

阿飞(dufyun)的博客

不管做什么，只要坚持下去就会看到不一样！在路上，不卑不亢！

个人资料

每天都在变得更好的阿飞

博客专家

原创	粉丝	喜欢	评论
179	375	180	213

等级：博客

访问量：56万+

积分：6755

排名：4339

高端网站设计

文章搜索

联系方式

其实我想要 一种美梦睡不着 一种心脏的狂跳

QQ: 742981086

java编程艺术 600756207 共同学习！

GitHub

个人博客

博客专栏

Quartz专栏

文章：9篇

阅读：67271

Java集合系列专栏

文章：10篇

阅读：12327

Java并发编程实战

文章：2篇

阅读：758

Spring常用注解介绍【经典总结】

标签：Spring注解Spring常用注解annotationSpring注解详解

2017年07月29日 11:02:211785人阅读评论(0)收藏举报

分类：Spring&SpringMVC (17)

版权声明：本文为博主-阿飞(dufyun)-原创文章，未经博主允许不得转载，但请标明出处，谢谢！https://blog.csdn.net/u010648555/article/details/76299467

目录(?)

[+]

Spring的一个核心功能是IOC，就是将Bean初始化加载到容器中，Bean是如何加载到容器的，可以使用Spring注解方式或者Spring XML配置方式。

Spring注解方式减少了配置文件内容，更加便于管理，并且使用注解可以大大提高了开发效率！

下面安装分类讲解Spring中常用的一些注解。

一： 组件类注解

- 1

思考：spring怎么知道应该哪些Java类当初bean类处理？
- 2

答案：使用配置文件或者注解的方式进行标识需要处理的java类！

1、注解类介绍

- @Component：标准一个普通的spring Bean类。
- @Repository：标注一个DAO组件类。
- @Service：标注一个业务逻辑组件类。
- @Controller：标注一个控制器组件类。

这些都是注解在平时的开发过程中出镜率极高，@Component、@Repository、@Service、@Controller实质上属于同一类注解，用法相同，功能相同，区别在于标识组件的类型。**@Component可以代替@Repository、@Service、@Controller，因为这三个注解是被@Component标注的。如下代码**

```
1 @Target({ElementType.TYPE})
2 @Retention(RetentionPolicy.RUNTIME)
3 @Documented
4 @Component
5 public @interface Controller {
6     String value() default "";
7 }
```



文章分类

- Java基础 (17)
- Java集合 (11)
- java Io流 (8)
- java并发编程 (7)
- Spring&SpringMVC (18)
- JS&Jquery&Ajax (14)
- javaweb (13)
- Linux学习 (16)
- WebService (1)
- tomcat学习 (11)
- Quartz学习 (10)
- mybatis (4)
- maven学习 (6)
- Redis (10)
- 日志记录 (1)
- 代码管理工具 (7)
- 数据库 (4)
- Nginx&OpenResty (10)
- 分布式相关技术 (8)
- 软件设计&软件质量 (2)
- 开发工具 (7)
- 阿飞的世界 (14)

文章存档

- 2018年3月 (10)
- 2018年1月 (3)
- 2017年12月 (3)
- 2017年11月 (6)
- 2017年10月 (9)

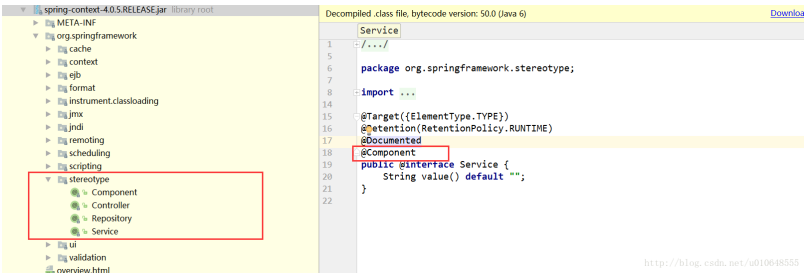
展开

阅读排行

- Spring学习——Spring中定时... (35114)
- get请求中url传参中文乱码问... (31705)
- Jquery 实现表单提交按钮变灰... (20304)
- Quartz学习——Quartz大致介... (19790)
- Quartz学习——Spring和Quartz... (17243)
- java基础—java.io.FileNotFoun... (16125)
- Tomcat学习—Tomcat7 修改/we... (14696)
- git学习——设置gitlab、github... (13825)
- java基础—java获取图片的大... (13431)
- Tomcat学习—Tomcat的tomcat-... (11946)

最新评论

- java集合系列——java集合概...
天道酬勤_v587 : 应该是实现的是Iterable接口不是iterat...
terator接口
- Spring学习——Spring中...



2、举例详解

(1) 当一个组件代表数据访问层（DAO）的时候，我们使用@Repository进行注解，如下

```
1 @Repository
2 public class HappyDaoImpl implements HappyDao{
3     private final static Logger LOGGER = LoggerFactory.getLogger(HappyDaoImpl .class);
4     public void club(){
5         //do something ,like drinking and singing
6     }
7 }
```

(2) 当一个组件代表业务层时，我们使用@Service进行注解，如下

```
1 @Service(value="goodClubService")
2 //使用@Service注解不加value ,默认名称是clubService
3 public class ClubServiceImpl implements ClubService {
4     @Autowired
5     private ClubDao clubDao;
6
7     public void doHappy(){
8         //do some Happy
9     }
10 }
```

(3) 当一个组件作为前端交互的控制层，使用@Controller进行注解，如下

```
1 @Controller
2 public class HappyController {
3     @Autowired //下面进行讲解
4     private ClubService clubService;
5
6     // Control the people entering the Club
7     // do something
8 }
9 /*Controller相关的注解下面进行详细讲解，这里简单引入@Controller*/
```

3、总结注意点

1、被注解的java类当做Bean实例，Bean实例的名称默认是Bean类的首字母小写，其他部分不变。@Service也可以自定义Bean名称，但是必须是唯一的！

2、尽量使用对应组件注解的类替换@Component注解，在spring未来的版本中，@Controller，@Service，@Repository会携带更多语义。并且便于开发和维护！

每天都在变得更好的阿飞 : [reply]tiandaochq[/reply]
] spring的

Spring学习——Spring中...
tiandaochq : <!-- 开启这个配置, spring才能识别
@Scheduled注解 --&am...

java基础—java获取图片的大...
ke_new : 谢谢啦, 做图片导出, 刚好用到这个~

HTML与XML的区别(转)
任蒙蒙-Angel : HTML是与显示信息相关的, XML
则是与描述信息相关的。

Quartz学习——SSMM(Sp...
每天都在变得更好的阿飞 : [reply]chaoqunwu[/repl
y] 你好, 不需要 quartz.xml文件 !

Quartz学习——SSMM(Sp...
movie__wang : [reply]chaoqunwu[/reply] 好久了 都
忘了

Quartz学习——SSMM(Sp...
chaoqunwu : [reply]wangdianying[/reply] 不缺少qua
rtz.xml文件吗

Quartz学习——SSMM(Sp...
chaoqunwu : 源码中是不是mapper包下是不是缺少
quartz.xml文件, 任务部分数据跟数据库没连接啊

Redis——windows环境安...
西小米点点 : 问题已经解决了

序号	名称	品牌	产地	规格	功率	电压	频率	重量	尺寸	价格
1	联想	联想	中国	15.6英寸	15.6英寸	15.6英寸	15.6英寸	15.6英寸	15.6英寸	15.6英寸
2	戴尔	戴尔	中国	15.6英寸	15.6英寸	15.6英寸	15.6英寸	15.6英寸	15.6英寸	15.6英寸
3	惠普	惠普	中国	15.6英寸	15.6英寸	15.6英寸	15.6英寸	15.6英寸	15.6英寸	15.6英寸
4	华硕	华硕	中国	15.6英寸	15.6英寸	15.6英寸	15.6英寸	15.6英寸	15.6英寸	15.6英寸
5	小米	小米	中国	15.6英寸	15.6英寸	15.6英寸	15.6英寸	15.6英寸	15.6英寸	15.6英寸
6	华为	华为	中国	15.6英寸	15.6英寸	15.6英寸	15.6英寸	15.6英寸	15.6英寸	15.6英寸
7	魅族	魅族	中国	15.6英寸	15.6英寸	15.6英寸	15.6英寸	15.6英寸	15.6英寸	15.6英寸
8	锤子	锤子	中国	15.6英寸	15.6英寸	15.6英寸	15.6英寸	15.6英寸	15.6英寸	15.6英寸
9	乐视	乐视	中国	15.6英寸	15.6英寸	15.6英寸	15.6英寸	15.6英寸	15.6英寸	15.6英寸
10	小米	小米	中国	15.6英寸	15.6英寸	15.6英寸	15.6英寸	15.6英寸	15.6英寸	15.6英寸
11	华为	华为	中国	15.6英寸	15.6英寸	15.6英寸	15.6英寸	15.6英寸	15.6英寸	15.6英寸
12	魅族	魅族	中国	15.6英寸	15.6英寸	15.6英寸	15.6英寸	15.6英寸	15.6英寸	15.6英寸
13	锤子	锤子	中国	15.6英寸	15.6英寸	15.6英寸	15.6英寸	15.6英寸	15.6英寸	15.6英寸
14	乐视	乐视	中国	15.6英寸	15.6英寸	15.6英寸	15.6英寸	15.6英寸	15.6英寸	15.6英寸
15	小米	小米	中国	15.6英寸	15.6英寸	15.6英寸	15.6英寸	15.6英寸	15.6英寸	15.6英寸

联系我们



请扫描二维码联系客服

webmaster@csdn.net

400-660-0108

QQ客服 客服论坛

关于 招聘 广告服务 百度

©1999-2018 CSDN版权所有

京ICP证09002463号

经营性网站备案信息

网络110报警服务

中国互联网举报中心

北京互联网违法和不良信息举报中心

3、指定了某些类可作为Spring Bean类使用后, 最好还需要让spring搜索指定路径, 在Spring配置文件加入如下配置:

```
1 <!-- 自动扫描指定包及其子包下的所有Bean类 -->
2 <context:component-scan base-package="org.springframework.*"/>
```

二: 装配bean时常用的注解

1、注解介绍

- @Autowired: 属于Spring 的org.springframework.beans.factory.annotation包下,可用于为类的属性、构造器、方法进行注值
- @Resource: 不属于spring的注解, 而是来自于JSR-250位于java.annotation包下, 使用该annotation为目标bean指定协作者Bean。
- @PostConstruct 和 @PreDestroy 方法 实现初始化和销毁bean之前进行的操作

2、举例说明

(1) : @Autowired

```
1
2 @Target({ElementType.CONSTRUCTOR, ElementType.FIELD, ElementType.METHOD, ElementType.ANNOTATIO
3 @Retention(RetentionPolicy.RUNTIME)
4 @Documented
5 public @interface Autowired {
6     boolean required() default true;
7 }
8
9 @Controller
10 public class HappyController {
11     @Autowired //默认依赖的ClubDao 对象 (Bean) 必须存在
12     //@Autowired(required = false) 改变默认方式
13     @Qualifier("goodClubService")
14     private ClubService clubService;
15
16     // Control the people entering the Club
17     // do something
18 }
19 }
```

(2) : @Resource

```
1 @Target({TYPE, FIELD, METHOD})
2 @Retention(RUNTIME)
3 public @interface Resource {
4     String name() default "";
5     Class type() default java.lang.Object.class;
6     ...
7
8 public class AnotationExp {
9
10     @Resource(name = "HappyClient")
11     private HappyClient happyClient;
12
13     @Resource(type = HappyPlayAno .class)
14     private HappyPlayAno happyPlayAno;
15 }
```

3、总结

(1)：相同点

@Resource的作用相当于@Autowired，均可标注在字段或属性的setter方法上。

(2)：不同点

a：提供方 @Autowired是Spring的注解，@Resource是javax.annotation注解，而是来自于JSR-250，J2EE提供，需要JDK1.6及以上。

b：注入方式 @Autowired只按照Type注入；@Resource默认按Name自动注入，也提供按照Type注入；

c：属性

@Autowired注解可用于为类的属性、构造器、方法进行注值。默认情况下，其依赖的对象必须存在（bean可用），如果需要改变这种默认方式，可以设置其required属性为false。

还有一个比较重要的点就是，@Autowired注解默认按照类型装配，如果容器中包含多个同一类型的Bean，那么启动容器时会报找不到指定类型bean的异常，解决办法是结合@Qualified注解进行限定，指定注入的bean名称。

@Resource有两个中重要的属性：name和type。name属性指定byName，如果没有指定name属性，当注解标注在字段上，即默认取字段的名称作为bean名称寻找依赖对象，当注解标注在属性的setter方法上，即默认取属性名作为bean名称寻找依赖对象。

需要注意的是，@Resource如果没有指定name属性，并且按照默认的名称仍然找不到依赖对象时，@Resource注解会回退到按类型装配。但一旦指定了name属性，就只能按名称装配了。

d：@Resource注解的使用性更为灵活，可指定名称，也可以指定类型；@Autowired注解进行装配容易抛出异常，特别是装配的bean类型有多个的时候，而解决的办法是需要增加@Qualified进行限定。

Spring中 @Autowired注解与@Resource注解的区别

注意点：使用@Resource也要注意添加配置文件到Spring，如果没有配置

```
1 <context:component-scan>
2 <!--<context:component-scan>的使用，是默认激活<context:annotation-config>功能-->
```

需要配置

```
1 <context:annotation-config/>
```

三：@Component vs @Configuration and @Bean

1、简单介绍

Spring的官方团队说@Component可以替代 @Configuration注解，事实上我们看源码也可以发现看到，如下

```
1 @Target({ElementType.TYPE})
2 @Retention(RetentionPolicy.RUNTIME)
3 @Documented
4 @Component //看这里!!!
5 public @interface Configuration {
6     String value() default "";
```

虽然说可以替代但是两个注解之间还是有区别的！

Bean注解主要用于方法上，有点类似于工厂方法，当使用了@Bean注解，我们可以连续使用多种定义bean时用到的注解，譬如用@Qualifier注解定义工厂方法的名称，用@Scope注解定义该bean的作用域范围，譬如如

singleton还是prototype等。

Spring 中新的 Java 配置支持的核心就是@Configuration 注解的类。这些类主要包括 @Bean 注解的方法来为 Spring 的 IoC 容器管理的对象定义实例，配置和初始化逻辑。

使用@Configuration 来注解类表示类可以被 Spring 的 IoC 容器所使用，作为 bean 定义的资源。

```
1 @Configuration
2 public class AppConfig {
3     @Bean
4     public MyService myService() {
5         return new MyServiceImpl();
6     }
7 }
```

这和 Spring 的 XML 文件中的非常类似

```
1 <beans>
2     <bean id="myService" class="com.acme.services.MyServiceImpl"/>
3 </beans>
```

@Bean 注解扮演了和元素相同的角色。

2、举例说明@Component 和 @Configuration

```
1 @Configuration
2 public static class Config {
3
4     @Bean
5     public SimpleBean simpleBean() {
6         return new SimpleBean();
7     }
8
9     @Bean
10    public SimpleBeanConsumer simpleBeanConsumer() {
11        return new SimpleBeanConsumer(simpleBean());
12    }
13 }
```

```
1 @Component
2 public static class Config {
3
4     @Bean
5     public SimpleBean simpleBean() {
6         return new SimpleBean();
7     }
8
9     @Bean
10    public SimpleBeanConsumer simpleBeanConsumer() {
11        return new SimpleBeanConsumer(simpleBean());
12    }
13 }
```

第一个代码正常工作，正如预期的那样，**SimpleBeanConsumer**将会得到一个单例SimpleBean的链接。第二个配置是完全错误的，因为spring会创建一个SimpleBean的单例bean，但是SimpleBeanConsumer将获得另一个SimpleBean实例，该实例是spring上下文控件之外的。

3、原因总结

使用@ configuration，所有标记为@ bean的方法将被包装成一个CGLIB包装器，它的工作方式就好像是这个方法的第一个调用，那么原始方法的主体将被执行，最终的对象将在spring上下文中注册。所有进一步的调用只返回从上下文检索的bean。

在上面的第二个代码块中，新的SimpleBeanConsumer(simpleBean())只调用一个纯java方法。为了纠正第二个代码块，我们可以这样做

```
1 @Component
2 public static class Config {
3     @Autowired
4     SimpleBean simpleBean;
5
6     @Bean
7     public SimpleBean simpleBean() {
8         return new SimpleBean();
9     }
10
11     @Bean
12     public SimpleBeanConsumer simpleBeanConsumer() {
13         return new SimpleBeanConsumer(simpleBean);
14     }
15 }
```

[Spring @Configuration vs @Component](#)

基本概念：@Configuration 和@Bean

四：spring MVC模块注解

1、web模块常用到的注解

- @Controller：表明该类会作为与前端作交互的控制层组件，通过服务接口定义的提供访问应用程序的一种行为，解释用户的输入，将其转换成一个模型然后将试图呈献给用户。

```
1 @Controller
2 public class HappyController {
3     //do something
4     ...
5 }
```

Spring MVC 使用 @Controller 定义控制器，它还允许自动检测定义在类路径下的组件（配置文件中配置扫描路径）并自动注册。

- @RequestMapping：这个注解用于将url映射到整个处理类或者特定的处理请求的方法。可以只用通配符！

```
1 @Controller
2 @RequestMapping("/happy")
3 public class HappyController {
4
5     @Autowired
6     private HappyService happyService;
7
8     @RequestMapping("/hello/*")
9     public void sayHello(){
```

```
10 //请求为 /happy/hello/* 都会进入这个方法!
11 //例如: /happy/hello/123 /happy/hello/adb
12 //可以通过get/post 请求
13 }
14 @RequestMapping(value="/haha",method=RequestMethod.GET)
15 public void sayHaHa(){
16 //只能通过get请求
17 }
18 ...
19 }
```

@RequestMapping 既可以作用在类级别，也可以作用在方法级别。当它定义在类级别时，标明该控制器处理所有的请求都被映射到 /favsoft 路径下。@RequestMapping中可以使用 method 属性标记其所接受的方法类型，如果不指定方法类型的话，可以使用 HTTP GET/POST 方法请求数据，但是一旦指定方法类型，就只能使用该类型获取数据。

- @RequestParam：将请求的参数绑定到方法中的参数上，有required参数，默认情况下，required=true，也就是改参数必须要传。如果改参数可以传可不传，可以配置required=false。

```
1 @RequestMapping("/happy")
2 public String sayHappy(
3     @RequestParam(value = "name", required = false) String name,
4     @RequestParam(value = "age", required = true) String age) {
5     //age参数必须传，name可传可不传
6     ...
7 }
```

- @PathVariable：该注解用于方法修饰方法参数，会将修饰的方法参数变为可供使用的uri变量（可用于动态绑定）。

```
1 @RequestMapping(value="/happy/{dayid}",method=RequestMethod.GET)
2 public String findPet(@PathVariable String dayid, Model mode) {
3     //使用@PathVariable注解绑定 {dayid} 到String dayid
4 }
```

@PathVariable中的参数可以是任意的简单类型，如int, long, Date等等。Spring会自动将其转换成合适的类型或者抛出 TypeMismatchException异常。当然，我们也可以注册支持额外的数据类型。

@PathVariable支持使用正则表达式，这就决定了它的超强大属性，它能在路径模板中使用占位符，可以设定特定的前缀匹配，后缀匹配等自定义格式。

- @RequestBody：@RequestBody是指方法参数应该被绑定到HTTP请求Body上。

```
1 @RequestMapping(value = "/something", method = RequestMethod.PUT)
2 public void handle(@RequestBody String body, @RequestBody User user){
3     //可以绑定自定义的对象类型
4 }
```

- @ResponseBody：@ResponseBody与@RequestBody类似，它的作用是将返回类型直接输入到HTTP response body中。

@ResponseBody在输出JSON格式的数据时，会经常用到。

```
1 @RequestMapping(value = "/happy", method = RequestMethod.POST)
2 @ResponseBody
```

```
3 public String helloWorld() {
4     return "Hello World";//返回String类型
5 }
```

- @RestController：控制器实现了REST的API，只为服务于JSON，XML或其它自定义的类型内容，@RestController用来创建REST类型的控制器，与@Controller类型。@RestController就是这样一种类型，它避免了你重复的写@RequestMapping与@ResponseBody。
- @ModelAttribute：@ModelAttribute可以作用在方法或方法参数上，当它作用在方法上时，标明该方法的目的是添加一个或多个模型属性（model attributes）。该方法支持与@RequestMapping一样的参数类型，但并不能直接映射成请求。控制器中的@ModelAttribute方法会在@RequestMapping方法调用之前而调用。

@ModelAttribute方法有两种风格：一种是添加隐形属性并返回它。另一种是该方法接受一个模型并添加任意数量的模型属性。用户可以根据自己的需要选择对应的风格。

五：Spring事务模块注解

1、常用到的注解

在处理dao层或service层的事务操作时，譬如删除失败时的回滚操作。使用@Transactional 作为注解，但是需要在配置文件激活

```
1 <!-- 开启注解方式声明事务 -->
2 <tx:annotation-driven transaction-manager="transactionManager" />
```

2、举例

```
1 @Service
2 public class CompanyServiceImpl implements CompanyService {
3     @Autowired
4     private CompanyDAO companyDAO;
5
6     @Transactional(propagation = Propagation.REQUIRED, readOnly = false, rollbackFor = Exception
7     public int deleteByName(String name) {
8
9         int result = companyDAO.deleteByName(name);
10        return company;
11    }
12    ...
13 }
```

3、总结

事务的传播机制和隔离机制比较重要！

事务传播行为类型	说明
PROPAGATION_REQUIRED	如果当前没有事务，就新建一个事务，如果已经存在一个事务中，加入到这个事务中。这是最常见的选择。
PROPAGATION_SUPPORTS	支持当前事务，如果当前没有事务，就以非事务方式执行。
PROPAGATION_MANDATORY	使用当前的事务，如果当前没有事务，就抛出异常。
PROPAGATION_REQUIRES_NEW	新建事务，如果当前存在事务，把当前事务挂起。

PROPAGATION_NOT_SUPPORTED	以非事务方式执行操作，如果当前存在事务，就把当前事务挂起。
PROPAGATION_NEVER	以非事务方式执行，如果当前存在事务，则抛出异常
PROPAGATION_NESTED	如果当前存在事务，则在嵌套事务内执行。如果当前没有事务，则执行与PROPAGATION_REQUIRED类 似的操作

一图学习 Spring事务传播性

readOnly : 事务的读写属性，取true或者false，true为只读、默认为false
rollbackFor : 回滚策略，当遇到指定异常时回滚。譬如上例遇到异常就回滚
timeout （补充的）: 设置超时时间，单位为秒
isolation : 设置事务隔离级别， 枚举类型，一共五种

类型	说明
DEFAULT	采用数据库默认隔离级别
READ_UNCOMMITTED	读未提交的数据（会出现脏读取）
READ_COMMITTED	读已提交的数据（会出现幻读，即前后两次读的不一樣）
REPEATABLE_READ	可重复读，会出现幻读
SERIALIZABLE 串行化	（对资源消耗较大，一般不使用）

透彻的掌握 Spring 中@Transactional 的使用

Spring事务配置及事务的传播性与隔离级别详解

参考博文

spring 常用注解
详解Spring MVC 4常用的那些注解

欢迎访问我的csdn博客，我们一同成长！
“不管做什么，只要坚持下去就会看到不一样！在路上，不卑不亢!”
博客首页：<http://blog.csdn.net/u010648555>

- [上一篇](#) [Nginx学习——Nginx基本配置](#)
- [下一篇](#) [MyBatis中Like语句使用方式](#)

世界上最好的语言，PHP凭什么这么“说”

课程包含PHP零基础入门、PHP基础编程、数据库、项目实战、linux、框架开发到高级+企业项目实战，深入浅出地剖析和分解了PHP企业级开发项目在实际工作中的应用。

18502

查看更多>>



app外包公司



二维码追溯系统



OA办公系统



近视恢复视力的方



丰胸有什么危害



免费云主机试用



近视眼视力恢复

您还没有登录,请[\[登录\]](#)或[\[注册\]](#)

springMVC的常用注解有哪些?



wuruijiang

2018年01月04日 14:26

1117

1、@Controller @Controller 用于标记在一个类上,使用它标记的类就是一个SpringMVC Controller 对象。分发处理器将会扫描使用了该注解的类的方法,...

Spring常用注解汇总



jeffleo

2017年02月16日 21:18

839

@Controller: 注册一个bean到spring (一般用于Controller层) @Service:注册一个bean到spring (一般用于service层) @Repository: .
..

程序员为什么英语比数学更重要? 如何学习?

老司机教你一个数学公式秒懂天下英语

广告



Spring常用注解汇总



hj7jay

2017年02月21日 09:37

380

使用注解前需要开启自动扫描功能,其中base-package为需要扫描的包(含子包)。1声明Bean注解 1 @Component 注解在类上,可以作用在任何层次。...

注解和注释以及Spring和SpringMVC常用的注解



u012661010

2017年06月19日 10:46

3852

1.两者区别注解: 参与代码编译,以@开头的。它是给应用程序看的,单独使用注解毫无意义,一定要跟工具一起使用.这个所谓的工具实际就是能读懂注解的应用程序 注释: 对代码没有影响。对代码起到解释、说...

Spring的几个注解解释



HistoryCreator

2017年05月25日 20:27

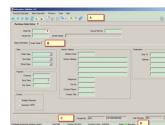
223

@Component==@Resource~~=@Autowired~~=@Inject @Resource默认是byName注入 @Autowired默认是byType@Repository(...

国内常用的erp软件

常用erp

百度广告



史上最全的java spring注解,没有之一



achenyuan

2017年05月27日 23:23

4367

注解是个好东西,但好东西我们也是看见过,整理过,理解过,用过才知道好。不求我们每个都记住,但求保有印象,在需要的时候能提取出来再查找相关资料,平时工作就不会显得那么被动了。1.@Configurati...

总结一下spring框架中的各种注解



hehaiwei1234

2017年02月21日 14:23

1490

@Autowired和@Resource等注解是将Spring容器中的bean注入到属性,而@Component等注解是将bean放入Spring容器中管理。@Au

towired s...

Spring注解详解



xyh820

2012年02月28日 19:04

281932

概述 注解配置相对于 XML 配置具有很多的优势： 它可以充分利用 Java 的反射机制获取类结构信息，这些信息可以有效减少配置的工作。如使用 JPA 注解配置 ORM 映射时，我们就不需要指...

Spring 注解总结



wangshfa

2013年08月02日 08:59

40009

声明：这是转载的。内容根据网上资料整理。相关链接：http://www.360doc.com/content/10/11/18/16/2371584_70449913.shtml [http...](http://...)

spring常用的一些注解以及注解注入总结



AILAN_

2015年08月15日 15:10

32451

常用的spring注解有如下几种：@Controller @Service @Autowired @RequestMapping @RequestParam @ModelAttribute @Ca...

对于程序员来说，英语到底多重要

不背单词和语法，一个公式学好英语



Spring 注解 小例子



2013年11月15日 21:34 16.18MB

下载

深入理解spring中的各种注解



wconvey

2017年01月05日 14:41

1286

转自：点击打开链接 Spring中的注解大概可以分为两大类：1) spring的bean容器相关的注解，或者说bean工厂相关的注解；2) springmvc相关的注解。spring的be...

JSTL的c:forEach标签模拟for循环



ikaraide

2013年12月22日 19:48

7623

使用c:forEach标签输出一维数组 \${ele } days为通过request.setAttribute("days", days);放入request的一维数组，取出后var的值ele表...

Spring-各组件注释以及作用



cuigaochong

2015年11月02日 16:29

2459

1、@controller 控制器（注入服务）2、@service 服务（注入dao）3、@repository dao（实现dao访问）4、@component（把普通pojo实例化到spr...

高端网站设计

高端网站美工设计 网站定制开发

百度广告



spring常用注解使用讲解



zhangjiekuan

2016年07月16日 19:57

6246

spring常用注解使用讲解 本文讲述spring的几个常用的注解 @RequestMapping @RequestParam @ResponseBody @RequestBody...

jstl标签使用-- 关于c for 循环



soconglin

2011年09月01日 16:54

6339

Action : request.setAttribute("patentInfo", (Map)map.get("patentInfoBean")); jsp展示： "patentbe...

Hibernate的事务管理



lifaming15 2008年06月19日 12:25 26440

Hibernate的事务管理事务（Transaction）是工作中的基本逻辑单位，可以用于确保数据库能够被正确修改，避免数据只修改了一部分而导致数据不完整，或者在修改时受到用户干扰。作为一名软件设计师...

Hibernate之三态篇



liu_yujie2011com 2014年10月31日 14:34 2898

一、概况 （一）瞬时状态（临时态） 在对象中如果对象刚被创建但没有被持久化的话就是瞬时态 特点： （1）不和 Session 实例关联 （2）在数据库...

50万码农评论：英语对于程序员有多重要！

老司机教你一个数学公式秒懂天下英语



【Spring】Spring MVC原理及配置详解



jianyuersheng 2016年04月27日 10:14 190620

【Spring】Spring MVC原理及配置1.Spring MVC概述：Spring MVC是Spring提供的一个强大而灵活的web框架。借助于注解，Spring MVC提供了几乎是POJO的开...

JSTL实现for(int i)循环



zjw10wei321 2012年10月29日 10:01 12116

只要不填写items就行了