

1，关于 Spring MVC 的核心控制器 DispatcherServlet 的作用，以下说法错误的是（ ）

- A，它负责接收 HTTP 请求
- B，加载配置文件
- C，实现业务操作
- D，初始化上下应用对象 ApplicationContext

正确答案：C

SpringMVC 是 Spring 中的模块，它实现了 mvc 设计模式，首先用户发起请求，请求到达 SpringMVC 的前端控制器(DispatcherServlet)，前端控制器根据用户的 url 请求处理器映射器查找匹配该 url 的 handle，并返回一个执行链，前端控制器再请求处理器适配器调用相应的 handle 进行处理并返回给前端控制器一个 ModelAndView，前端控制器再请求视图解析器对返回的逻辑视图进行解析，最后前端控制器将返回的视图进行渲染，并把数据装入到 request 域，返回给用户。

DispatcherServlet 作为 SpringMVC 的前端控制器，负责接收用户的请求，并根据用户的请求返回相应的视图给用户。实现业务在 service 层，所以 c 答案错误。

2，下面有关 spring 的依赖注入，说法错误的是（ ）

- A，依赖注入通常有如下两种：设值注入和构造注入：
- B，构造注入可以在构造器中决定依赖关系的注入顺序，优先依赖的优先注入
- C，当设值注入与构造注入同时存在时，先执行构造注入，再执行设值注入
- D，设值注入是指 IoC 容器使用属性的 setter 方法来注入被依赖的实例。这种注入方式比较简单、直观

正确答案：C

使用构造函数依赖注入时，Spring 保证一个对象所有依赖的对象先实例化后，才实例化这个对象。使用 set 方法依赖注入时，Spring 首先实例化对象，然后才实例化所有依赖的对象。

当设值注入与构造注入同时存在时，先执行设值注入，再执行构造注入。

3，将元数据配置到 Spring 容器，下面哪个方法是不正确的（ ）

- A，通过编组与解组对象
- B，注解基础配置
- C，Java 基础配置
- D，XML 基础配置

正确答案：A

有三种方式向 Spring 容器提供元数据:1，XML 配置文件；2，基于注解配置；3，基于 Java 的配置，死概念记住即可。

4, 下面哪个不是 Spring 的注解 ()

- A , @Aspect
- B , @Service
- C , @Component
- D , @Controller
- E , @View

正确答案 : E

考察对 spring 的熟悉程度, 除 E 选项之外都是较常用注解。

5, 下面有关 SPRING 的事务传播特性, 说法错误的是 ()

- A , 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
- B , PROPAGATION_REQUIRED : 支持当前事务, 如果当前没有事务, 就抛出异常
- C , PROPAGATION_REQUIRES_NEW : 新建事务, 如果当前存在事务, 把当前事务挂起
- D , PROPAGATION_NESTED : 支持当前事务, 新增 Savepoint 点, 与当前事务同步提交或回滚

正确答案 : B

事务的传播属性介绍:

PROPAGATION_REQUIRED --支持当前事务, 如果当前没有事务, 就新建一个事务。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--支持当前事务, 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。

6, 下列关于 Spring 特性中 IoC 描述错误的是 ()

- A , IoC 就是指程序之间的关系由程序代码直接操控
- B , 所谓“控制反转”是指控制权由应用代码转到外部容器, 即控制权的转移
- C , IoC 将控制创建的职责搬进了框架中, 从应用代码脱离开来
- D , 使用 Spring 的 IoC 容器时只需指出组件需要的对象, 在运行时 Spring 的 IoC 容器会根据 XML 配置数据提供给它

正确答案：A

IOC，即控制反转（Inversion of Control，英文缩写为 IoC），是一个重要的面向对象编程的法则来削减计算机程序的耦合问题，也是轻量级的 Spring 框架的核心。控制反转一般分为两种类型，依赖注入（Dependency Injection，简称 DI）和依赖查找（Dependency Lookup）。