

# Developing EJB applications

[← Developing application clients](#)

[↑ Tutorials](#)

[Developing JPA applications →](#)

Enterprise JavaBeans (EJB) is a platform for developing reusable and portable Java applications. In very simple terms we can say that EJB is a Java code that runs in a runtime environment called EJB container.

EJB has three different types of components: Session beans, Message Driven Beans and Entity bean. Session beans and Message Driven Beans are used for developing the business logic and Entity beans are used for persistence.

EJB provides the following services: transactions, persistence, state management, messaging, security and many more. EJB can be considered as a framework because of the components and services they provide for enterprise application development.

This document is organized in the following sections:

- [EJB2 Vs EJB3](#)
- [Types of Beans](#)
- [EJB tutorials](#)

## EJB2 Vs EJB3

EJB2 was quite complex for developers. It involved a lot of code-home interface, remote interface, local interface, bean class and deployment descriptors. Also you have to write lots of bean classes which are never called, but are required by the bean class to implement the interface. You need to implement life cycle callback methods like `ejbCreate`, `ejbRemove`, `ejbActivate`, `ejbPassivate` compulsorily even if those are not required. Also many object oriented features such as inheritance, polymorphisms were not supported by EJB2.

EJB3 as against EJB2 has undergone a great change. The hard work and innovations from the community has simplified EJB development to a great extent. In EJB3, classes are nothing but a simple Plain Old Java Object(POJO) and interfaces are Plain Old Java Interfaces(POJI). EJB3 objects are plain Java object which are given special powers with Annotations. A POJO plus Annotation is nothing but a EJB3 object. EJB3 is one of the best framework available for server side development due to the following reasons:

1. Ease of use- A EJB is a simple POJO plus annotation.
2. Complete stack of solutions for EJB development which includes persistence, messaging, annotations, and dependency injection.
3. Broad vendor support which includes the support from market players like IBM, Oracle and open source vendors like Apache Geronimo.
4. Annotations can be used in place of deployment descriptors.
5. Dependency injection as against performing a JNDI lookup for the resources.

We will discuss each of these in upcoming sections and EJB applications.

## Types of Beans

There are three different types of beans as was discussed earlier:

- Session Beans
  1. Stateless Session Bean
  2. Stateful Session Bean
- Message Driven Bean
- Entity Beans

We will discuss each type of Bean one by one.

## EJB tutorials

- [5-minute Tutorial on Enterprise Application Development with Eclipse and Geronimo](#)
- [JMS application with Message-Driven Bean](#)
- [Referring to an EJB from outside its EAR](#)
- [Stateful Session Bean](#)
- [Stateless Session Bean](#)