**36. 什么是基于Java的Spring注解配置? 给一些注解的例子.**

**37. 什么是基于注解的容器配置?**

**38. 怎样开启注解装配？**

**39. @Required 注解**

**40. @Autowired 注解**

**41. @Qualifier 注解**

**42.在Spring框架中如何更有效地使用JDBC?**

**43. JdbcTemplate**

**44. Spring对DAO的支持**

**45. 使用Spring通过什么方式访问Hibernate?**

**46. Spring支持的ORM**

**47.如何通过HibernateDaoSupport将Spring和Hibernate结合起来？**

**48. Spring支持的事务管理类型**

**49. Spring框架的事务管理有哪些优点？**

**50. 你更倾向用那种事务管理类型？**

|  |  |
| --- | --- |
|  |  |

**答案解析**ID:javaxxf

**36. 什么是基于Java的Spring注解配置? 给一些注解的例子.**

基于Java的配置，允许你在少量的Java注解的帮助下，进行你的大部分Spring配置而非通过XML文件。

以@Configuration 注解为例，它用来标记类可以当做一个bean的定义，被Spring IOC容器使用。另一个例子是@Bean注解，它表示此方法将要返回一个对象，作为一个bean注册进Spring应用上下文。

**37. 什么是基于注解的容器配置?**

相对于XML文件，注解型的配置依赖于通过字节码元数据装配组件，而非尖括号的声明。

开发者通过在相应的类，方法或属性上使用注解的方式，直接组件类中进行配置，而不是使用xml表述bean的装配关系。

**38. 怎样开启注解装配？**

注解装配在默认情况下是不开启的，为了使用注解装配，我们必须在Spring配置文件中配置<context:annotation-config/>元素。

**39. @Required 注解**

这个注解表明bean的属性必须在配置的时候设置，通过一个bean定义的显式的属性值或通过自动装配，若@Required注解的bean属性未被设置，容器将抛出BeanInitializationException。

**40. @Autowired 注解**

@Autowired 注解提供了更细粒度的控制，包括在何处以及如何完成自动装配。它的用法和@Required一样，修饰setter方法、构造器、属性或者具有任意名称和/或多个参数的PN方法。

**41. @Qualifier 注解**

当有多个相同类型的bean却只有一个需要自动装配时，将@Qualifier 注解和@Autowire 注解结合使用以消除这种混淆，指定需要装配的确切的bean。

**42.在Spring框架中如何更有效地使用JDBC?**

使用SpringJDBC 框架，资源管理和错误处理的代价都会被减轻。所以开发者只需写statements 和 queries从数据存取数据，JDBC也可以在Spring框架提供的模板类的帮助下更有效地被使用，这个模板叫JdbcTemplate （例子看43）

**43. JdbcTemplate**

JdbcTemplate 类提供了很多便利的方法解决诸如把数据库数据转变成基本数据类型或对象，执行写好的或可调用的数据库操作语句，提供自定义的数据错误处理。

**44. Spring对DAO的支持**

Spring对数据访问对象（DAO）的支持旨在简化它和数据访问技术如JDBC，Hibernate or JDO结合使用。这使我们可以方便切换持久层。编码时也不用担心会捕获每种技术特有的异常。

**45. 使用Spring通过什么方式访问Hibernate?**

在Spring中有两种方式访问Hibernate：

* 控制反转 HibernateTemplate和 Callback。
* 继承HibernateDAOSupport提供一个AOP 拦截器。

**46. Spring支持的ORM**

Spring支持以下ORM：

* Hibernate
* iBatis
* JPA (Java Persistence API)
* TopLink
* JDO (Java Data Objects)
* OJB

**47.如何通过HibernateDaoSupport将Spring和Hibernate结合起来？**

用Spring的 SessionFactory 调用 LocalSessionFactory。集成过程分三步：

* 配置the HibernateSessionFactory。
* 继承HibernateDaoSupport实现一个DAO。
* 在AOP支持的事务中装配。

**48. Spring支持的事务管理类型**

Spring支持两种类型的事务管理：

* **编程式事务管理：**这意味你通过编程的方式管理事务，给你带来极大的灵活性，但是难维护。
* **声明式事务管理：**这意味着你可以将业务代码和事务管理分离，你只需用注解和XML配置来管理事务。

**49. Spring框架的事务管理有哪些优点？**

* 它为不同的事务API 如 JTA，JDBC，Hibernate，JPA 和JDO，提供一个不变的编程模式。
* 它为编程式事务管理提供了一套简单的API而不是一些复杂的事务API如它支持声明式事务管理。
* 它和Spring各种数据访问抽象层很好得集成。

**50. 你更倾向用那种事务管理类型？**

大多数Spring框架的用户选择声明式事务管理，因为它对应用代码的影响最小，因此更符合一个无侵入的轻量级容器的思想。声明式事务管理要优于编程式事务管理，虽然比编程式事务管理（这种方式允许你通过代码控制事务）少了一点灵活性。