Objectif : Dans cet atelier, on va aborder les sujets suivants

1- Créer une première entité JPA

2-Interagir les entités JPA en utilisant EntityManager

3- Génération de formulaires JSF à partir des entités JPA

4- Générer des entités JPA à partir d'un schéma existant d'une base de données

5- requêtes JPA nommées et JPQL

6-relations entre les entités

7-Générer des applications compètes JSF à partir des entités JPA

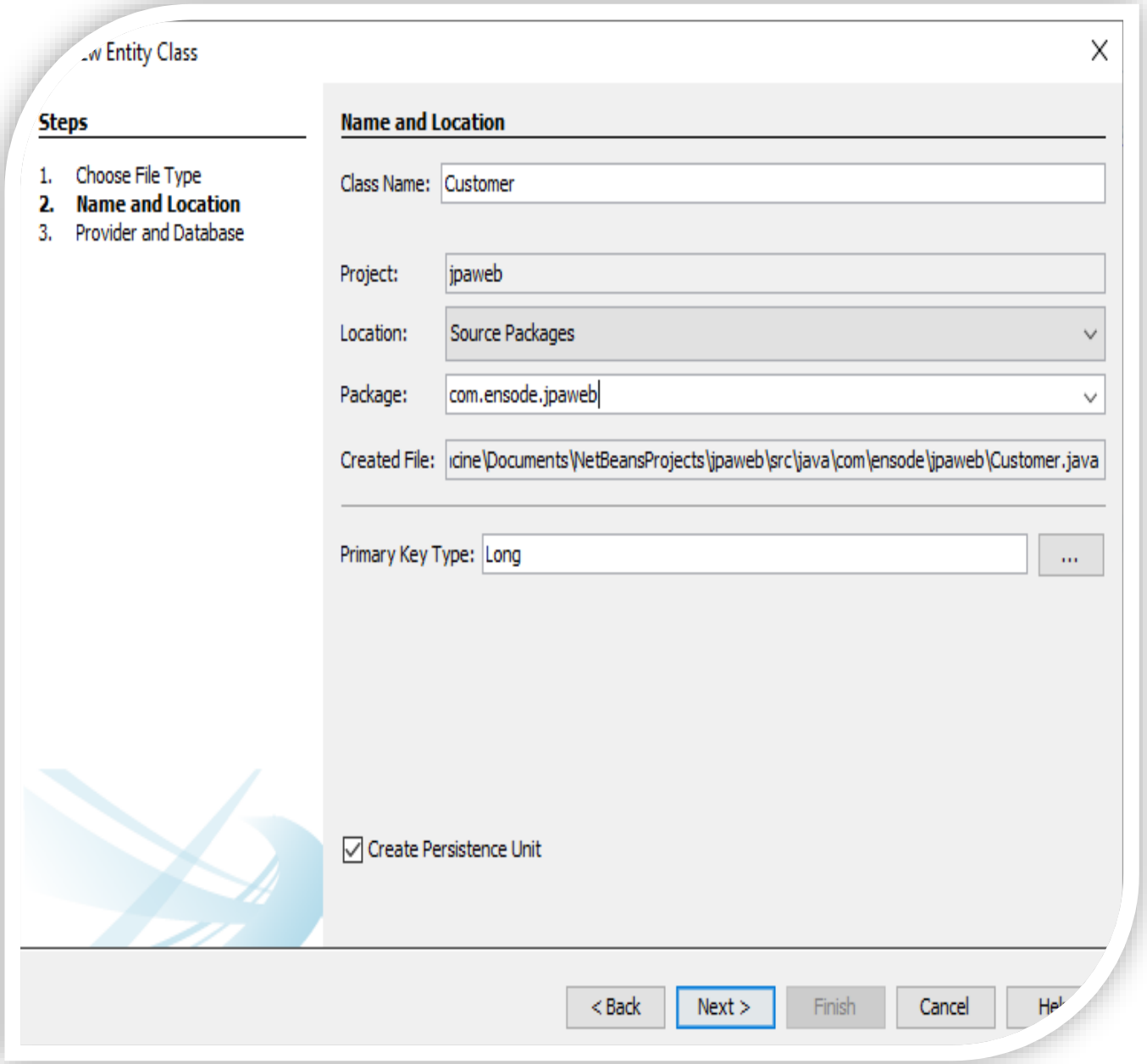
Java Persistence API (abrégée en JPA), est une interface de programmation Java permettant aux développeurs d'organiser des données relationnelles dans des applications utilisant la plateforme Java.

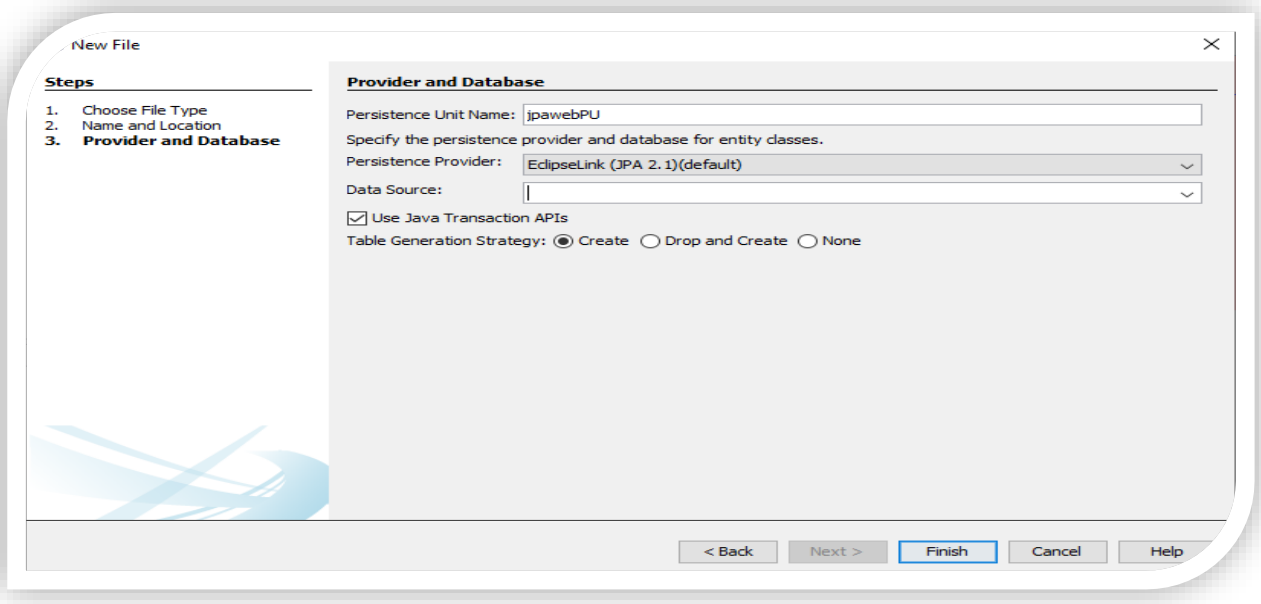
Java Persistence API repose essentiellement sur l'utilisation des annotations, introduites dans Java 5. Elles permettent de définir facilement des objets métier, qui pourront servir d'interface entre la base de données et l'application, dans le cadre d'un mapping objet-relationnel.

ETAPE 1 : Créer notre première entité JPA

Entité JPA : Les entités JPA sont des classes Java dont les champs sont conservés dans une base de données par l'API JPA. Les entités JPA sont des POJOs (Plain Old Java Objects), en tant que tels, ils n'ont pas besoin d'hériter d'aucune classes spécifiques ou d'implémenter aucune interface spécifique. Une classe Java est désignée comme étant une entité JPA avec l'annotation **@Entity**
La premier étape à faire est crée un nouveau projet base sure le Framework serverface.

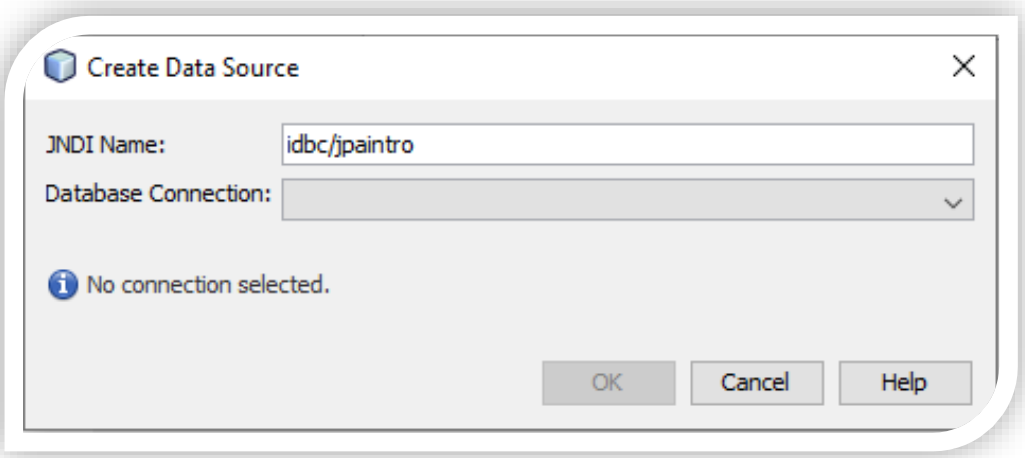
Deuxième étape crée un class java de type entity class, cette premier class nous permet de crée une persistance pour notre application et le connecter avec un base de donnes de plus.

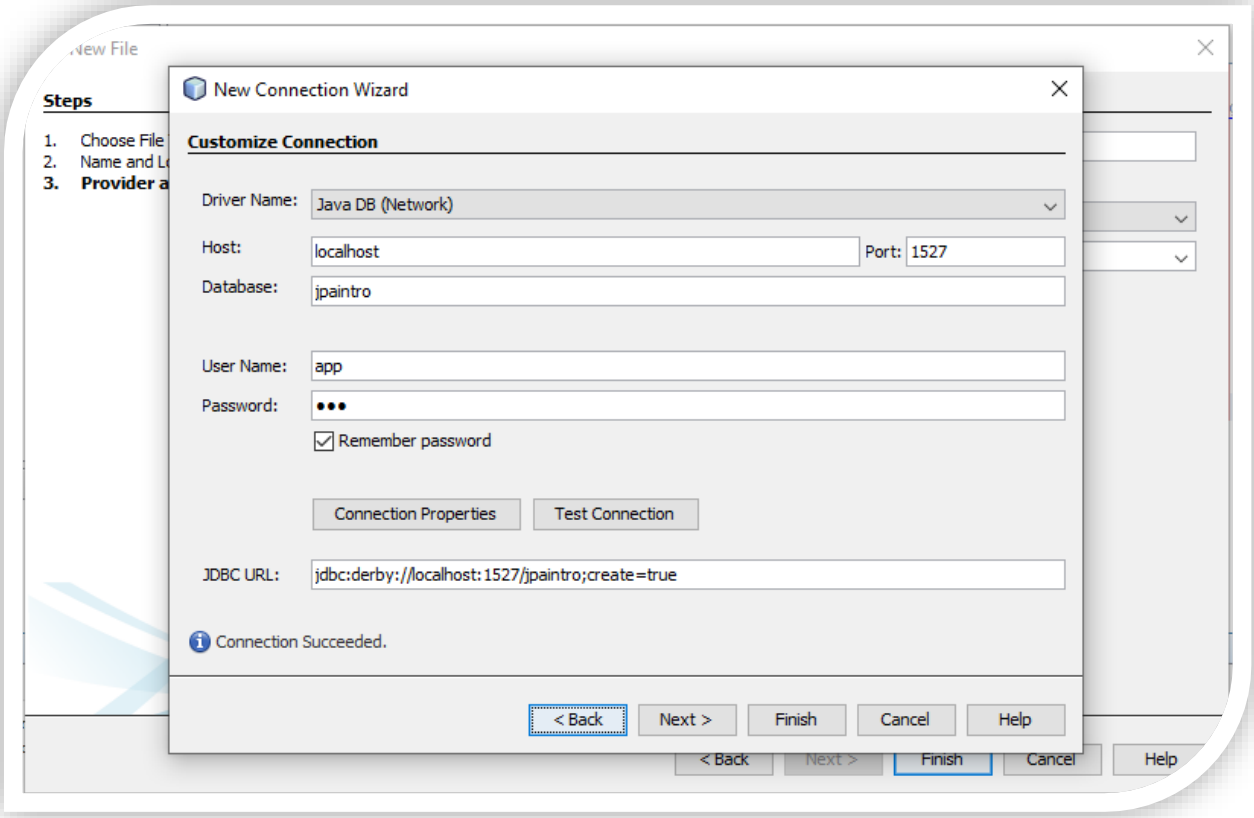

w Entity Class
Steps
1. Choose File Type
2. Name and Location
3. Provider and Database
Name and Location
Class Name: Customer
Project: jpaweb
Location: Source Packages
Package: com.ensode.jpaweb
Created File: ıcine\Documents\NetBeansProjects\jpaweb\src\java\com\ensode\jpaweb\Customer.java
Primary Key Type: Long
...
Create Persistence Unit
< Back
Next >
Finish
Cancel

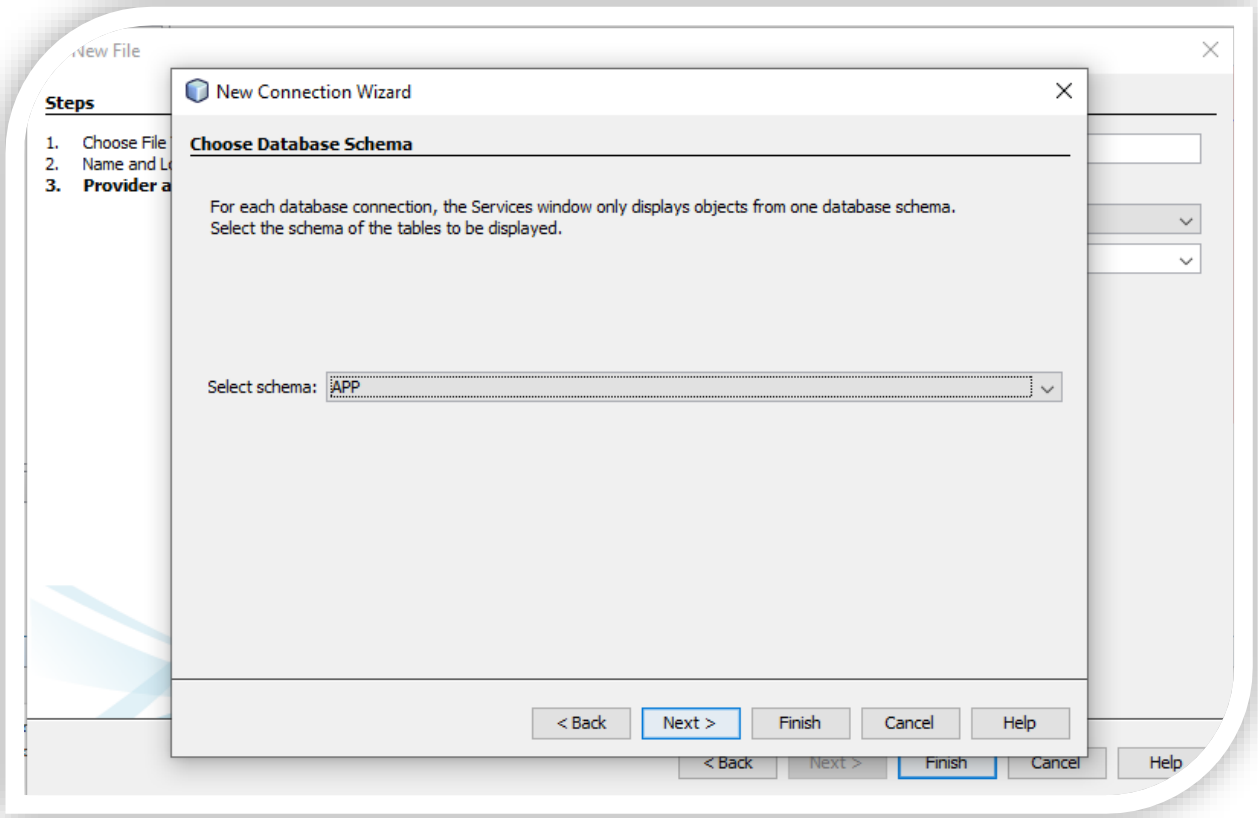


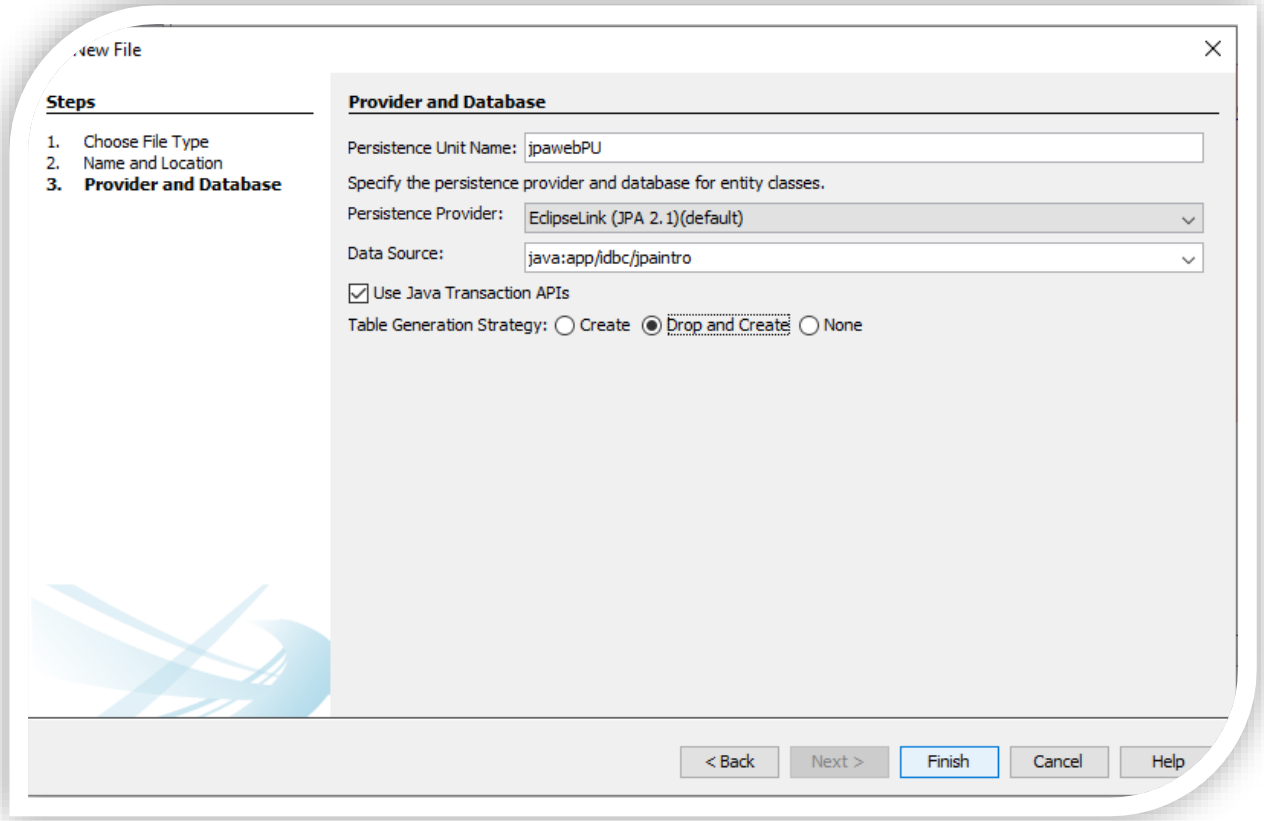
Dans cette étape, on doit sélection notre basse de donne, dans notre cas on va crée un nouveau base de donne vide.

Dans les étapes suivant on va crée notre base de donnes (choisir le driver, port, username …).

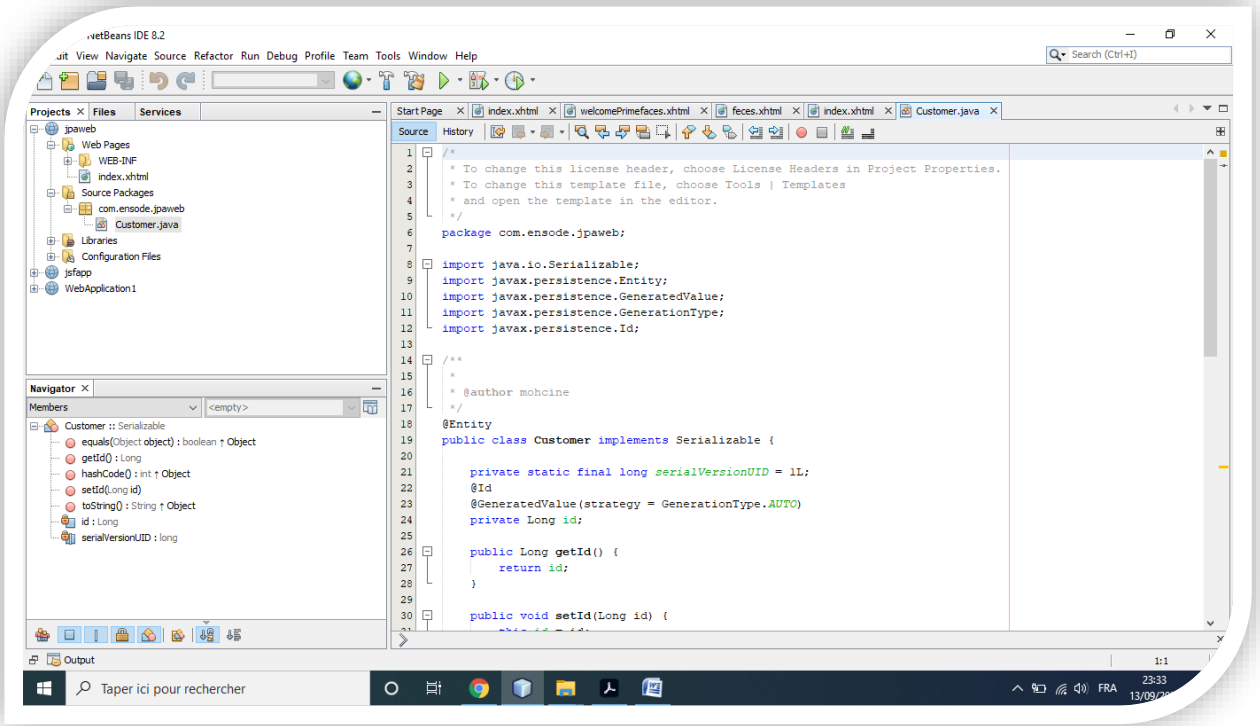



New File
Steps
1. Choose File
2. Name and L
3. Provider a
New Connection Wizard
Customize Connection
Driver Name: Java DB (Network)
Host: localhost
Port: 1527
Database: jpaintro
User Name: app
Password: •••
Remember password
Connection Properties
Test Connection
JDBC URL: jdbc:derby://localhost:1527/jpaintro;create=true
Connection Succeeded.
< Back
Next >
Finish
Cancel
Help


New File
Steps
1. Choose File
2. Name and L
3. Provider a
New Connection Wizard
Choose Database Schema
For each database connection, the Services window only displays objects from one database schema.
Select the schema of the tables to be displayed.
Select schema: APP
< Back
Next >
Finish
Cancel
Help



Dans cette étape il est recommandé de sélectionné Drop and create dans les transactions Apis.



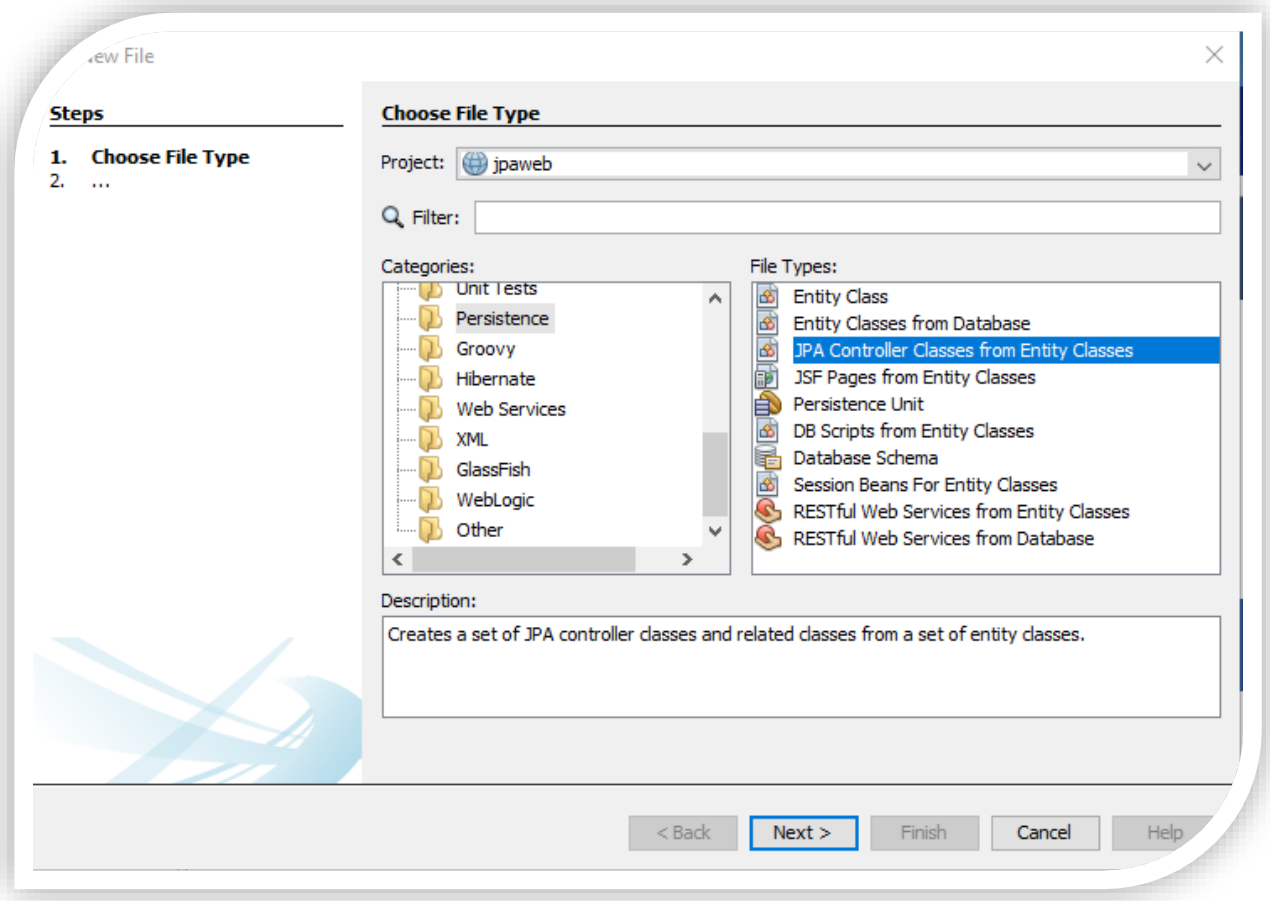
En remarque qu'il un ensemble d'annotation dans notre projet :

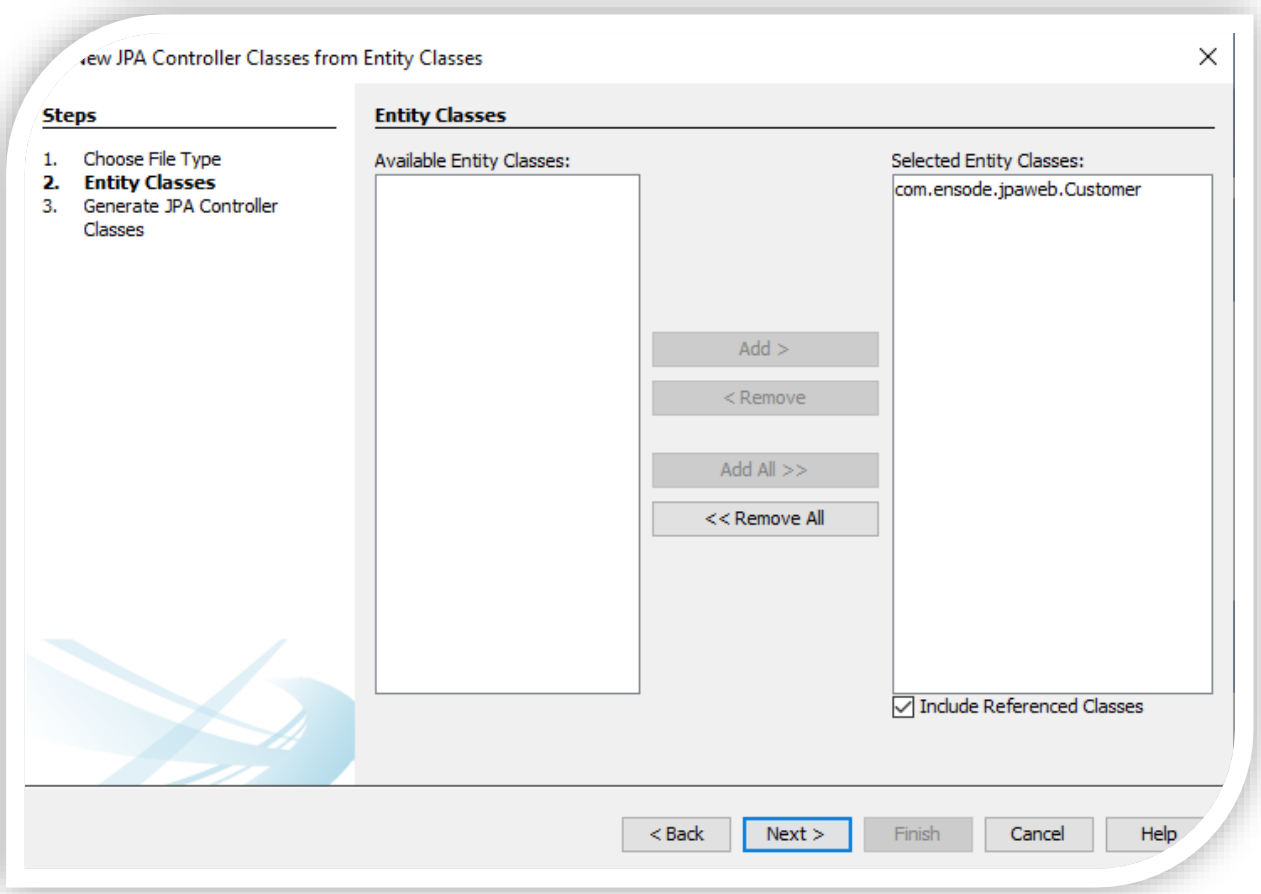
@Entity signifie que cette classe est un Entity classe.

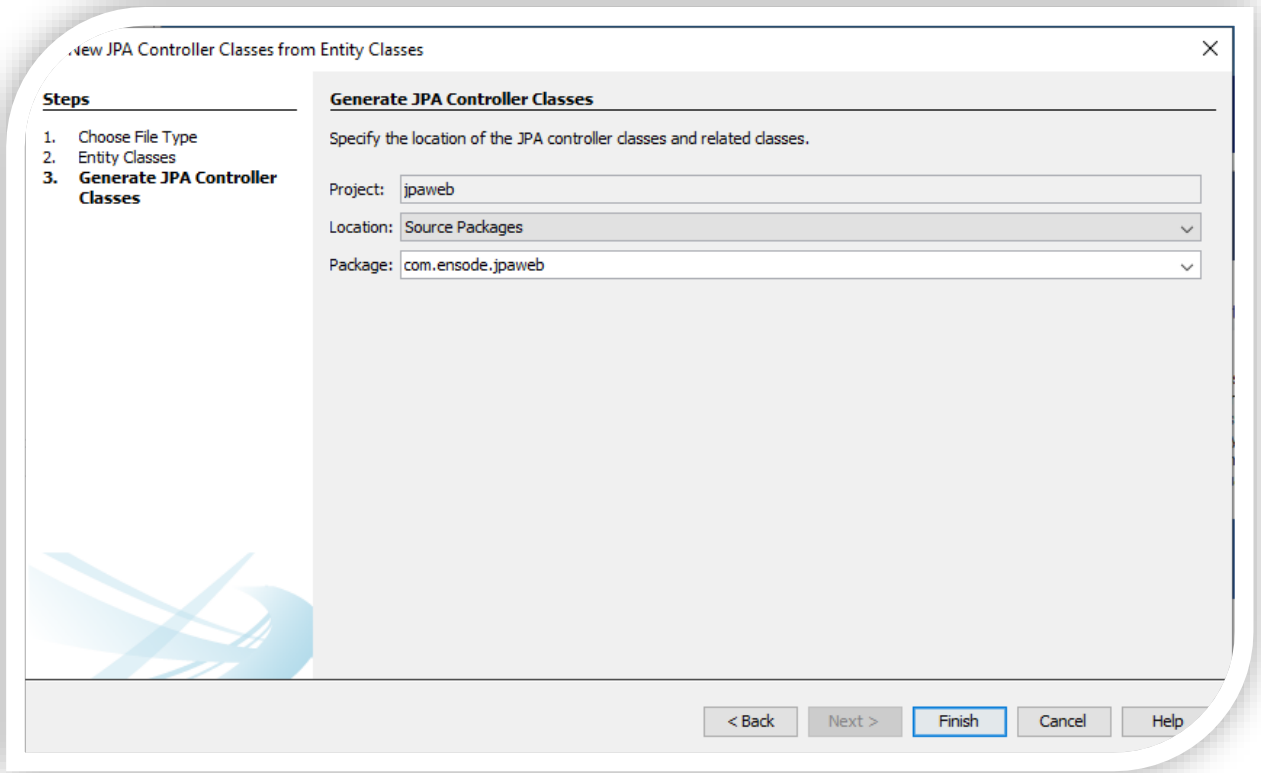
@id et **@generetedVale** signifie que l'attribue qu'il va les suivre est une clé primaire qui va être généré selon une stratégie, dan notre cas on choisi de le generer automatiquement.

ETAPE 2 : Création d'un DAO (Data Access Object)

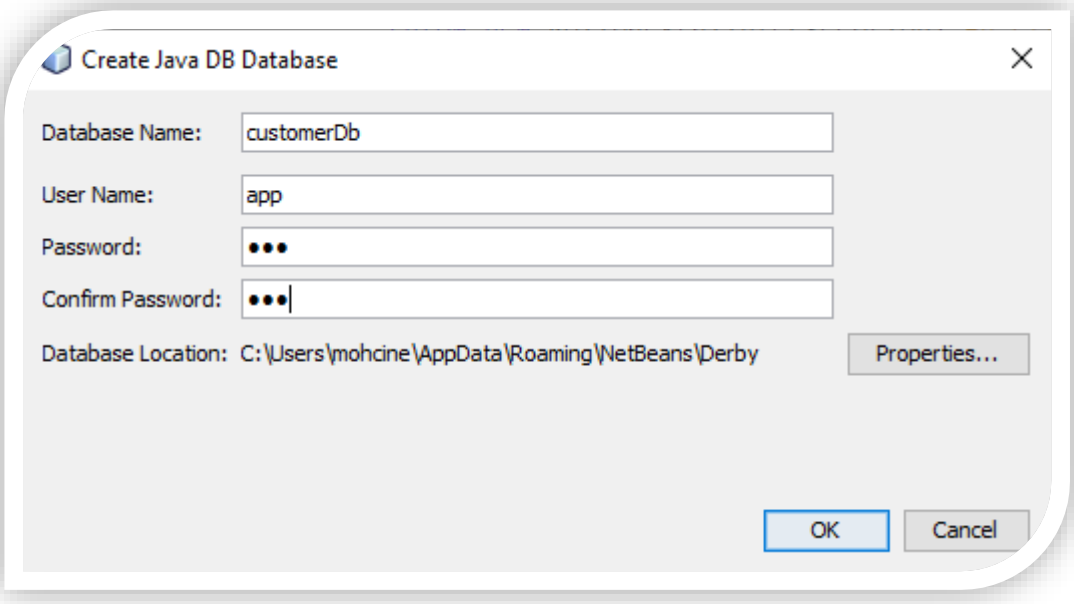
Le design pattern Data Access Object (DAO) consiste à écrire du code qui interagit avec une base de données. Le DAO permet de garder toutes les fonctionnalités d'accès à la base de données dans les classes DAO. Ce qui permet de séparer les rôles, et libérer les autres couches (couche de présentation et couche métier) de toute logique de persistance.

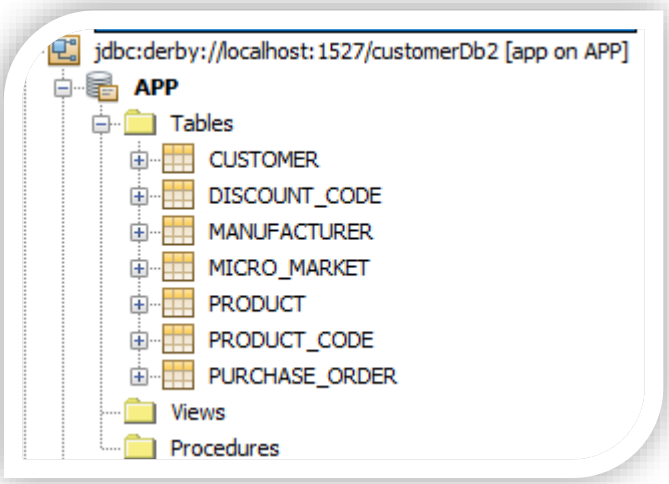



ew JPA Controller Classes from Entity Classes
Steps
1. Choose File Type
2. Entity Classes
3. Generate JPA Controller Classes
Entity Classes
Available Entity Classes:
Selected Entity Classes:
com.ensode.jpaweb.Customer
Add >
< Remove
Add All >>
<< Remove All
Include Referenced Classes
< Back
Next >
Finish
Cancel
Help

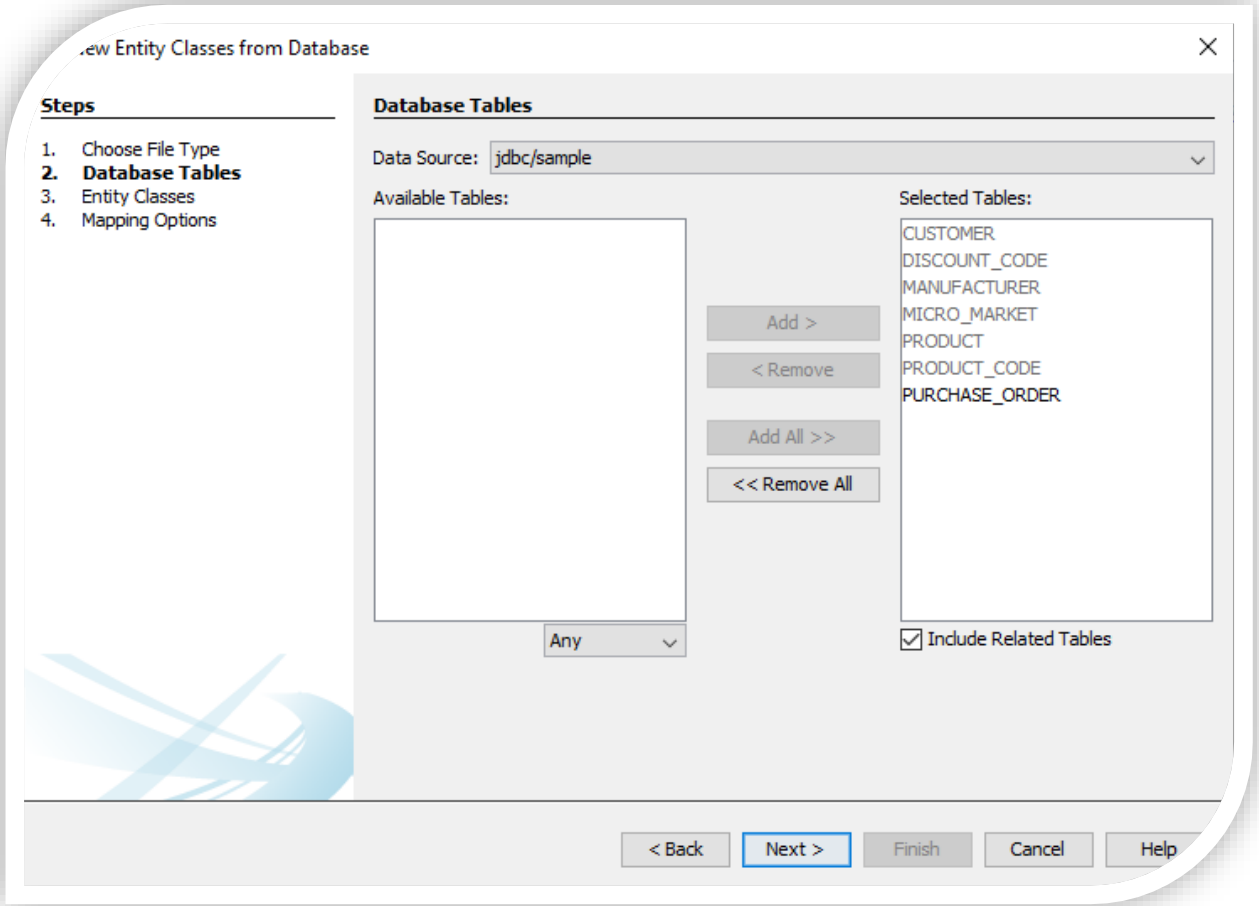

New JPA Controller Classes from Entity Classes
Steps
1. Choose File Type
2. Entity Classes
3. Generate JPA Controller Classes
Generate JPA Controller Classes
Specify the location of the JPA controller classes and related classes.
Project: jpaweb
Location: Source Packages
Package: com.ensode.jpaweb
< Back
Next >
Finish
Cancel
Help

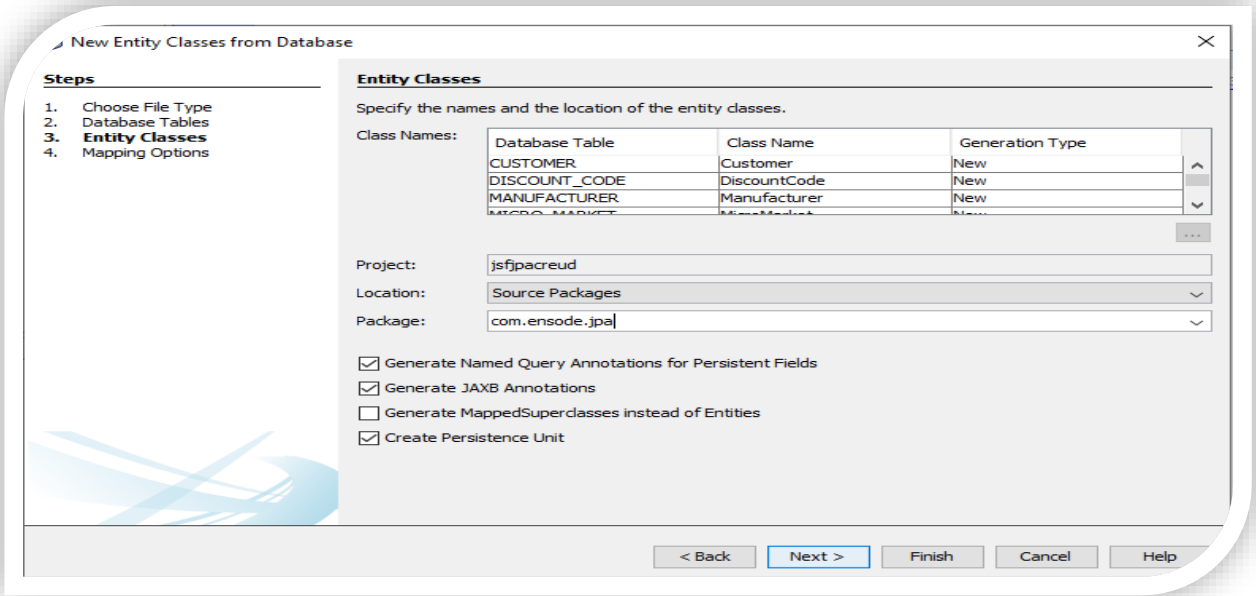
ETAPE 3 : Génération automatique des entités JPA

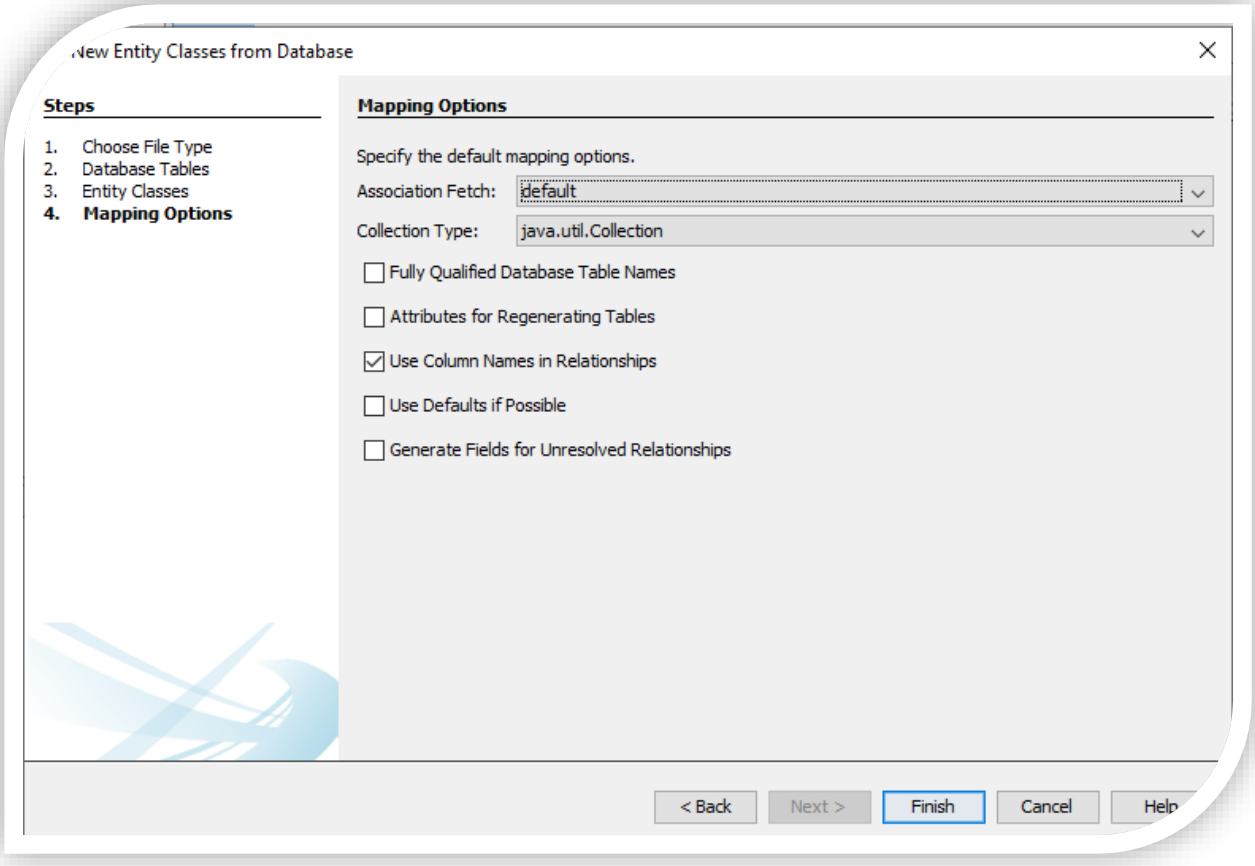


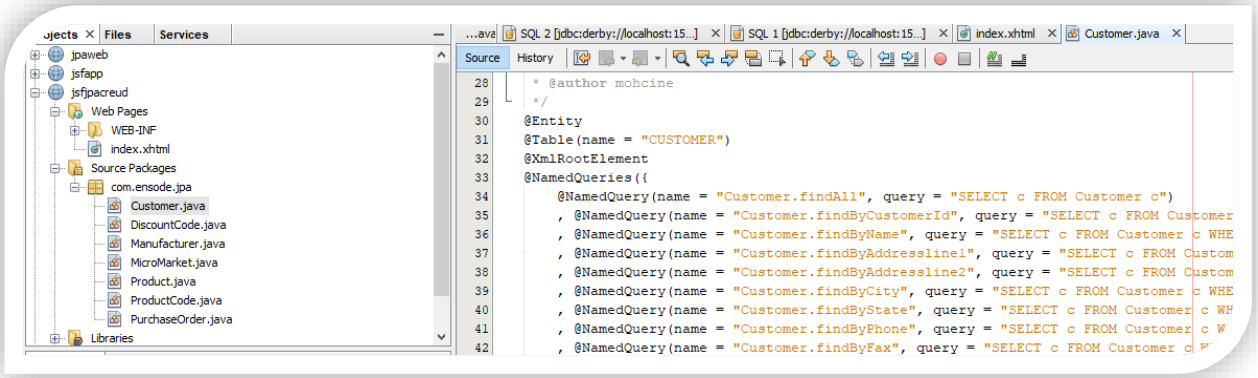


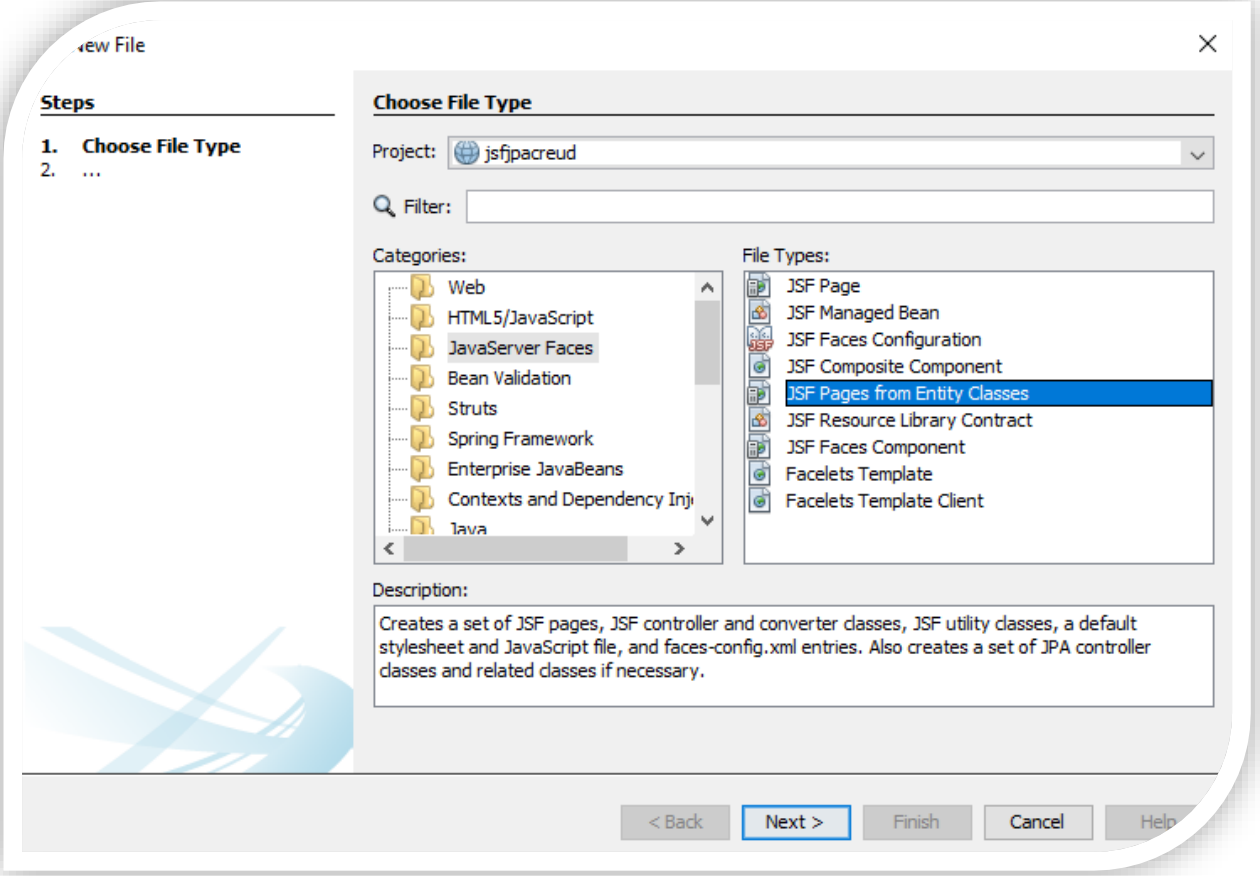
ETAPE 4 : Génération des entités à partir d'un schéma existant

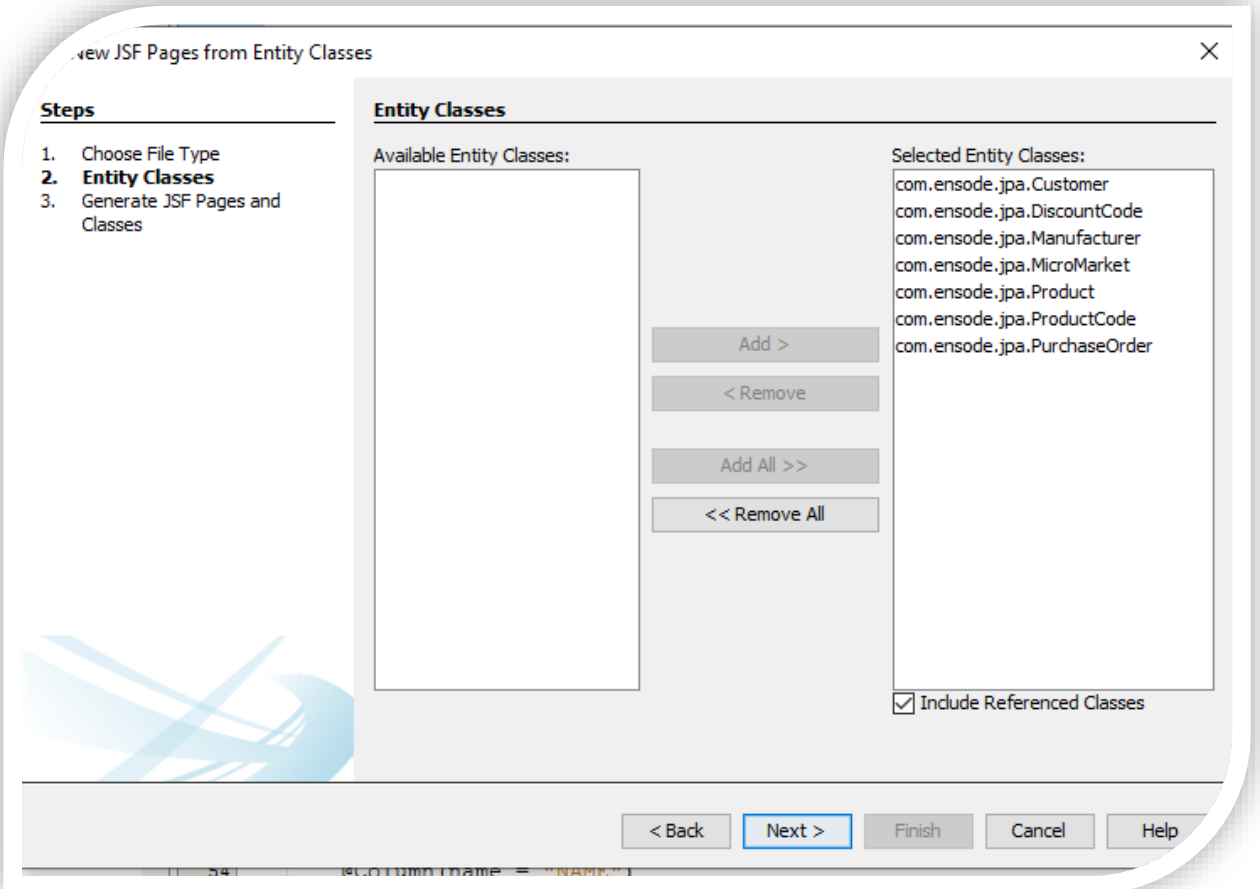

ew Entity Classes from Database
Steps
1. Choose File Type
2. Database Tables
3. Entity Classes
4. Mapping Options
Database Tables
Data Source: jdbc/sample
Available Tables:
Selected Tables:
CUSTOMER
DISCOUNT_CODE
MANUFACTURER
MICRO_MARKET
PRODUCT
PRODUCT_CODE
PURCHASE_ORDER
Add >
< Remove
Add All >>
<< Remove All
Any
Include Related Tables
< Back
Next >
Finish
Cancel
Help

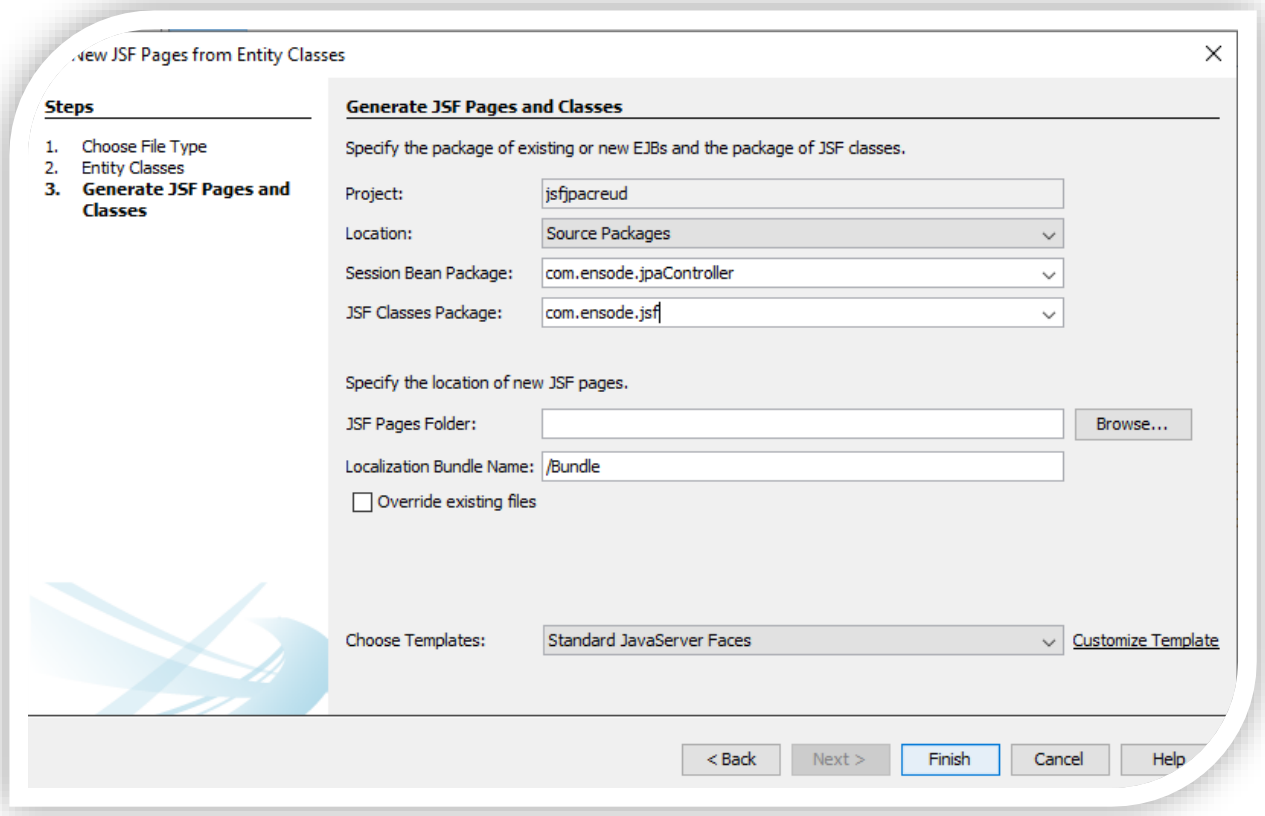

New Entity Classes from Database
Steps
1. Choose File Type
2. Database Tables
3. Entity Classes
4. Mapping Options
Entity Classes
Specify the names and the location of the entity classes.
Class Names:
Database Table
Class Name
Generation Type
CUSTOMER
Customer
New
DISCOUNT_CODE
DiscountCode
New
MANUFACTURER
Manufacturer
New
Project:
jsfjpacreud
Location:
Source Packages
Package:
com.ensode.jpa
Generate Named Query Annotations for Persistent Fields
Generate JAXB Annotations
Generate MappedSuperclasses instead of Entities
Create Persistence Unit
< Back
Next >
Finish
Cancel
Help

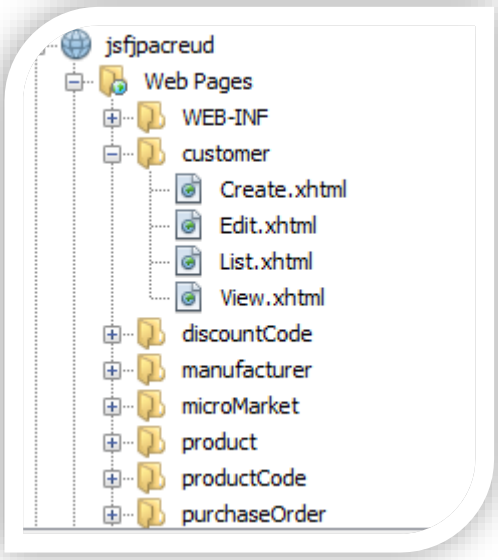

New Entity Classes from Database
Steps
1. Choose File Type
2. Database Tables
3. Entity Classes
4. Mapping Options
Mapping Options
Specify the default mapping options.
Association Fetch: default
Collection Type: java.util.Collection
Fully Qualified Database Table Names
Attributes for Regenerating Tables
Use Column Names in Relationships
Use Defaults if Possible
Generate Fields for Unresolved Relationships
< Back
Next >
Finish
Cancel
Help

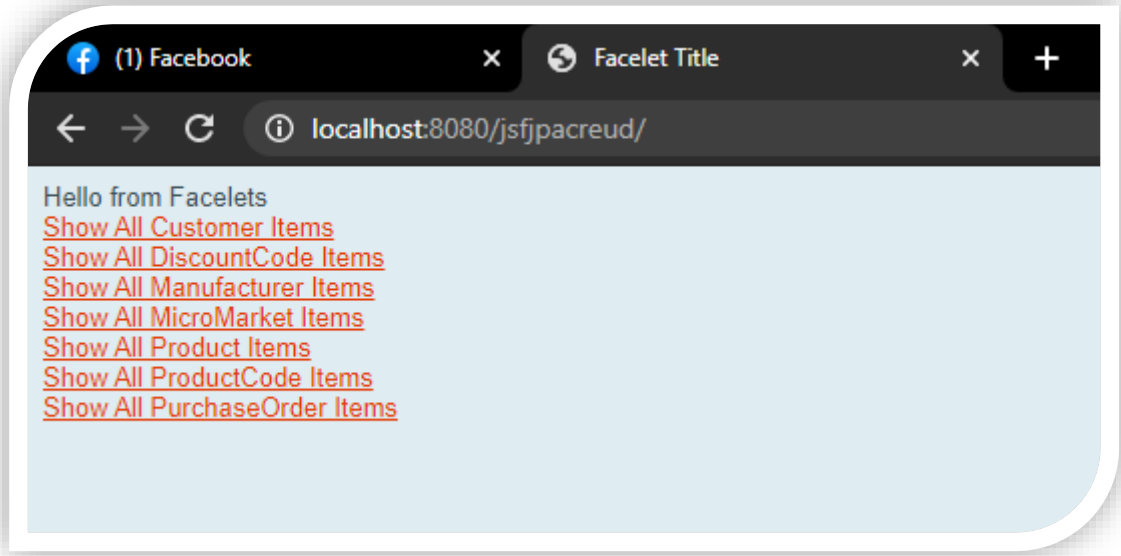

Files
Services
jpaweb
jsfapp
jsfjpacreud
Web Pages
WEB-INF
index.xhtml
Source Packages
com.ensode.jpa
Customer.java
DiscountCode.java
Manufacturer.java
MicroMarket.java
Product.java
ProductCode.java
PurchaseOrder.java
Libraries
SQL 2 [jdbc:derby://localhost:15...]
SQL 1 [jdbc:derby://localhost:15...]
index.xhtml
Customer.java
Source
History
* @author mohcine
*/
@Entity
@Table(name = "CUSTOMER")
@XmlRootElement
@NamedQueries({
@NamedQuery(name = "Customer.findAll", query = "SELECT c FROM Customer c")
, @NamedQuery(name = "Customer.findByCustomerId", query = "SELECT c FROM Customer
, @NamedQuery(name = "Customer.findByName", query = "SELECT c FROM Customer c WHE
, @NamedQuery(name = "Customer.findByAddressline1", query = "SELECT c FROM Custom
, @NamedQuery(name = "Customer.findByAddressline2", query = "SELECT c FROM Custom
, @NamedQuery(name = "Customer.findByCity", query = "SELECT c FROM Customer c WHE
, @NamedQuery(name = "Customer.findByState", query = "SELECT c FROM Customer c WH
, @NamedQuery(name = "Customer.findByPhone", query = "SELECT c FROM Customer c W
, @NamedQuery(name = "Customer.findByFax", query = "SELECT c FROM Customer c W

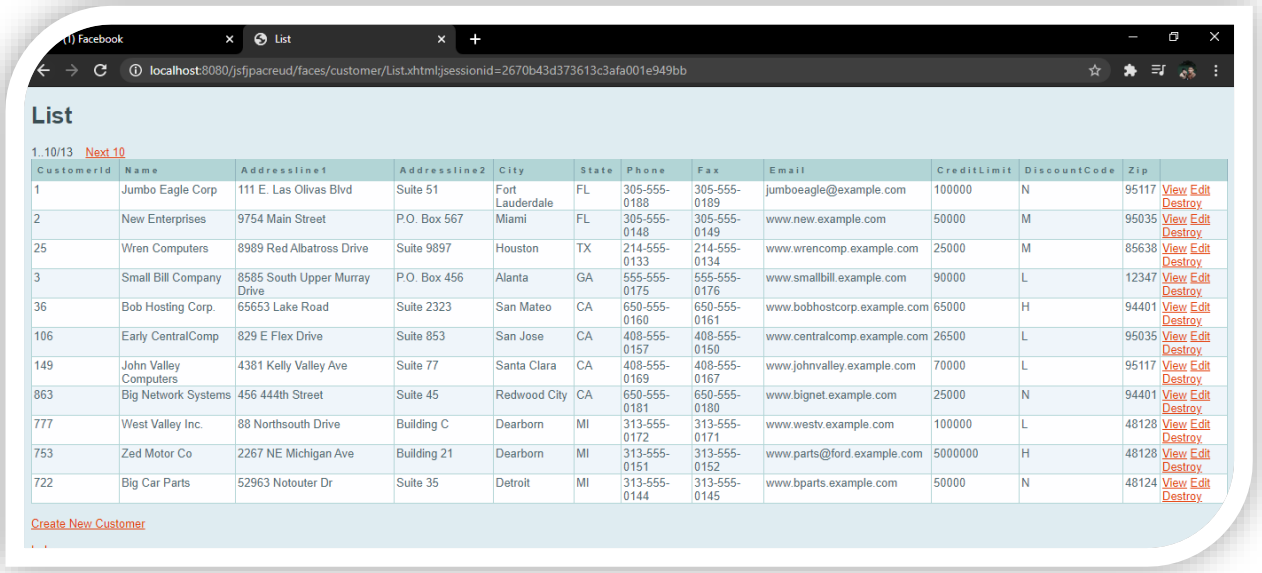

New File
Steps
1. Choose File Type
2. ...
Choose File Type
Project: jsfjpacreud
Filter:
Categories:
Web
HTML5/JavaScript
JavaServer Faces
Bean Validation
Struts
Spring Framework
Enterprise JavaBeans
Contexts and Dependency Inj
Java
File Types:
JSF Page
JSF Managed Bean
JSF Faces Configuration
JSF Composite Component
JSF Pages from Entity Classes
JSF Resource Library Contract
JSF Faces Component
Facelets Template
Facelets Template Client
Description:
Creates a set of JSF pages, JSF controller and converter classes, JSF utility classes, a default stylesheet and JavaScript file, and faces-config.xml entries. Also creates a set of JPA controller classes and related classes if necessary.
< Back
Next >
Finish
Cancel
Help


New JSF Pages from Entity Classes
Steps
1. Choose File Type
2. Entity Classes
3. Generate JSF Pages and Classes
Entity Classes
Available Entity Classes:
Add >
< Remove
Add All >>
<< Remove All
Selected Entity Classes:
com.ensode.jpa.Customer
com.ensode.jpa.DiscountCode
com.ensode.jpa.Manufacturer
com.ensode.jpa.MicroMarket
com.ensode.jpa.Product
com.ensode.jpa.ProductCode
com.ensode.jpa.PurchaseOrder
Include Referenced Classes
< Back
Next >
Finish
Cancel
Help


New JSF Pages from Entity Classes
Steps
1. Choose File Type
2. Entity Classes
3. Generate JSF Pages and Classes
Generate JSF Pages and Classes
Specify the package of existing or new EJBs and the package of JSF classes.
Project:
jsfjpacreud
Location:
Source Packages
Session Bean Package:
com.ensode.jpaController
JSF Classes Package:
com.ensode.jsf
Specify the location of new JSF pages.
JSF Pages Folder:
Browse...
Localization Bundle Name:
/Bundle
Override existing files
Choose Templates:
Standard JavaServer Faces
Customize Template
< Back
Next >
Finish
Cancel
Help


jsfjpacreud
Web Pages
WEB-INF
customer
Create.xhtml
Edit.xhtml
List.xhtml
View.xhtml
discountCode
manufacturer
microMarket
product
productCode
purchaseOrder



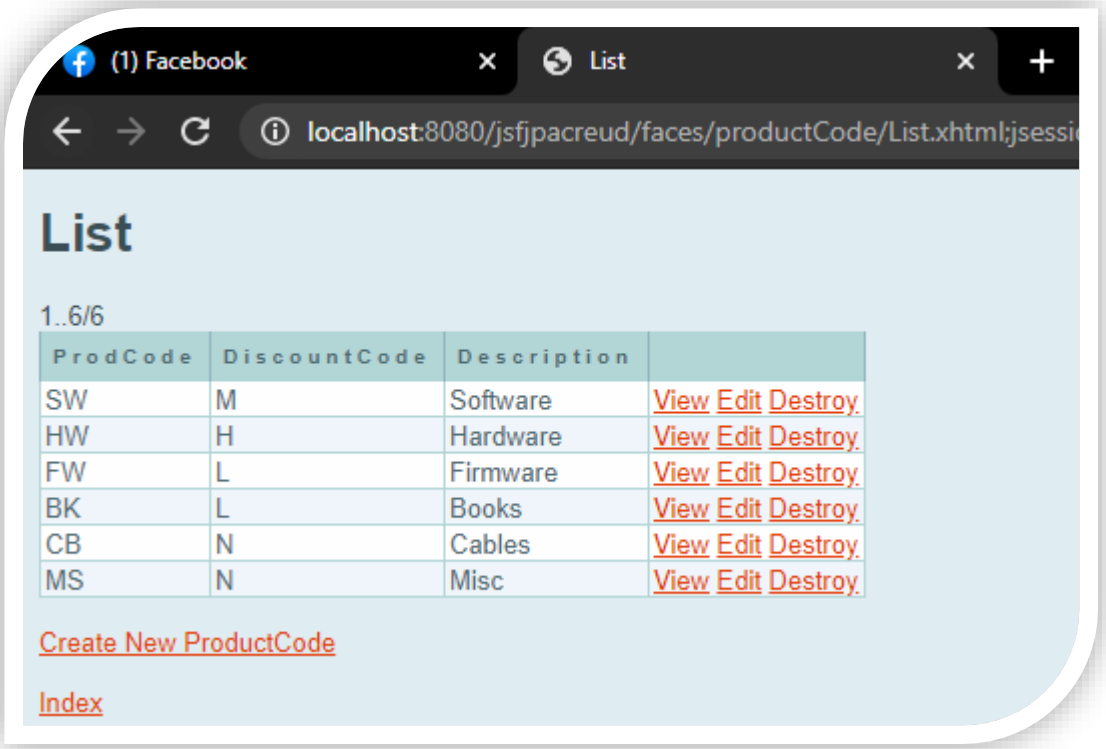


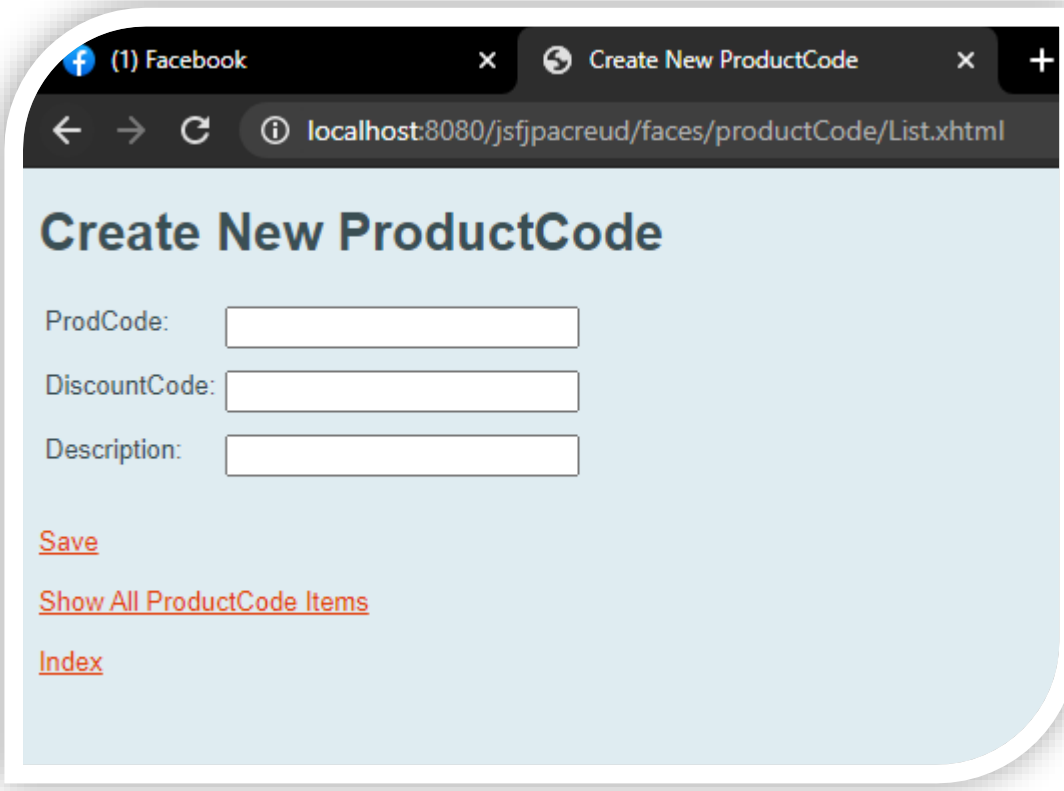
List

1..10/13 Next 10

CustomerId	Name	Addressline1	Addressline2	City	State	Phone	Fax	Email	CreditLimit	DiscountCode	Zip	
1	Jumbo Eagle Corp	111 E. Las Olivas Blvd	Suite 51	Fort Lauderdale	FL	305-555-0188	305-555-0189	jumboeagle@example.com	100000	N	95117	View Edit Destroy
2	New Enterprises	9754 Main Street	P.O. Box 567	Miami	FL	305-555-0148	305-555-0149	www.new.example.com	50000	M	95035	View Edit Destroy
25	Wren Computers	8989 Red Albatross Drive	Suite 9897	Houston	TX	214-555-0133	214-555-0134	www.wrencomp.example.com	25000	M	85638	View Edit Destroy
3	Small Bill Company	8585 South Upper Murray Drive	P.O. Box 456	Alanta	GA	555-555-0175	555-555-0176	www.smallbill.example.com	90000	L	12347	View Edit Destroy
36	Bob Hosting Corp.	65653 Lake Road	Suite 2323	San Mateo	CA	650-555-0160	650-555-0161	www.bobhostcorp.example.com	65000	H	94401	View Edit Destroy
106	Early CentralComp	829 E Flex Drive	Suite 853	San Jose	CA	408-555-0157	408-555-0150	www.centralcomp.example.com	26500	L	95035	View Edit Destroy
149	John Valley Computers	4381 Kelly Valley Ave	Suite 77	Santa Clara	CA	408-555-0169	408-555-0167	www.johnvalley.example.com	70000	L	95117	View Edit Destroy
863	Big Network Systems	456 444th Street	Suite 45	Redwood City	CA	650-555-0181	650-555-0180	www.bignet.example.com	25000	N	94401	View Edit Destroy
777	West Valley Inc.	88 Northsouth Drive	Building C	Dearborn	MI	313-555-0172	313-555-0171	www.westv.example.com	100000	L	48128	View Edit Destroy
753	Zed Motor Co	2267 NE Michigan Ave	Building 21	Dearborn	MI	313-555-0151	313-555-0152	www.parts@ford.example.com	5000000	H	48128	View Edit Destroy
722	Big Car Parts	52963 Notouter Dr	Suite 35	Detroit	MI	313-555-0144	313-555-0145	www.bparts.example.com	50000	N	48124	View Edit Destroy

Create New Customer


(1) Facebook
List
localhost:8080/jsfjpacreud/faces/productCode/List.xhtml;jsessi
List
1..6/6
ProdCode
DiscountCode
Description
SW M Software View Edit Destroy
HW H Hardware View Edit Destroy
FW L Firmware View Edit Destroy
BK L Books View Edit Destroy
CB N Cables View Edit Destroy
MS N Misc View Edit Destroy
Create New ProductCode
Index


(1) Facebook
Create New ProductCode
localhost:8080/jsfjpacreud/faces/productCode/List.xhtml
Create New ProductCode
ProdCode:
DiscountCode:
Description:
Save
Show All ProductCode Items
Index