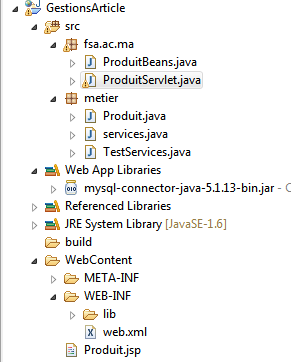
J2EE Master OTI

Atelier MVC

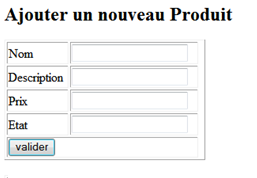
Couche Présentation

Structure du projet



**Etape1 : Ecrire fichier Article.jsp permettant de saisir les infos concernant un article**

|  |
| --- |
| <%@page import=*"metier.Produit"*%>  <%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"* pageEncoding=*"ISO-8859-1"*%>  <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">  <html>  <head>  <meta http-equiv=*"Content-Type"* content=*"text/html; charset=ISO-8859-1"*>  <title>GestionProduits</title>  </head>  <body>  <h3>Tutorial JEE( JSP Servlet MVC et JDBC)</h3>  <h2>Ajouter un nouveau Produit</h2>  <table border=*"1"* width=*"30%"*>  <tr>  <td>Nom </td>  <td><input type=*"text"* name=*"nom"*/> </td>  </tr>  <tr>  <td>Description </td>  <td><input type=*"text"* name=*"desc"*/></td>  </tr><tr>  <td>Prix </td>  <td><input type=*"text"* name=*"prix"*/> </td>  </tr><tr>  <td>Etat</td>  <td><input type=*"text"* name=*"etat"*/> </td>  </tr><tr>  <td colspan=*"2"*><input type=*"submit"* value=*"valider"*/> </td>  <td> </td>  </tr>  </table>  </form>  <table border=*"1"* width=*"60%"*>  </body>  </html> |

****

**Controleur**

**Le controleur est représenté par une servlet dans le pattern MVC**

**Etape 2. Ecrire une servlet permettant de saisir les données du fomulaire et l'afficher dans le navigateur**

|  |
| --- |
| **package** fsa.ac.ma;  **import** java.io.IOException;  **import** java.io.PrintWriter;  **import** java.util.Iterator;  **import** javax.servlet.ServletException;  **import** javax.servlet.annotation.WebServlet;  **import** javax.servlet.http.HttpServlet;  **import** javax.servlet.http.HttpServletRequest;  **import** javax.servlet.http.HttpServletResponse;  **import** metier.Produit;  **import** fsa.ac.ma.ProduitBeans;  **import** metier.services;  @WebServlet("/prodServ")  **public** **class** ProduitServlet **extends** HttpServlet {  **private** **static** **final** **long** *serialVersionUID* = 1L;    **protected** **void** doPost(HttpServletRequest request, HttpServletResponse response) **throws** ServletException, IOException {      String nom=request.getParameter("nom");  String desc=request.getParameter("desc");  String prix=request.getParameter("prix");  String etat=request.getParameter("etat");    PrintWriter pr=response.getWriter();  pr.println("<html>"+ "<head>" +"<title>"+"MVC Application"+"</title>"+"</hread> <body>"+  nom + "," + desc + "," + prix + "," + etat + "," +  "</body>"+"</html>");}} |

**Il faut ajouter la balise form dans le fichier Produit jsp:**

|  |
| --- |
| <form action=*"prodServ"* method=*"post"*> |

**Ici on recommonde de donner, pour des questions de sécurité, un autre nom au servlet par exemple "prodServ" au lieu de** ProduitServlet

**Il faut que cette modification soit prise en compte dans le fichier de déploiement web.xml au niveau de la balise <servlet-mapping>**

|  |
| --- |
| <?xml version=*"1.0"* encoding=*"UTF-8"*?>  <web-app xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"* xmlns=*"http://java.sun.com/xml/ns/javaee"* xmlns:web=*"http://java.sun.com/xml/ns/javaee/web-app\_2\_5.xsd"* xsi:schemaLocation=*"http://java.sun.com/xml/ns/javaee http://java.sun.com/xml/ns/javaee/web-app\_3\_0.xsd"* id=*"WebApp\_ID"* version=*"3.0"*>  <display-name>GestionsArticle</display-name>  <servlet>  <servlet-name>ProduitServlet</servlet-name>  <servlet-class>fsa.ac.ma.ProduitServlet</servlet-class>  </servlet>  <servlet-mapping>  <servlet-name>ProduitServlet</servlet-name>  <url-pattern>/prodServ</url-pattern>  </servlet-mapping>  <welcome-file-list>  <welcome-file>index.html</welcome-file>  <welcome-file>index.htm</welcome-file>  <welcome-file>index.jsp</welcome-file>  <welcome-file>default.html</welcome-file>  <welcome-file>default.htm</welcome-file>  <welcome-file>default.jsp</welcome-file>  </welcome-file-list>  </web-app> |

**3 étape exécuter votre projet.**

**Couche Metier**

**La couche métier est représentée par la classe services.java permettant de créer, ajouter, modifier et supprimer les objet à partir de la classe Javabean Produit ( une classe POJO). La classe services.java peut aussi ajouter, modifier et supprimer les objet dans un contenneur Liste ArrayList. Commençant d'abord par développer la classe Produit :**

|  |
| --- |
| **package** metier;  **public** **class** Produit {  **private** Long id; // utiliser l'objet Long et non pqs les le primitive long pour // l'utiliser dans le mapping avec le framework hibernate par exemple  **private** String nom, desc;  **private** **int** prix,etat;  **public** Produit() {  **super**();  // **TODO** Auto-generated constructor stub  }  **public** Produit(Long id, String nom, String desc, **int** prix, **int** etat) {  **super**();  **this**.id = id;  **this**.nom = nom;  **this**.desc = desc;  **this**.prix = prix;  **this**.etat = etat;}  **public** Produit(String nom, String desc, **int** prix, **int** etat) {  **super**();  **this**.nom = nom;  **this**.desc = desc;  **this**.prix = prix;  **this**.etat = etat;  }  **public** Long getId() {**return** id;}  **public** **void** setId(Long id) {**this**.id = id;}  **public** String getNom() {**return** nom;}  **public** **void** setNom(String nom) {**this**.nom = nom;}  **public** String getDesc() {**return** desc;}  **public** **void** setDesc(String desc) {**this**.desc = desc;}  **public** **int** getPrix() {**return** prix;}  **public** **void** setPrix(**int** prix) {**this**.prix = prix;}  **public** **int** getEtat() {**return** etat;}  **public** **void** setEtat(**int** etat) {**this**.etat = etat;}  @Override  **public** String toString() {  **return** "Produit [id=" + id + ", nom=" + nom + ", desc=" + desc + ", prix="  + prix + ", etat=" + etat + "]";}  **public** **void** show(){System.*out*.println(toString());}  } |

**Passons ensuite au développement de la classe metier services.java**

|  |
| --- |
| package metier;  import java.sql.Connection;  import java.sql.DriverManager;  import java.sql.ResultSet;  import java.util.ArrayList;  public class services {  private ArrayList<Produit> produit=new ArrayList<Produit>();  public ArrayList<Produit> getProduit() {return produit;}  public void setProduit(ArrayList<Produit> produit){this.produit = produit;}  public void add(Produit p){produit.add(p);}  public void remove (Long id)  { for(Produit p:produit)  if (p.getId()==id){produit.remove(p); break;}  }  //affiche la liste des produits  public ArrayList<Produit> getAll(){  return produit;  } |

**Avant de continuer, nous allons tester notre classe metier services pour cela nous allons créer une classe TestServices.java**

|  |
| --- |
| **public** **class** TestServices {  **public** **static** **void** main(String[] args) {  services serv=**new** services();    serv.add(**new** Produit(**new** Long(1),"PC", "PC Portable",8000,1));  serv.add(**new** Produit(2L,"CLAVIER", "PC Portable",200,1));  serv.add(**new** Produit(3L,"SOURIS", "PC Portable",80,1));  serv.add(**new** Produit(**new** Long(4),"USB", "PC Portable",30,1));  Iterator<Produit> it\_listProduit= serv.getAll().iterator();  **while**(it\_listProduit.hasNext()){  Produit prod = it\_listProduit.next();  prod.show();  }  }  } |

**Compiler et exécuter TestServices. Essayer d'ajouter et supprimer des objet à la liste.**

**Couche Modèle**

**Ajoutons dans la couche modele La classe Javabean ProduitBeans.java**

|  |
| --- |
| ProduitBeans**.java**  **package** fsa.ac.ma;  **import** java.util.ArrayList;  **import** metier.Produit;  **public** **class** ProduitBeans {  **private** Produit produit= **new** Produit();  **private** ArrayList<Produit> list= **new** ArrayList <Produit>();  **public** Produit getProduit() {  **return** produit;  }  **public** **void** setProduit(Produit produit) {  **this**.produit = produit;  }  **public** ArrayList<Produit> getList() {  **return** list;  }  **public** **void** setList(ArrayList<Produit> list) {  **this**.list = list;  }  } |

**Nous allons reprendre notre controlleur (Servlet ) est au lieu d'afficher les attribut de l'objet article dans le navigateur on va les enregistrer dans notre contenneur ArrayList. Pour cela il faut créer un objet Services et utiliser les méthodes services proposées par la classe services pour enregistrer les objet produits dans la liste.**

|  |
| --- |
| **package** fsa.ac.ma;  **import** java.io.IOException;  **import** java.io.PrintWriter;  **import** java.util.Iterator;  **import** javax.servlet.ServletException;  **import** javax.servlet.http.HttpServlet;  **import** javax.servlet.http.HttpServletRequest;  **import** javax.servlet.http.HttpServletResponse;  **import** javax.servlet.annotation.WebServlet;  **import** metier.Produit;  **import** fsa.ac.ma.ProduitBeans;  **import** metier.services;  @WebServlet("/prodServ")  **public** **class** ProduitServlet **extends** HttpServlet {  **private** **static** **final** **long** *serialVersionUID* = 1L;  **protected** **void** doPost(HttpServletRequest request, HttpServletResponse response) **throws** ServletException, IOException {    String nom=request.getParameter("nom");  String desc=request.getParameter("desc");  String prix=request.getParameter("prix");  String etat=request.getParameter("etat");    //Produit p= new Produit(1L,nom,desc,Integer.parseInt(prix),Integer.parseInt(etat));  // création d'une collection liste "prodBeans" pour sauvegarder les objets produits  ProduitBeans prodBeans= new ProduitBeans();  // Créer un service il s'agit ici du service "add" permettant d'ajouter un objet  services serv=new services();  serv.add(**new** Produit(1L,nom,desc,Integer.*parseInt*(prix),Integer.*parseInt*(etat)));  prodBeans.setList(serv.getAll())  **//transmettre l'objet collection rempli au page JSP "Produit.jsp" pour l'affichage pour cela on utilise l'objet**  **// request et la méthode setAttribute**  **//objet prodBeans est transmit sous le nom modele qui sera dispatché**  **request.setAttribute("modele", prodBeans); request.getRequestDispatcher("Produit.jsp").forward(request,response);**  **}}** |

**Mainteant pour récuppérer la liste des produits contenus dans la liste prodBeans transmis sous le nom modele dans le page Produit.JSP et l'afficher on va ajouter une autre table dans le fichier Produit.jsp.**

**On va modifier le fichier produit.jsp de manière à afficher la liste des produit das une table**

|  |
| --- |
| <%@page import=*"fsa.ac.ma.ProduitBeans"*%>  <%@page import=*"java.util.Iterator"*%>  <%@page import=*"metier.Produit"*%>  <%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"* pageEncoding=*"ISO-8859-1"*%>  <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">  <html>  <head>  <meta http-equiv=*"Content-Type"* content=*"text/html; charset=ISO-8859-1"*>  <title>GestionProduits</title>  </head>  <body>  // Nous allons récupèrer la liste d'objet nommé "modele" par la méthode getAttribut() // sur l'objet request  <%ProduitBeans produits=(ProduitBeans)request.getAttribute("modele");  %>  \*  \*  \*  \*  \*  \*  <table border=*"1"* width=*"60%"*>  <tr>  <th>Id </th>  <th> Nom</th>  <th> Descriptif</th>  <th> Prix</th>  <th> Etat</th>  <th> Option</th>  </tr>  <% Iterator <Produit> list=produits.getList().iterator();  **while**(list.hasNext())  **{**  Produit p=list.next();%>  <tr>  <td> <%= p.getId()%></td>  <td> <%= p.getNom()%></td>  <td> <%= p.getDesc()%></td>  <td> <%= p.getPrix()%></td>  <td> <%= p.getEtat()%></td>  <td>  <form action=*"prodServ"* method=*"post"*>  <input type=*"hidden"* name=*"id"* value=*"*<%= p.getId()%>*"*/>  <input type=*"hidden"* name=*"action"* value=*"supprimer"*/>  <input type=*"submit"* value=*"supprimer"*/>  </form>  </td>  </tr>  <%**}** %>  </table>  </body>  </html> |

**Note : il y 'aura un probleme qu niveau de synchronization entre remplissage de la liste et son affichage. Parfois on affiche la liste d'objet transmis qui n'est pas encore rempli**

**Pour remedier à ce problème on modifie code en instance l'objet produit directement à partir de la classe produit si il ya problème de transmission de la liste d'objet produit identifié par modele**

<%ProduitBeans produits;

**if**(request.getAttribute("modele")!=**null**){

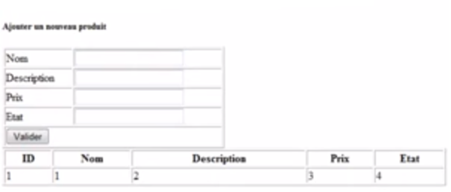
produits=(ProduitBeans)request.getAttribute("modele");}

**else**

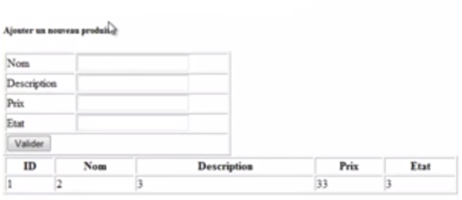
{ produits=**new** ProduitBeans();} // instanciation directe à partir d'une classe

%>

1iere Execution



**2 éme exécution**

****

**On remarque qu'il affiche juste le dernier objet et non pas tous les objets saisis. Ce problème est du au fait qu'on crée une liste au même moment que la creation d'un objet alors qu'il faut une seul liste suceptible de contenir plusieur objet. Pour corriger ce problème on va créer une seul instance de la classe services.java. Il faut rappeler que classe services.java propose d'ajouter des objets produits à une liste.**

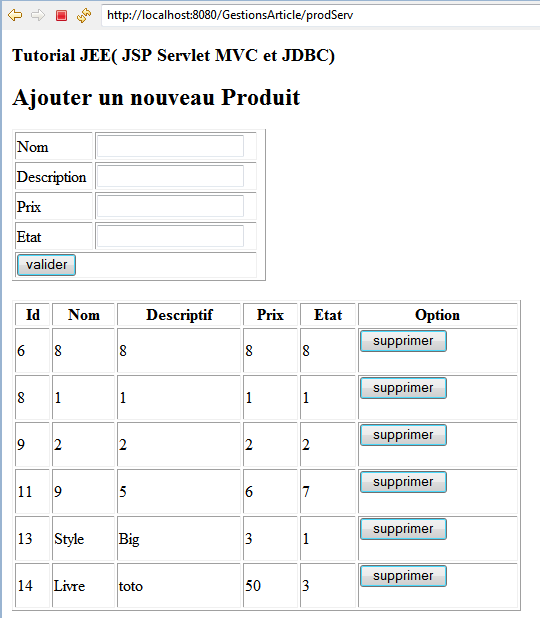
**Pour corriger ce problème on va instancier une seul fois l'objet services, pour cela on effectuer l'opération d'instanciation dans la méthode init() du servlet. On sait que la servlet appel la méthode init() une seul fois et non pas à chaque requête de l'utilisateur.**

**On va modifier le servlet de la façons suivante:**

|  |
| --- |
| **package** fsa.ac.ma;  **import** java.io.IOException;  **import** java.io.PrintWriter;  **import** java.util.Iterator;  **import** javax.servlet.ServletException;  **import** javax.servlet.http.HttpServlet;  **import** javax.servlet.http.HttpServletRequest;  **import** javax.servlet.http.HttpServletResponse;  **import** javax.servlet.annotation.WebServlet;  **import** metier.Produit;  **import** fsa.ac.ma.ProduitBeans;  **import** metier.services;  @WebServlet("/prodServ")  **public** **class** ProduitServlet **extends** HttpServlet {  **private** **static** **final** **long** *serialVersionUID* = 1L;  **private** services serv;  **public** **void** init() **throws** ServletException  {  serv=**new** services(); //un seul objet qui sera créé  }  **protected** **void** doPost(HttpServletRequest request, HttpServletResponse response) **throws** ServletException, IOException {    String nom=request.getParameter("nom");  String desc=request.getParameter("desc");  String prix=request.getParameter("prix");  String etat=request.getParameter("etat");    //Produit p= new Produit(1L,nom,desc,Integer.parseInt(prix),Integer.parseInt(etat));  // création d'une collection liste "prodBeans" pour sauvegarder les objets produits  ProduitBeans prodBeans= new ProduitBeans();  // Créer un service il s'agit ici du service "add" permettant d'ajouter un objet  // services serv=new services();  serv.add(**new** Produit(1L,nom,desc,Integer.*parseInt*(prix),Integer.*parseInt*(etat)));  prodBeans.setList(serv.getAll())  **//transmettre l'objet collection rempli au page JSP "Produit.jsp" pour l'affichage pour cela on utilise l'objet**  **// request et la méthode setAttribute**  **//objet prodBeans est transmit sous le nom modele qui sera dispatché**  **request.setAttribute("modele", prodBeans); request.getRequestDispatcher("Produit.jsp").forward(request,response);**  **}}** |

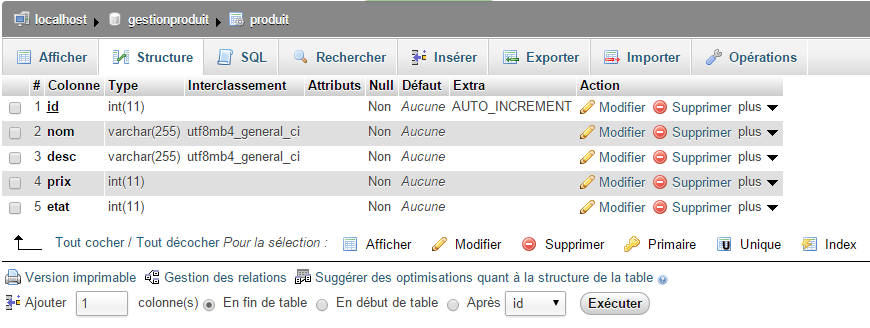
**On peut ajouter un champs Option proposant des possibilité de suprimer un objet produit. Pource la il faut modifier la partie présentation (vue) c'est-à-dire la modification du fichier Produits.jsp**

|  |
| --- |
| <table border=*"1"* width=*"60%"*>  <tr>  <th>Id </th>  <th> Nom</th>  <th> Descriptif</th>  <th> Prix</th>  <th> Etat</th>  <th> Option</th>  </tr>  <% Iterator <Produit> list=produits.getList().iterator();  **while**(list.hasNext()){Produit p=list.next();%>  <tr>  <td> <%= p.getId()%></td>  <td> <%= p.getNom()%></td>  <td> <%= p.getDesc()%></td>  <td> <%= p.getPrix()%></td>  <td> <%= p.getEtat()%></td>  <td>  <form action=*"prodServ"* method=*"post"*>  <input type=*"hidden"* name=*"id"* value=*"*<%= p.getId()%>*"*/>  <input type=*"hidden"* name=*"action"* value=*"supprimer"*/>  <input type=*"submit"* value=*"supprimer"*/>  </form>  </td>  </tr>  <%} %>  </table> |

****

**Enregistrement des objets Produits dans la base de données on utilisant le pont JDBC**

**Structure de la table produit de la base GestionProduit**

****

**Struture du projet**

|  |
| --- |
| package metier;  import java.sql.Connection;  import java.sql.DriverManager;  import java.sql.ResultSet;  import java.util.ArrayList;  import com.mysql.jdbc.PreparedStatement;  public class services {  private ArrayList<Produit> produit=new ArrayList<Produit>();  public ArrayList<Produit> getProduit() {  return produit;  }  public void setProduit(ArrayList<Produit> produit)  {this.produit = produit;}  public void add(Produit p)  {try {  Class.forName("com.mysql.jdbc.Driver");  Connection cn = (Connection) DriverManager.getConnection("jdbc:mysql://localhost:3306/gestionproduit","root","");  PreparedStatement pr= (PreparedStatement) cn.prepareStatement("INSERT INTO produit VALUE (NULL,?,?,?,?)");  pr.setString(1, p.getNom());  pr.setString(2, p.getDesc());  pr.setInt(3, p.getPrix());  pr.setInt(4, p.getEtat());  pr.execute();  } catch (Exception e) {  // TODO Auto-generated catch block  e.printStackTrace();  }  produit.add(p);  }  public void remove (Long id)  {try {  Class.forName("com.mysql.jdbc.Driver");  Connection cn = (Connection) DriverManager.getConnection("jdbc:mysql://localhost:3306/gestionproduit","root","");  PreparedStatement pr= (PreparedStatement) cn.prepareStatement("DELETE FROM produit where id = ?");  pr.setLong(1, id);  pr.execute();  } catch (Exception e) {  // TODO Auto-generated catch block  e.printStackTrace();  }      for(Produit p:produit)  {if (p.getId()==id){produit.remove(p); break;}  }  }  public Produit getProduitDetails(Long id)  { for(Produit p:produit)  if (p.getId()==id) {return p;}  return null;  }  public ArrayList<Produit> getAll(){  ArrayList<Produit> listproduit=new ArrayList<Produit>();  try {  Class.forName("com.mysql.jdbc.Driver");  Connection cn = (Connection) DriverManager.getConnection("jdbc:mysql://localhost:3306/gestionproduit","root","");  PreparedStatement pr= (PreparedStatement) cn.prepareStatement("SELECT \* FROM produit");  ResultSet rs= pr.executeQuery();  while(rs.next()){  Produit p=new Produit();  p.setId(rs.getLong("id"));  p.setNom(rs.getString("nom"));  p.setDesc(rs.getString("desc"));  p.setPrix(rs.getInt("prix"));  p.setEtat(rs.getInt("etat"));  listproduit.add(p);  }    } catch (Exception e) {  // TODO Auto-generated catch block  e.printStackTrace();  }    return listproduit;}  } |

Note : N'oublier pas d'jouter le driver Msql ou Oracle dans le répertoire lib de tomcat