What is EJB

1. EJB is an acronym for enterprise java bean. It is a specification provided by Sun Microsystems to develop secured, robust and scalable distributed applications.
2. It is a reusable piece of software that you can plug in any other application.
3. It is a specially constructed Java class written in the Java and coded according to the JavaBeans API specifications.
4. To run EJB application, you need an application server (EJB Container) such as Jboss, Glassfish, Weblogic, Websphere etc. It performs:
5. transaction management: a transaction can be started automatically before a method of the EJB is invoked, and committed or rollbacked once this method returns. This transactional context is propagated to calls to other EJBs.
6. Security: a check can be made that the callerhas the necessary roles to execute the method
7. concurrency: the container makes sure that only one thread at a time invokes a method of your EJB instance.
8. distribution: some EJBs can be called remotely, from another JVM
9. EJB application is deployed on the server, so it is called server side component also.
10. EJB is like COM (Component Object Model) provided by Microsoft. But, it is different from Java Bean, RMI and Web Services.

Following are the unique characteristics that distinguish a JavaBean from other Java classes:

1. It provides a default, no-argument constructor.
2. It should be serializable and implement the Serializable interface.
3. It may have a number of properties which can be read or written.
4. It may have a number of "getter" and "setter" methods for the properties.

When use Enterprise Java Bean?

1. Application needs Remote Access. In other words, it is distributed.
2. Application needs to be scalable. EJB applications supports load balancing, clustering and fail-over.
3. Application needs encapsulated business logic. EJB application is separated from presentation and persistent layer.

Types of Enterprise Java Bean

There are 3 types of enterprise bean in java.

1. Session Bean

Session bean contains business logic that can be invoked by local, remote or webservice client.

1. Message Driven Bean

Like Session Bean, it contains the business logic but it is invoked by passing message.

1. Entity Bean

It encapsulates the state that can be persisted in the database. It is deprecated. Now, it is replaced with JPA (Java Persistent API).