第一次作业

J2EE**的概念：**J2EE是一套全然不同于传统应用开发的技术架构，包含许多组件，主要可简化且规范应用系统的开发与部署，进而提高可移植性、安全与再用价值。

J2EE核心是一组[技术规范](https://baike.so.com/doc/7885976-8160071.html" \t "_blank)与指南，其中所包含的各类组件、服务架构及技术层次，均有共同的标准及规格，让各种依循J2EE架构的不同平台之间，存在良好的[兼容性](https://baike.so.com/doc/6448046-6661728.html" \t "_blank)，解决过去企业后端使用的信息产品彼此之间无法兼容，企业内部或外部难以互通的窘境。

J2EE组件和"标准的" [Java](https://baike.so.com/doc/2886868-3046592.html)类的不同点在于:它被装配在一个J2EE应用中，具有固定的格式并遵守J2EE规范，由J2EE服务器对其进行管理。J2EE规范是这样定义J2EE组件的:客户端应用程序和applet是运行在客户端的组件;Java Servlet和Java Server Pages (JSP) 是运行在服务器端的Web组件;Enterprise Java Bean (EJB )组件是运行在服务器端的业务组件。

J2EE的三个框架：

1. Struts
2. struts是开源软件。是[MVC](https://baike.so.com/doc/5411388-5649495.html)经典设计模式中的一个经典产品。

性能：a.帮助我们减少在运用MVC设计模型来开发Web应用的时间，它最初是Jakarta项目中的一个子项目，并在2004年3月成为ASF的顶级项目。b.我们可以用struts来混合使用Servlets和JSP的[优点](https://baike.so.com/doc/5411537-5649647.html" \t "_blank)达到建立可扩展的应用它通过采用Java[Servlet](https://baike.so.com/doc/2354665-2490021.html)/[JSP](https://baike.so.com/doc/3533044-3716131.html)技术，实现了基于[Java EE](https://baike.so.com/doc/998397-1055441.html)Web应用的[MVC](https://baike.so.com/doc/5411388-5649495.html)设计模式的应用框架。

1. Hibernate

[Hibernate](https://baike.so.com/doc/2610851-2756769.html)是一个开放源代码的对象关系映射[框架](https://baike.so.com/doc/1863840-1971314.html)

性能：a.对JDBC进行了非常轻量级的对象封装，使得Java程序员可以随心所欲的使用对象[编程](https://baike.so.com/doc/3390106-3568658.html)思维来操纵[数据库](https://baike.so.com/doc/5447468-5685836.html)。 b.Hibernate可以应用在任何使用JDBC的场合，既可以在Java的客户端程序使用，也可以在Servlet/JSP的Web应用中使用.c. Hibernate可以在应用EJB的J2EE架构中取代CMP，完成[数据持久化](https://baike.so.com/doc/5030500-5256785.html)的重任。

1. spring

Spring是一个开源框架,Spring是一个分层的JavaSE/EE**full-stack(一站式)** [轻量级](https://baike.so.com/doc/585215-619452.html)开源框架。

性能：a.它是为了解决企业应用开发的复杂性而创建的。b.框架的主要优势之一就是其分层架构，分层架构允许使用者选择使用哪一个组件，同时为 J2EE [应用程序](https://baike.so.com/doc/3417785-3597266.html)开发提供集成的框架。Spring使用基本的[JavaBean](https://baike.so.com/doc/2778481-2932635.html)来完成以前只可能由EJB完成的事情。然而，Spring的用途不仅限于[服务器](https://baike.so.com/doc/4487696-4696885.html)端的开发。从简单性、可测试性和松耦合的角度而言，任何Java应用都可以从Spring中受益。Spring的核心是控制反转(IoC)和面向切面(AOP)。

MVC的架构

MVC全名是Model View Controller，是模型(model)-视图(view)-控制器(controller)的缩写，一种软件设计典范，用一种业务逻辑、数据、界面显示分离的方法组织代码，将业务逻辑聚集到一个部件里面，在改进和个性化定制界面及用户交互的同时，不需要重新编写业务逻辑。MVC被独特的发展起来用于映射传统的输入、处理和输出功能在一个逻辑的图形化用户界面的结构中。MVC开始是存在于桌面程序中的，M是指业务模型，V是指[用户界面](https://baike.so.com/doc/5571913-5787091.html)，C则是控制器，使用MVC的目的是将M和V的实现代码分离，从而使同一个程序可以使用不同的表现形式。

J2EE的三个架构在MVC的位置和作用

1、View 层 主要完成 数据处理功能，因此就和三层结构的表示层相对应，因此这一层的实现上多用于 JSP ,原因是jsp能够很容易的展现给用户一个可操作健壮视图。2、Control层 主要完成企业的业务逻辑处理（并且包括页面间的连接关系），因为和三层结构的逻辑层想对应，这一层的实现技术，多用于servlet/javabean/session bean。3、Model 层 对应这三层结构的数据层（存储层）。这一层的实现技术对应于j2ee的Entity bean .如此划分后，为了增强企业应用的健壮性，和可扩展性及效率，可以采用适当的设计模式（桥模式，和value object模式这两种设计模式，