EJB注释是通过@来实现的Java认证考试 PDF转换可能丢失图片或格式，建议阅读原文 https://www.100test.com/kao_ti2020/594/2021_2022_EJB_E6_B3_A8_E9_87_8A_E6_c104_594671.htm 1.有状态@Stateful和无状态@Stateless会话 EJB注释是EJB技术的一个特点。@Stateless 定义本会话为无状态会话。无状态会话Bean 是一个简单的POJO（纯粹的面向对象思想的java 对象），EJB3.0 容器自动地实例化及管理这个Bean.Stateless Session Bean不负责记录使用者状态，Stateless Session Bean一旦实例化就被加进会话池中，各个用户都可以共用。即使用户已经消亡，Stateless Session Bean的生命期也不一定结束，它可能依然存在于会话池中，供其他用户调用。 @Stateful定义本会话为有状态会话。有状态Bean是一个可以维持自身状态的会话Bean.每个用户都有自己的一个实例，在用户的生存期内，Stateful Session Bean 保持了用户的信息，即“ 有状态 ”；一旦用户灭亡（调用结束或实例结束），Stateful Session Bean的生命期也告结束。 一个bean可以同时是有状态的和无状态的 2.Local接口@Local和Remote接口@Remote 当@Local和@Remote的EJB注释都不存在时，会话Bean实现的接口默认为Local接口。如果在本机调用EJB（确保客户端与EJB容器运行在同一个JVM），采用Local接口访问EJB优于 Remote接口，因为Remote接口访问EJB需要经过远程方法调用（RPCs）环节，而Local接口访问EJB直接从JVM中返回EJB的引用。 3.JNDI的命名规则 JNDI 名的组成规则是“ 上层名称/下层名称 ”，每层之间以“ / ”分隔。默认的JNDI名称是 会话Bean 接口类型 4.改变Session Bean 的JNDI名称 要自定义JNDI名称，可以使用@LocalBinding 和

@RemoteBinding 注释，@LocalBinding注释指定Session Bean的Local接口的JNDI名称，@RemoteBinding注释指定Session Bean的Remote接口的JNDI名称。例如：@RemoteBinding（jndiBinding="com/RemoteHello"） @LocalBinding（jndiBinding="com/LocalHello"）第一句定义JNDI为com/RemoteHello，第二句定义JNDI 为com/LocalHello 5.Bean 的生命周期 @PostConstruct：当bean对象完成实例化后，使用了这个注释的方法会被立即调用。这个注释同时适用于有状态和无状态的会话bean. @PreDestroy：使用这个注释的方法会在容器从它的对象池中销毁一个无用的或者过期的bean 实例之前调用。这个注释同时适用于有状态和无状态的会话bean. @PreDestroy：当一个有状态的session bean实例空闲过长的时间，容器将会钝化（passivate）它，并把它的状态保存在缓存当中。使用这个注释的方法会在容器钝化bean实例之前调用。这个注释适用于有状态的会话bean.当钝化后，又经过一段时间该bean 仍然没有被操作，容器将会把它从存储介质中删除。以后，任何针对该bean方法的调用容器都会抛出例外。 @PreDestroy：当客户端再次使用已经被钝化的有状态session bean时，新的实例被创建，状态被恢复。使用此注释的session bean会在bean的激活完成时调用。这个注释只适用于有状态的会话bean. @Init：这个注释指定了有状态session bean初始化的方法。它区别于@PostConstruct注释在于：多个@Init注释方法可以同时存在于有状态session bean 中，但每个bean实例只会有一个@Init注释的方法会被调用。这取决于bean是如何创建的（细节请看EJB 3.0规范）。@PostConstruct在@Init之后被调用。另一个有用的生命周期

方法注释是@Remove，特别是对于有状态session bean.当应用通过存根对象调用使用了@Remove注释的方法时，容器就知道在该方法执行完毕后，要把bean实例从对象池中移走。 6.拦截器（Interceptor） 拦截器可以监听程序的一个或所有方法。拦截器对方法调用流提供了细粒度控制。 @Interceptors 注释指定一个或多个在外部类中定义的拦截器。
@AroundInvoke 注释指定了要用作拦截器的方法。
用@AroundInvoke注释指定的方法必须遵守以下格式：public Object XXX（InvocationContext ctx） throws Exception.XXX 代表方法名可以任意。（以下同） 除了可以在外部定义拦截器之外，还可以将Session Bean 中的一个或多个方法定义为拦截器。 7.依赖注入 为了存取那些服务对象，你需要通过服务器的JNDI 来查找存根对象（session bean）或消息队列（MDB）。JNDI查找是把客户端与实际的服务端实现解藕的关键步骤。但是，直接使用一个字符串来进行JNDI查找并不优雅。
@EJB注释EJB存根对象注入到任何EJB 3.0容器管理的POJO 中。 @EJB （beanName="HelloWorldBean"） //@EJB （mappedName="HelloWorldBean/remote"）beanName 的beanName属性指定EJB的类名，mappedName指定Bean实例的JNDI名。 @EJB注释如果被用在JavaBean风格的setter 方法上时，容器会在属性第一次使用之前，自动地用正确的参数调用bean的setter 方法。 @EJB注释只能注入EJB存根对象，除@EJB注释之外，EJB 3.0也支持@Resource注释来注入来自JNDI的任何资源。 如果JNDI对象在本地（java：comp/env）JNDI目录中，你只需给定他的映谢名称即可，不需要带前缀。 @Resource注释可以不指定JNDI名就能注入他们，他通

过变量的类型就能获得他的JNDI名。@Resource注释可以被用在JavaBean风格的setter 方法上。 8. 定时服务 定时服务用作在一段特定的时间后执行某段程序，使用@Timeout 注释声明定时器方法。 通过依赖注入@Resource SessionContext ctx，获得SessionContext对象，调用ctx.getTimerService（）。createTimer（Date arg0， long arg1， Serializable arg2）方法创建定时器，三个参数的含义如下：Date arg0 定时器启动时间，如果传入时间小于现在时间，定时器会立刻启动。 long arg1 间隔多长时间后再次触发定时事件。单位：毫秒当定时器创建完成后，还需声明定时器方法。EJB注释是EJB技术的一个特点。 更多优质资料尽在百考试题论坛 百考试题在线题库 java认证更多详细资料 100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com