Enterprise Java Beans

Enterprise Java Beans (EJB) is development architecture for building highly scalable and robust enterprise level applications to be deployed on J2EE compliant Application Server such as JBOSS, Web Logic etc.

Types of EJB

EJB is primarily divided into three categories:

1. **Session Bean**: Session bean stores data of a particular user for a single session. It can be stateful or stateless. It is less resource intensive as compared to entity bean. Session bean gets destroyed as soon as user session terminates.
2. **Entity Bean**: Entity beans represent persistent data storage. User data can be saved to database via entity beans and later on can be retrieved from the database in the entity bean.
3. **Message Driven Bean**: A message driven bean is a type of enterprise bean, which is invoked by EJB container when it receives a message from queue or topic. Message driven bean is a stateless bean and is used to do task asynchronously.

Benefits/Advantages of using EJB

The benefits of EJB are as follows:

1. Simplified development of large-scale enterprise level application.
2. Developer has to focus only on business logic of the application. Application Server/EJB container provides most of the system level services like transaction handling, logging, load balancing, exception handling, etc.
3. EJB container manages life cycle of EJB instances, thus developer needs not to worry about when to create/delete EJB objects.

Steps to Create a Stateless EJB

Following are the steps required to create a stateless EJB −

1. Create a remote/local interface exposing the business methods that will be used by the EJB client application.
2. Use @Local annotation, if EJB client is in same environment where EJB session bean is to be deployed.
3. Use @Remote annotation, if EJB client is in different environment where EJB session bean is to be deployed.
4. Create a stateless session bean, implementing the above interface.
5. Use @Stateless annotation to signify it a stateless bean.

Steps to Create Stateful EJB

Following are the steps required to create a stateful EJB −

1. Create a remote/local interface exposing the business methods that will be used by the EJB client application.

2. Use @Local annotation if EJB client is in same environment where EJB session bean need to be deployed.
3. Use @Remote annotation if EJB client is in different environment where EJB session bean need to be deployed.
4. Create a stateful session bean, implementing the above interface.
5. Use @Stateful annotation to signify it a stateful bean.