# **EJB Life Cycle**

Reprendre HelloEJB

## **Etape**

Dans le package com.formation.service créez une interface java avec comme nom **SessionLife.java**

package **com.formation.service;**

import **javax.ejb.Remote;**

**@Remote**

publicinterface **SessionLife {**

public **String affiche(String param) ;**

**}**

## **Etape**

Maintenant dans le package **com.formation.service.implementation** créez un EJB Stateless **SessionLifeBean.java** qui implémente l’interface **SessionLife.java**

**package** com.formation.service.implementation;

**import** javax.annotation.PostConstruct;

**import** javax.annotation.PreDestroy;

**import** javax.ejb.Stateless;

**import** com.formation.service.SessionLife;

@Stateless

**public** **class** SessionLifeBean **implements** SessionLife {

@Override

**public** String affiche(String param) {

System.out.println("Dans ma methode affiche...");

**return** "Requete avec " + param + " executer.";

}

@PostConstruct

**public** **void** initialize () {

//code avant écécution

System.*out*.println("Initialisation ....");

}

@PreDestroy

**public** **void** destroyBean() {

// code aprés éxécution

System.*out*.println(" Destruction ....");

}

}

## **Etape**

Nous allons maintenant crée un client remote pour le EJB.

Reprendre le projet HelloEJBClient

### **Création du client remote**

Dans le projet **HelloEJBClient** créez une classe java **LifeCycleClient.java** dans le package **com.formation.client**

Ajouter le contenu comme suit :

**package** com.formation.client;

**import** javax.naming.Context;

**import** javax.naming.NamingException;

**import** com.formation.service.SessionLife;

**import** com.formation.service.implementation.SessionLifeBean;

**import** com.formation.util.ClientUtil;

**public** **class** LifeCycleClient {

**public** **static** **void** main(String[] args) {

SessionLife bean = *doLookup*();

System.*out*.println(bean.affiche("Hello")); // 4. Appel du service

}

**private** **static** SessionLife doLookup() {

Context context = **null**;

SessionLife bean = **null**;

**try** {

// 1. Obtenir le context

context = ClientUtil.*getInitialContext*();

// 2. Generate JNDI Lookup name

String lookupName = *getLookupName*();

// 3. Lookup and cast

bean = (SessionLife) context.lookup(lookupName);

} **catch** (NamingException e) {

e.printStackTrace();

}

**return** bean;

}

**private** **static** String getLookupName() {

// nom ear

String appName = "";

//nom du jar ejb

String moduleName = "HelloEJB";

//nom optionnel JBoss7

String distinctName = "";

// Nom de la classe implementant le EJB

String beanName = SessionLifeBean.**class**.getSimpleName();

// Nom complet de l'interface

**final** String interfaceName = SessionLife.**class**.getName();

// Create a look up string name

String name = "ejb:" + appName + "/" + moduleName + "/" + distinctName

+ "/" + beanName + "!" + interfaceName;

**return** name;

}

}

### Test

Deployer le EJB dans JBoss

Lancer le client

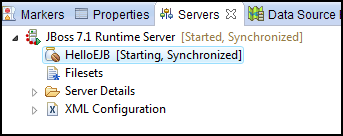
Dans la console du client java on obtient



Dans la console du Serveur JBoss on obtient



Dans l’onglet server de JBoss , choisir le EJB **HelloEJB**

****

(click-droit) puis **Remove**

Dans la console du Serveur JBoss on obtient

