



المدرسة العليا
للتكنولوجيا - الصويرة
L'ÉCOLE SUPÉRIEURE DE
TECHNOLOGIE – ESSAOUIRA

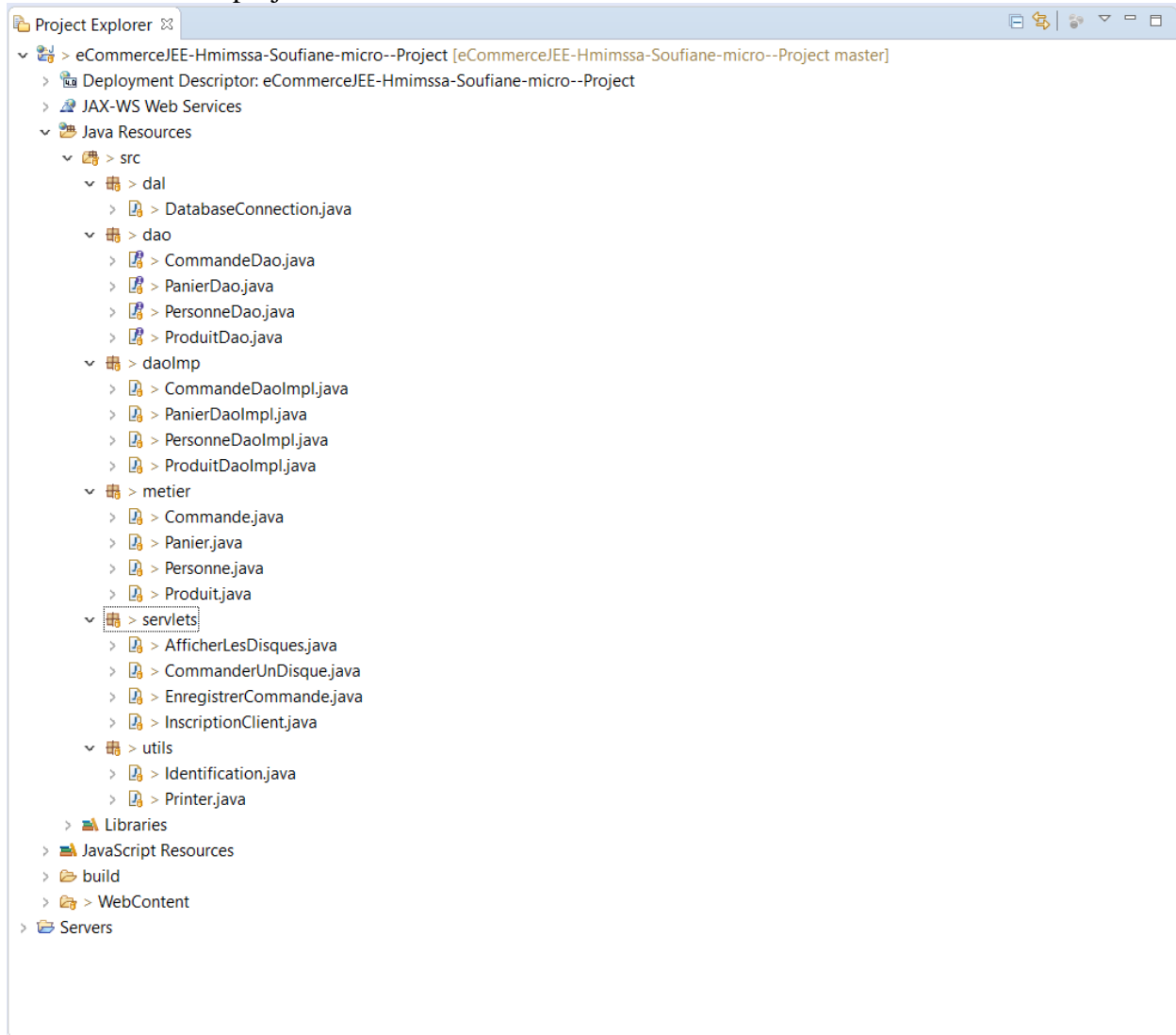
Micro Projet

Gestion Magasin

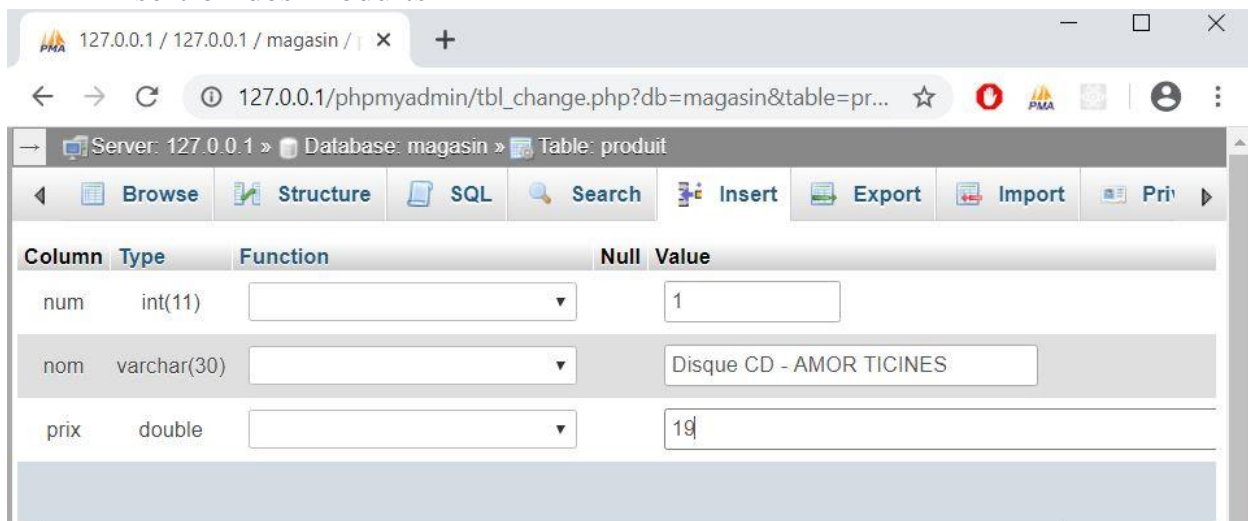
Hmimssa Soufiane

2020/2021

- Gestion de projet



- Insertion des Produits



Servlet InscriptionClient.Java

Inscription et authentification est dans la meme Servlet qui permet à un client de créer un compte ou bien de se connecter au site web.

Au premie temps on vérifier et initialiser les cookies et les paramètres si elle existe

```
24 String nomRecu = null, motPasseRecu = null;
25 String nomCookie = null, motPasseCookie = null;
26 Cookie[] cookies = request.getCookies();
27 if (cookies != null)
28     for (Cookie cookie : cookies) {
29         if (cookie.getName().equals("nomCookie")) {
30             nomCookie = cookie.getValue();
31             log("cookie path " + cookie.getPath());
32         }
33         if (cookie.getName().equals("motPasseCookie")) {
34             motPasseCookie = cookie.getValue();
35         }
36     }
37 nomRecu = request.getParameter("nom");
38 motPasseRecu = request.getParameter("motdepasse");
39 if (nomRecu == "" || motPasseRecu == "") {
40     motPasseRecu = null;
41     nomRecu = null;
42 }
43 }
44
```

On cas où il n'y a ni de cookies ni de parametres

```
45 if (nomCookie == null && nomRecu == null) {
46     // Cas 1 : cas où il n'y a ni de cookies ni de parametres
47     out.println("<html>");
48     out.println("<body>");
49     out.println("<head>");
50     out.println("<title> Inscription d'un client </title>");
51     out.println("</head>");
52     out.println("<body bgcolor='white' >");
53     out.println(nomRecu + " | " + motPasseRecu + " | " + nomCookie + " | " + motPasseCookie);
54     out.println("<h1> Inscription : </h1>");
55     out.println("<h4> Attention mettre nom et le mot de passe avec plus de 3 caracteres </h4>");
56     out.println("<form action='Inscription' method='GET' >");
57     out.println("<table>");
58     out.println("<tr><td>Nom</td><td> <input type='text' size='20' name='nom' ></td></tr>");
59     out.println("<tr><td>Mot de passe</td><td> <input type='password' size='20' name='motdepasse' ></td></tr>");
60     out.println("<tr><td><input type='submit' value='Inscription'></td></tr>");
61     out.println("</table>");
62     out.println("</form>");
63     out.println("</body>");
64     out.println("</html>");
65 }
```

Resultat

Inscription d'un client

http://localhost:8080/eCommerce/JEE/Inscription

null | null | null | null

Inscription :

Attention mettre nom et le mot de passe avec plus de 3 caracteres

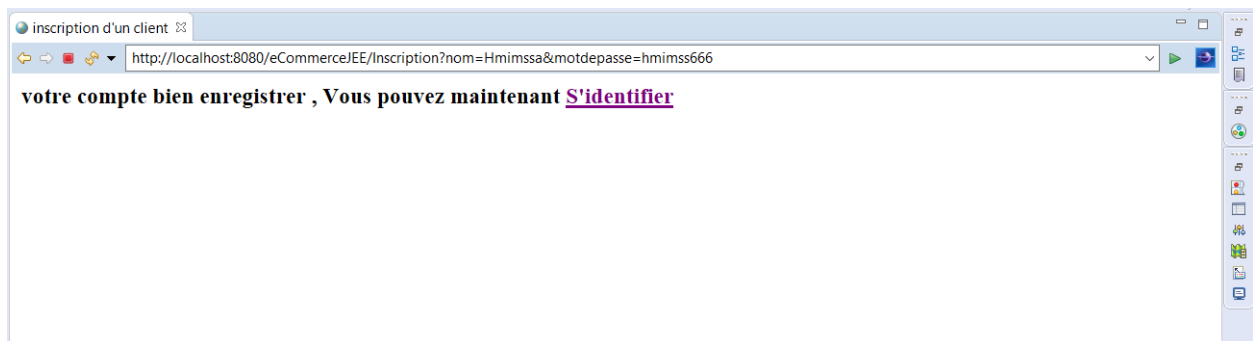
Nom

Mot de passe

Cas 2 : cas où il n'y a pas de cookies mais les paramètres nom et mot de passe sont présents on crée les cookies:

```
66     else if (nomCookie == null && nomRecu != null && motPasseRecu != null) {
67         // Cas 2 : cas où il n'y a pas de cookies mais les paramètres nom et mot de
68         // passe sont présents :
69
70         response.addCookie(new Cookie("nomCookie", nomRecu));
71         response.addCookie(new Cookie("motPasseCookie", motPasseRecu));
72         out.println("<html>");
73         out.println("<body>");
74         out.println("<head>");
75         out.println("<title> inscription d'un client </title>");
76         out.println("</head>");
77         out.println("<body bgcolor='white' >");
78         out.println(
79             "<h3> votre compte bien enregistré , Vous pouvez maintenant "
80             + "<a href=\"/eCommerceJEE/Inscription\">S'identifier</a></h3>");
81         out.println("</body>");
82         out.println("</html>");
83     }
84 }
```

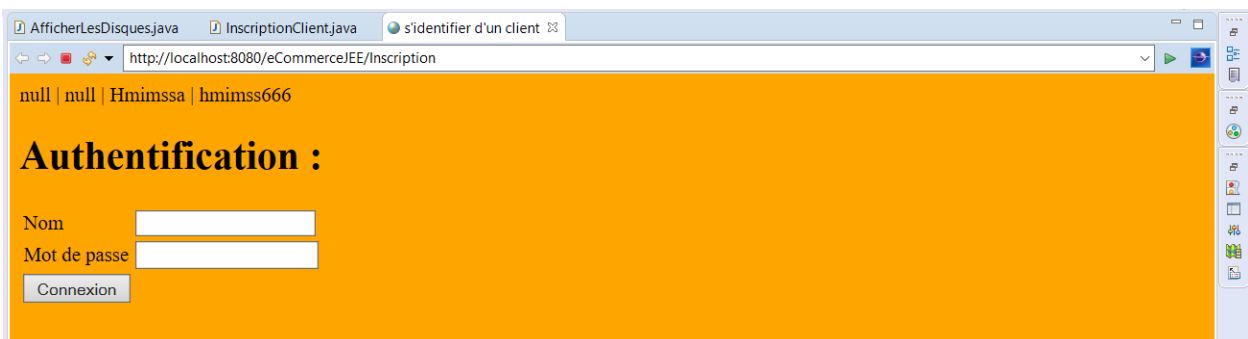
Le résultat



Cas 4 : Si les cookies sont présents demande de s'identifier

```
94     else if (cookies != null) {
95         // Cas 4 : les cookies sont présents demande de s'identifier
96
97         out.println("<html>");
98         out.println("<body>");
99         out.println("<head>");
100        out.println("<title> s'identifier d'un client </title>");
101        out.println("</head>");
102        out.println("<body bgcolor='orange' >");
103        out.println(nomRecu + " | " + motPasseRecu + " | " + nomCookie + " | " + motPasseCookie);
104        out.println("<h1>" + "Authentification : " + "</h1>");
105        out.println("<form action='Inscription' method='get' >");
106
107        out.println("<table>");
108        out.println("<tr><td>Nom</td><td> <input type='text' size='20' name='nom' > </td></tr>");
109        out.println("<tr><td>Mot de passe</td><td> <input type='password' size='20' name='motdepasse' > </td></tr>");
110        out.println("<tr><td><input type='submit' value='Connexion'></td></tr>");
111
112        out.println("</table>");
113        out.println("</form>");
114        out.println("</body>");
115        out.println("</html>");
116    }
```

Le résultat



Cas 5 : après le client click sur Connexion on vérifie ses données on appelle le servlet achat avec la création d'une session.

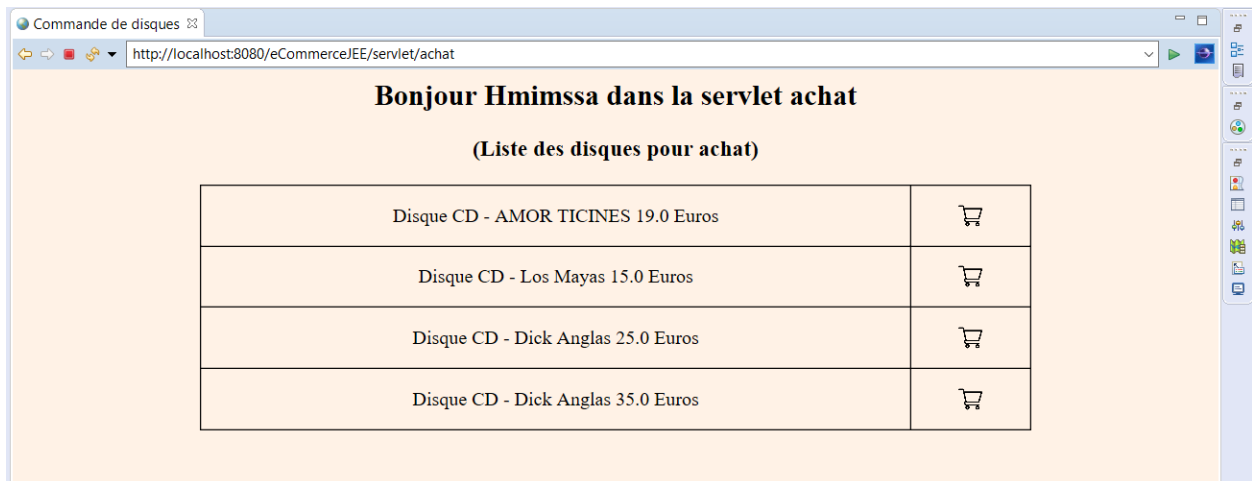
```
85     else if ((identique(nomRecu, nomCookie) && identique(motPasseRecu, motPasseCookie))) {  
86         // Cas 3 : cas où le nom et le mot passe sont correctes, appel à la servlet  
87         // achat  
88  
89         // Create session and redirect to Achat Servlet  
90         request.getSession(true);  
91         response.sendRedirect("/eCommerceJEE/servlet/achat");  
92  
93     }
```

Servlet AfficherLesDisques

Cette servlet il vérifier si le requête contient une session sinon il fait l'appel a la Servlet InscriptionClient.

```
27 protected void doGet(HttpServletRequest request, HttpServletResponse response)
28     throws ServletException, IOException {
29     response.setContentType("text/html");
30     PrintWriter out = response.getWriter();
31
32     String nom = null;
33     Cookie[] cookies = request.getCookies();
34     nom = Identification.chercheNom(cookies);
35
36     if (request.getSession(false) != null) {
37
38         List<Produit> produits = new ProduitDaoImpl().getProduits();
39
40         out.println("<html>");
41         out.println("<body>");
42         out.println("<head>");
43         out.println("<title> Commande de disques </title>");
44         out.println("<style>" + "th {background-color: #cc6300;color: white;}"
45             + " table, td, th { border: 1px solid;text-align: center;}"
46             + "table { border-collapse: collapse; width: 70%}" + "th, td {padding: 15px;}" + "</style>");
47         out.println("</head>");
48         out.println("<body style='\"background-color:#fff2e6;\">");
49         out.println("<center>");
50         out.println("<h2>" + "Bonjour " + nom + " dans la servlet achat" + "</h2>");
51         out.println("<h3>(Liste des disques pour achat)</h3>");
52         Printer.vente(out, produits);
53         out.println("</center>");
54         out.println("</body>");
55         out.println("</html>");
56     } else {
57         response.sendRedirect("servlets/InscriptionClient");
58     }
59 }
60 }
```

Si le requête contient une session il fait l'appel couche DaoImp qui fait une liaison entre la couche métier et la base de donnée.







Commande de disques

http://localhost:8080/eCommerceJEE/servlet/achat

Bonjour Hmimssa dans la servlet achat

(Liste des disques pour achat)

Disque CD - AMOR TICINES 19.0 Euros	
Disque CD - Los Mayas 15.0 Euros	
Disque CD - Dick Anglas 25.0 Euros	
Disque CD - Dick Anglas 35.0 Euros	

Bonjour soufiane dans la servlet achat

(Votre panier)

Informations disque commandes
Disque CD - AMOR TICINES disque897TR566 15.0 euros
Disque CD - Frederic Angonas disque98HUYU56 35.0 euros
Disque CD - Dick Anglas disque87YHG564 25.0 euros

Votre panier contient : 3 disque(s)

[Vous pouvez commandez un autre disque](#)

[Vous pouvez enregistrer votre commande](#)

Couche metier

Exemple de couche métier

```
Produit.java
1 package metier;
2
3 public class Produit {
4
5     int num;
6     String nom;
7     Double prix;
8
9
10
11     public Produit() {
12         super();
13     }
14     public Produit(int num, String nom, Double prix) {
15         super();
16         this.num = num;
17         this.nom = nom;
18         this.prix = prix;
19     }
20     public int getNum() {
21         return num;
22     }
23     public void setNum(int num) {
24         this.num = num;
25     }
26     public String getNom() {
27         return nom;
28     }
29     public void setNom(String nom) {
30         this.nom = nom;
31     }
32     public Double getPrix() {
33         return prix;
34     }
35     public void setPrix(Double prix) {
36         this.prix = prix;
37     }
38     @Override
39     public String toString() {
40         return "Produit [num=" + num + ", nom=" + nom + ", prix=" + prix + "];";
41     }
42 }
43
```

Dao pattern

Exemple de Dao pattern

```
Produit.java  ProduitDao.java
1 package dao;
2
3 import java.util.List;
4
5
6
7 public interface ProduitDao {
8
9     public List<Produit> getProduits();
10    public Produit getProduit(int id);
11    public void addProduit(Produit P);
12    public void updateProduit(Produit P);
13    public void deleteProduit(String id);
14 }
15
```

Exemple de l'implémentation Dao pattern


```

14 public class ProduitDaoImpl implements ProduitDao {
15     Statement statement;
16     ResultSet resultSet;
17     DatabaseConnection databaseConnection;
18     String message = "l'opération est terminée avec succès";
19     public ProduitDaoImpl() {
20         databaseConnection = DatabaseConnection.getInstance();
21     }
22     @Override
23     public List<Produit> getProduits() {
24         List<Produit> Produits = new ArrayList<Produit>();
25
26         try {
27             statement = databaseConnection.getConnection().createStatement();
28             resultSet = statement.executeQuery("select * from Produit");
29
30             while (resultSet.next()) {
31                 Produit c = new Produit(resultSet.getInt(1), resultSet.getString(2), resultSet.getDouble(3));
32                 Produits.add(c);
33             }
34         } catch (SQLException e) {
35             e.printStackTrace();
36         }
37         return Produits;
38     }
39     @Override
40     public Produit getProduit(int num) {
41         Produit Produit = null;
42         try {
43             statement = databaseConnection.getConnection().createStatement();
44             PreparedStatement preparedStatement = databaseConnection.getConnection()
45                 .prepareStatement("SELECT * FROM `produit` WHERE num=?");
46             preparedStatement.setInt(1, num);
47             resultSet = preparedStatement.executeQuery();
48
49             if (resultSet.next()) {
50                 Produit = new Produit(resultSet.getInt(1), resultSet.getString(2), resultSet.getDouble(3));
51             }
52         } catch (SQLException e) {
53             e.printStackTrace();
54         }
55         return Produit;
56     }
57     @Override
58     public void addProduit(Produit Produit) {
59         try {
60             String queryString = "INSERT INTO `Produit`(`nom`) VALUES (?)";
61             PreparedStatement preparedStatement = databaseConnection.getConnection().prepareStatement(queryString);
62             preparedStatement.setString(1, Produit.getNom());
63             int rowsInserted = preparedStatement.executeUpdate();
64             if (rowsInserted > 0) {
65             }
66         } catch (SQLException e) {
67             e.printStackTrace();
68         }
69     }
70     @Override
71     public void updateProduit(Produit Produit) {
72         try {
73             String queryString = "UPDATE `produit` SET `nom`=?,`prix`=? WHERE num=?";
74             PreparedStatement preparedStatement = databaseConnection.getConnection().prepareStatement(queryString);
75             preparedStatement.setString(1, Produit.getNom());
76             preparedStatement.setDouble(2, Produit.getPrix());
77             preparedStatement.setInt(3, Produit.getNum());
78
79             int rowsUpdated = preparedStatement.executeUpdate();
80             if (rowsUpdated > 0) {
81             }
82         } catch (SQLException e) {
83             e.printStackTrace();
84         }
85     }
86     @Override
87     public void deleteProduit(String id) {
88         try {
89             String queryString = "DELETE FROM `Produit` WHERE num=?";
90             PreparedStatement preparedStatement = databaseConnection.getConnection().prepareStatement(queryString);
91             preparedStatement.setString(1, id);
92             int rowsDeleted = preparedStatement.executeUpdate();
93             if (rowsDeleted > 0) {
94             }
95         } catch (SQLException e) {
96             e.printStackTrace();
97         }
98     }
99 }

```

L'implémentation Dao pattern demandé la connexion de couche DAL.

```
1 package dal;
2
3 import java.sql.Connection;
4
5 @SuppressWarnings("serial")
6 public class DatabaseConnection extends HttpServlet {
7     private Connection connection;
8     private static DatabaseConnection Instance;
9
10    private DatabaseConnection() {
11        try {
12            //String driver = getServletContext().getInitParameter("jdbc");
13            //String databaseURL = getServletContext().getInitParameter("localisation");
14            Class.forName("com.mysql.jdbc.Driver");
15            connection = DriverManager.getConnection("jdbc:mysql://localhost:3306/magasin","root","");
16            System.out.println("Base de donnees correctement connecte !!");
17        } catch (Exception e) {
18            e.printStackTrace();
19        }
20    }
21
22    public Connection getConnection() {
23        return connection;
24    }
25
26    public static DatabaseConnection getInstance() {
27        if (Instance == null)
28            return new DatabaseConnection();
29        return Instance;
30    }
31 }
32
33
34
35
36
```

Package utils

fournir des classes qui en utiliser en autre classes

```

> > utils
> > Identification.java
> > Printer.java

```

```
1 package utils;
2
3 import javax.servlet.http.Cookie;
4
5 public class Identification {
6     public static String chercheNom(Cookie[] cookies) {
7         if (cookies != null)
8             for (Cookie cookie : cookies)
9                 if (cookie.getName().equals("nomCookie"))
10                    return cookie.getValue();
11
12         return "inconnu";
13     }
14 }
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100

```

```
1 package utils;
2
3 import java.io.PrintWriter;
4
5 public class Printer {
6     public static void vente(PrintWriter out, List<Produit> produits) {
7         out.println("<table border=1>");
8         for (Produit produit : produits) {
9             out.println("<tr><td>" + produit.getNom() + " " + produit.getPrix() + " Euros </td>");
10            out.println("<td><a href='\"comande\"'>");
11            out.println("<img src='\"L:\\JEE2\\eCommerceJEE-Hmimssa-Soufiane-micro--Project\\img\\supermarket.png' width='20' border="
12            out.println("</table> </form>");
13        }
14    }
15 }
16
17
18
19

```