources --remove https://rubygems.org/ 删掉原来的源

gem sources -a https://gems.ruby-china.com/ 改用国内的源

gem sources -l 查看现有的源

gem install redis 安装redis依赖

1. 需要ruby2.2.2以上

教程：https://blog.csdn.net/FengYe\_YuLu/article/details/77628094

1，安装rvm: curl -L get.rvm.io | bash -s stable

2，重启控制台

3，查看ruby版本：rvm list known

4，安装高级版的ruby：rvm install 2.3.3

5，.使用一个ruby版本：rvm use 2.3.3

1. 使用默认版本：rvm use 2.3.3 –default
2. 查看ruby版本: *ruby* --*version*
3. 安装redis依赖: gem install redis

程序集成redies

教程：<https://www.jianshu.com/p/60a322d5c3c6>

连接节点

./redis-cli -c -h 127.0.0.1 -p 7001

清空数据库

flushdb

关闭节点

redis-cli -h 127.0.0.1 -p 6379 shutdown

./redis-trib.rb create --replicas 1 127.0.0.1:7000 127.0.0.1:7001 127.0.0.1:7002 127.0.0.1:7003 127.0.0.1:7004 127.0.0.1:7005

ERR Slot 741 is already busy (Redis::CommandError)

解决方案：https://blog.csdn.net/qiushisoftware/article/details/78837855

## 13，用户登录注册，密码的加密解密

教程：https://blog.csdn.net/M\_Kerry/article/details/82725622

# 10.定时任务

# 11.Logback

# 12 ElasticSearch

# 13,ActiveMQ5消息队列

# 14.多环境配置介绍

https://blog.csdn.net/qq\_33689414/article/details/81812783

# 15.springboot响应式编程webFlux

# 16.服务端推送常用技术

# 17.阿里云服务器