# 选择题

1. 下列关于SSM框架的整合说法错误的是（D）

A、Spring MVC与Spring之间不存在整合的问题。

B、SSM框架的整合就涉及到Spring与MyBatis的整合。

C、SSM框架的整合就涉及到Spring MVC与MyBatis的整合。

D、SSM框架的整合就涉及到Spring MVC与Spring之间的整合。

2. 下面属性中不属于association元素属性的是（C）

A、property

B、column

C、ofType

D、javaType

3. 下列哪一个不是数据库（D）

A、MySQL B、SQL Server C、Oralce D、POJO

4. 以下关于Spring核心容器相关说法错误的是（A）

A、Spring框架的所有功能都是通过其核心容器来实现的。

B、创建BeanFactory实例时，需要提供Spring所管理容器的详细配置信息，这些信息通常采用XML文件形式来管理。

C、ApplicationContext不仅包含了BeanFactory的所有功能，还添加了对国际化、资源访问、事件传播等方面的支持。

D、通常在Java项目中，会采用通过ClassPathXmlApplicationContext类来实例化ApplicationContext容器的方式，而在Web项目中，ApplicationContext容器的实例化工作会交由Web服务器来完成。

5. 以下有关Spring框架优点的说法正确的是（B）

A、Spring具有简单、可测试和松耦合等特点，从这个角度出发，Spring就是应用于任何Java应用的开发中。

B、Spring提供了对AOP的支持，它允许将一些通用任务，如安全、事务、日志等进行集中式处理，从而提高了程序的复用性。

C、Spring就是一个大工厂，可以将所有对象的创建和依赖关系的维护工作都交给Spring容器管理，杜绝了组件之间的耦合性。

D、Spring增加了Java EE开发中一些API的使用难度。

6. MyBatis的更新操作有关说法错误的是（D）

A、MyBatis的更新操作在映射文件中是通过配置update元素来实现的。

B、MyBatis的更新操作也需要进行事务提交。

C、MyBatis的更新操作执行了SqlSession的update()方法。

D、MyBatis的更新操作和添加操作时，只需要将insert()方法改为update()方法即可。

7. 使用ORM框架后，应用程序不再直接访问底层数据库，而是以( )的方式来操作持久化对象（C）

A、面向业务 B、Hibernate C、面向对象 D、MyBatis

8. Spring MVC是Spring提供的一个实现了( A )设计模式的轻量级Web框架

A、Web MVC B、Web C、单例 D、工厂

9.针对GET方式RESTful风格的请求http://localhost:8080/chapter14/user/1，后台方法接收参数的路径映射写法可行的是 ( C )

A、@RequestMapping("/user/{id}", method=RequestMethod.GET)

B、@RequestMapping(value='/user/id')

C、@GetMapping(value='/user/{id}')

D、@GetMapping(value='/user/{id}'，method=RequestMethod.GET)

10. 下面关于Java对象之间的关联关系描述正确的是( B )

A、一对一的关系就是在本类和对方类中定义同一个类型的对象。

B、一对多的关系就是一个A类类型对应多个B类类型的情况。

C、多对多的关系只需要在一方的类中引入另一方类型的集合。

D、多对多关联关系需要在本类中引入本类的集合。

1. 下面选项中，不属于整合SSM框架的配置文件的是( D )

A、db.properties

B、applicationContext.xml

C、mybatis-config.xml

D、struts.xml

2. 下面关于collection元素的描述正确的是（A）

A、MyBatis就是通过collection元素来处理一对多关联关系的。

B、collection元素的属性与association元素完全相同。

C、ofType属性与javaType属性对应，它用于指定实体对象中所有属性所包含的元素类型。

D、collection 元素只能使用嵌套查询方式。

3.在JavaEE中，以下哪个是用于创建web应用程序的基础组件（A）

A. Servlet B. JDBC C. ISP D. Spring

4. Spring中定义了一系列的注解，以下有关其常用的注解说明错误的是（A）

A、Autowired注解用于对Bean的属性变量、属性的setter方法及构造方法进行标注，配合对应的注解处理器完成Bean的自动配置工作，默认按照Bean的名称进行装配。

B、@Repository用于将数据访问层（DAO层）的类标识为Spring中的Bean

C、@Service通常作用在业务层（Service层），用于将业务层的类标识为Spring中的Bean。

D、@Controller通常作用在控制层（如Spring MVC的Controller），用于将控制层的类标识为Spring中的Bean。

5. 以下有关Spring的四个基础包说法正确的是（D）

A、Spring的四个基础包，它们分别对应Spring Web容器的四个模块。

B、Spring四个基础包有spring-core.RELEASE.jar、spring-beans-.RELEASE.jar、spring-context-.RELEASE.jar和spring-aop-.RELEASE.jar。

C、spring-context-.RELEASE.jar是所有应用都要用到的JAR包，它包含访问配置文件以及进行IoC或者DI操作相关的所有类。

D、spring-core.RELEASE.jar包含Spring框架基本的核心工具类，Spring其它组件都要用到这个包里的类，是其它组件的基本核心。

6. 有关MyBatis工作原理说法错误的是（C）

A、MyBatis的全局配置文件配置了MyBatis的运行环境等信息，其中主要内容是获取数据库连接。

B、MyBatis映射文件中配置了操作数据库的SQL语句，需要在MyBatis的全局配置文件中加载才能执行。

C、可以通过MyBatis的环境等配置信息构建会话对象SqlSession。

D、SqlSession对象，该对象中包含了执行SQL的所有方法。

7. 关于MyBatis模糊查询中进行SQL字符串拼接时，说法错误的是（C）

A、使用“${}”进行SQL字符串拼接时，无法防止SQL注入问题。

B、可以使用MySQL中的concat()函数进行字符串拼接。

C、使用MySQL中的concat()函数进行字符串拼接，也无法防止SQL注入。

D、使用MySQL中的concat()函数进行字符串拼接，导致数据库移植性变差。

8. 下列选项中，属于Spring MVC所必须的JAR包的是( B )

A、spring-web-4.3.6.RELEASE.jar

B、spring-webmvc-portlet-4.3.6.RELEASE.jar

C、spring-webmvc-4.3.6.RELEASE-javadoc.jar

D、spring-websocket-4.3.6.RELEASE.jar

9. JSON对象结构中，关键字key必须为。( C )类型

A、Object B、Array C、String D、Number

10. 下面关于@RequestMapping注解说法错误的是( D )

A、@RequestMapping注解的默认属性是value。

B、@RequestMapping注解的value属性值可以省略

C、@ RequestMapping注解的value属性名可以省略。

D、@RequestMapping注解的value属性必须标注。

# 判断题

1. 在MyBaits映射文件中使用set和if元素组合进行update语句动态SQL组装时，如果set元素内包含的内容都为空，则会出现SQL语法错误。( 正确 )

2. MyBatis的更新操作也需要进行事务提交。( 正确 )

3. 在MyBatis中if元素用于单条件分支判断，当某个if的test属性判断成立，然后跳过其他if元素进行动态SQL组装。( 错误 )

4. @Autowired注解需要标注在Service层的实现类上，这样才能实现依赖注入。( 错误 )

5. Spring中的AOP代理默认就是使用CGLIB代理的方式来实现的。( 错误 )

6. Spring的事务管理简化了传统的事务管理流程，并且在一定程度上减少了开发者的工作量。( 正确 )

7. @Responsebody注解与@RequestMapping注解配合使用时，页面中将可以获取到方法的返回值。( 正确 )

8. 在Spring MVC的配置文件中，视图解析器是必须配置的。( 错误 )

9. Spring MVC框架灵活性强，易于与其他框架集成。( 正确 )

10. 每次客户端请求singleton作用域的Bean时，Spring容器都会创建一个新的实例，并且不会管那些被配置成singleton作用域的Bean的生命周期。( 错误 )。

1. @Transactional注解主要是针对数据的增加、修改、删除和查询进行事务管理。( 错误 )

2. resultMap的子元素association和collection用于处理多表时的关联关系。( 正确 )

3. MyBaits中对原始SQL中的“where 1=1”的条件可以使用where与if元素组合进行替换。( 正确 )

4. Spring的声明式事务管理可以通过两种方式来实现，一种是基于XML的方式，另一种是基于Annotation的方式。( 正确 )

5. Spring与Spring MVC，Spring MVC与MyBatis需要相互整合。( 错误 )

6. Spring的事务管理简化了传统的事务管理流程，并且在一定程度上减少了开发者的工作量。( 正确 )

7. Web表现层：主要包括Spring MVC中的Controller类和JSP页面。( 正确 )

8. ModelAndView 是Model 和 View 的一个集合类型，其中Model 是一个Map类型的空间，其主要用来存储页面请求的返回数据。( 正确 )

9. Spring框架采用的是分层架构，它一系列的功能要素被分成20个模块。( 正确 )

10. 每次客户端请求singleton作用域的Bean时，Spring容器都会创建一个新的实例，并且不会管那些被配置成singleton作用域的Bean的生命周期。( 错误 )

# 简答题

1. 什么是MVC设计模式。

MVC是一种设计模式，将应用程序的输入、处理、输出流程按照Model、View、Controller的方式进行分离，并被分成三层——模型层、视图层、控制层。

2. mybatis编程步骤是什么样的。

创建SqlSessionFactory SqlSessionFactory sqlSessionFactory = new SqlSessionFactoryBuilder().build(inputStream); 2、通过SQLSessionFactory创建SqlSession SqlSession openSession = sqlSessionFactory.openSession(); 3、通过sqlsession执行数据库操作 4、调用session.commit()或session.rollback() 5、调用session.close()关闭会话。

3. 请简述SSM框架整合思路。

由于Spring MVC是Spring框架中的一个模块，所以Spring MVC与Spring之间不存在整合的问题，只要引入相应JAR包就可以直接使用。因此SSM框架的整合就只涉及到了Spring与MyBatis的整合，以及Spring MVC与MyBatis的整合。Spring与MyBatis框架的整合时，通过Spring实例化Bean，然后调用实例对象中的查询方法来执行MyBatis映射文件中的SQL语句的，如果能够正确查询出数据库中的数据，那么就可以认为Spring与MyBatis框架整合成功。

4. Spring依赖注入通常有哪些实现方式，列举并分别进行说明。

Spring的实现方式通常有两种，一种是属性setter方法注入，另一种是构造方法注入。属性setter方法注入是指IoC容器使用setter方法来注入被依赖的实例。通过调用无参构造器或无参静态工厂方法实例化Bean后，调用该Bean的setter方法，即可实现基于setter方法的依赖注入。构造方法注入是指IoC容器使用构造方法来注入被依赖的实例。基于构造方法的依赖注入通过调用带参数的构造方法来实现，每个参数代表着一个依赖。

5. 请简述Spring框架概念和优点。

Spring是分层的JavaSE/EE full-stack 轻量级开源框架，以IoC（Inverse of

EJB的臃肿、低效的开发模式。非侵入式设计；方便解耦、简化开发；支持AOP；支持声明式事务处理；方便程序测试；方便集成各种优秀框架；降低Java EE API的使用难度。

6. 为什么需要自定义数据绑定？

一般情况下，使用基本数据类型和POJO类型的参数数据已经能够满足需求，然而有些特殊类型的 参数是无法在后台进行直接转换的，例如日期数据就需要开发者自定义转换器（Converter）或 格式化（Formatter）来进行数据绑定。

7. #{}和${}的区别是什么。

#{}会将sql语句编译好并使用占位符？然后再进行取值 值转换成字符串 很大的程度防止sql注入 一般能用#{}的就别用${}。 ${}先取值 不做任何处理直接引用 然后再去编译sql语句 无法防止sql注入 一般用于传入数据库对象 例如传入表名。

9. 请简述Spring MVC的工作原理。

Spring MVC的工作原理可以概括为以下几个步骤：

（1）用户发送请求到前端控制器(DispatcherServlet)。（2）DispatcherServlet接收请求，通过处理器映射器(HandlerMapping)找到处理请求的处理器(Controller)。（3）DispatcherServlet将请求交给处理器适配器(HandlerAdapter)。（4）处理器适配器执行Controller，Controller执行完成后返回ModelAndView对象。（5）处理器适配器将Controller执行结果返回给DispatcherServlet。（6）DispatcherServlet将ModelAndView对象传递给视图解析器(ViewResolver)，得到视图对象。（7）DispatcherServlet将视图对象渲染到客户端。

5. JDBC编程有哪些不足之处，Mybatis是如何解决这些问题的。

1.jdbc连接数据库，会对每一次的连接创建一个线程，使用完毕后关闭线程，频繁的操作会造成系统资源的浪费从而影响性能;在mybatis的配置文件中配置数据库连接池。 2.jdbc的sql语句写在代码中造成代码不易维护,sql变动需要改变Java代码;mybatis将sql语句配置在mapper.xml文件中与Java代码分离,也可以接口映射（约定大于配置)。 3.jdbc的sql语句传参比较麻烦，sql语句的where条件不一定，占位符和参数一一对应；mybatis会自动将Java对象映射至sql语句。 4.jdbc对结果集解析比较麻烦,需要使用游标将结果集一条一条的遍历出来；mybatis自动将sql执行结果映射到Java对象。

10. 请对Spring的通知类型进行简要说明。

Spring中主要包含5种通知类型，分别是环绕通知，前置通知，后置通知，异常通知和引介通知。其中，环绕通知表示在目标方法执行前后实施增强，可以应用于日志、事务管理等功能。前置通知表示在目标方法执行前实施增强，可以应用于权限管理等功能。后置通知表示在目标方法执行后实施增强，可以应用于关闭流、上传文件、删除临时文件等功能。异常通知表示在方法抛出异常后实施增强，可以应用于处理异常记录日志等功能。引介通知表示在目标类中添加一些新的方法和属性，可以应用于修改老版本程序（增强类）。

# 程序填空题

采用CGLib实现动态代理实现AOP编程，MyAspect为切面类，其中，check\_Permissions方法前置增强，log方法为后置增强，请填写空白处代码。

|  |
| --- |
| public class CglibProxy implements MethodInterceptor {  public Object createProxy （Object target) {  Enhancer enhancer = new Enhancer();  enhancer.setSuperclass(target.getClass());  enhancer.setCallback(this);  return enhancer.create();  }  public Object intercept(Object proxy, Method method, Object[] args,  MethodProxy methodProxy) throws Throwable {  MyAspect myAspect = new MyAspect();  myAspect.check\_Permissions();  Object obj = methodProxy.invokeSuper(proxy, args);  myAspect.log();  return obj;  }  } |