@PersistenceContext(unitName="OPEN\_JPA")

**PersistenceContext** is the connection between your in-memory instances and the database. It is manipulated through “EntityManager” interface. A persistence context is a set of

entity instances in which for any persistent entity identity there is a unique entity instance. Within the

persistence context, the entity instances and their lifecycle are managed

**EntityManager**:-

An EntityManager instance is associated with a persistence context. The EntityManager interface

defines the methods that are used to interact with the persistence context. The EntityManager

API is used to create and remove persistent entity instances, to find entities by their primary key, and to

query over entities.

The set of entities that can be managed by a given EntityManager instance is defined by a persistence

unit.

**Persistence provider** :- the mechanism to transfer entity information back and forth between java object and the database is implemented within your persistence provider. The persistence provider is typically tailored to a particular type of database. Which particular implementation is used is transparent to the entity. The persistence provider implementation worries about the proper time to load and store your data. You don’t have to worry about synchronizing your objects with the underlying database: the persistence provider handles it for you.

**Entity**

The entity class must be annotated with the Entity annotation or denoted in the XML descriptor as an

entity.

**Obtaining an Entity Manager in the Java EE Environment**

A container-managed entity manager is obtained by the application through dependency injection, or

direct lookup of the entity manager in the JNDI namespace. The container manages the persistence

context lifecycle and the creation and the closing of the entity manager instance transparently to the

application.

The PersistenceContext annotation is used for entity manager injection.

.

For example,

@PersistenceContext(unitName="OPEN\_JPA")

EntityManager em;

**Id Annotation**

The Id annotation specifies the primary key property or field of an entity.