GCR2

TD3- JEE
Servlet – JSP – JSTL - JDBC

Énoncé

On souhaite créer une application de gestion d'un emploi du temps simple avec les servlets et les pages Web JSP en se basant sur ***le pattern DAO***. Pour ce faire on doit suivre les étapes suivantes :

Étape 1 : Développement de la couche model

- Créer la classe **Enseignant** caractérisée par : ide (de type entier) et nom (de type String)
- Créer la classe **Matiere** caractérisée par : idmat (de type entier), nom (de type String) et l'enseignant concerné.
- Créer la classe **Seance** caractérisée par : ids (de type entier), nums (de type String), jour (de type String) et la Matière et l'enseignant concernés.

Étape 2 : Développement de la couche dao

- Créer l'interface générique **IGestionEmploi**, cette interface contient la déclaration des méthodes :
  - `List <Enseignant> getAllEnseignant ()` : récupère la liste des enseignants de la BDD.
  - `List <Matiere> getAllMatiere ()` : récupère la liste des matières de la BDD.
  - `List <Seance> getAllSeance ()` : récupère la liste des matières de la BDD.
  - `Seance addSeance(Enseignant e, Matiere m, int nums , int jour)` : Permet d'ajouter une séance à la BDD,
  - ...
- Créer la classe **GestionEmploiImp** qui implémente l'interface **IGestionEmploi** et redéfinissez toutes les méthodes de l'interface.

CamScanner

Étape 3 : Développement de la couche contrôleur et vue

1. Développement de la page JSP pour ajouter une séance :
   a. La page Web « **gestion.jsp** » permettant d'ajouter une nouvelle séance à la base de données et d'afficher l'emploi du temps :
   Lorsque l'utilisateur sélectionne le nom de l'enseignant, le nom de la matière, le numéro de la séance et le nom du jour et il clique sur le bouton « **ajouter** » les données seront transférées vers la Servlet « **AjoutSeance** ». Cette dernière, permet d'ajouter une séance dans la base de données et elle renvoie comme résultat l'affichage de l'emploi du temps avec toutes les séances programmées dans la semaine encours.
   b. Vous devez gérer les conflits des enseignants et des séances dans la même journée.

2. Affichage de l'emploi du temps : se fait dans la même page Web en affichant que les séances programmées et vide pour les autres.

Travail demandé : (voir ANNEXE 1)

1. Implémenter les classes, les méthodes de l'interface IGestionEmploi et la servlet susmentionnée.
2. Apporter les **ajouts nécessaires** dans la page Web « getion.jsp ».
3. Les données sont stockées dans une BDD MySQL nommée « emploi ».

ANNEXE 1

JEE

`pageContext` : utilisé pour partager directement des variables entre des pages JSP ou dans la même page JSP à travers les méthodes `getAttribute` et `setAttribute`

CamScanner

- List <Matiere> getAllMatiere(){

List <Matiere> LM = new ArrayList();

try {

preparedStatement ps = conn.prepareStatement("Select * From Matiere");

ResultSet res = ps.executeQuery();

While (res.next()) {

idmat, mon, idens

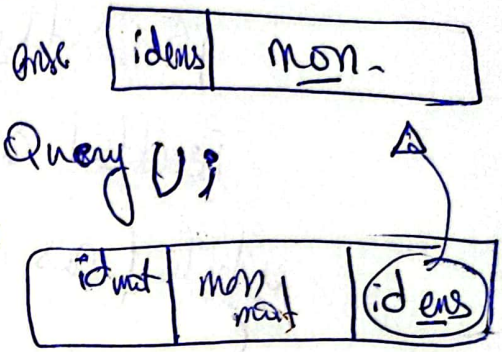
Matiere M = new Matiere(res.getInt(1), res.getString(2), res.getInt(3));

LM.add(M);

} catch (exception e) { e.printStackTrace(); }

return LM;

}



- List <Matiere> getMatiereById (int id) {

List <Matiere> Lm = new ArrayList();

try { preparedStatement ps = conn.prepareStatement("Select * From Matiere where id = ?");

ps.setInt(1, id);

ResultSet res = ps.executeQuery();

while (res.next()) { Matiere m = new Matiere(res.getInt(1), res.getString(2), res.getInt(3)); Lm.add(m); }

```
List<Enseignant> getEnseignantbyid(int id){
List<Enseignant>le = new ArrayList();
try
{ preparedStatement ps = Connection.prepareStatement("select *
                                From enseignant where id=?")
  ps.setInt(1,id);
  ResultSet res = ps.executeQuery();
while(res.next())
{
Enseignante e = new enseignant(res.getInt(1); res.getString(2);
  le.add(e)
} catch (Exception e){ e.printStackTrace();}

return le;
}
```

getId()
setId()
getNom()
setNom()

getAllenseignantbyid.nom(enseignant e){
try{ ps = new preparedStatement("selec* from enseignant where id=? and nom=?")
ps.setInt(1, e.getInt());
ps.setString(2, e.getNom());

List<Seance> getAllSeance() {

List<Seance> ls = new arrayList<>;

try {

prepareStatement ps = con.prepared Statement ("Select * From Seance")

ResultSet res = ps.executeQuery();

Enseignant e = getEnseignantById (res.getInt(4))

Matiere m = getMatiereById(res.getInt(5))

Seance s = new Seance (res.getInt(1), res.getInt(2), res.getString(3), m)

ls.add(s) }

{

Return ls; }

Seance addSeance (Enseignant e, Matiere m, int nums, int jour)

List <Seance> ls = new arraylist();

try {

preparedStatement ps = con.prepared ("insert into seance (nums, jour, ide, idm) Values (?,?,?,?)")

ps.setInt(1, nums)

ps.setInt(2, jours)

ps.setInt(3, e.getide())

ps.setInt(4, m.getIdm());

Result Se

Seance addSeance (int nums, String jour, Enseignant e, Matiere m).

List<Seance> ls = new arraylist();

try {

PrepareStatement ps = con.preparedStatement ("insert into Seance (nums, jour, ide, idm)")

ps.setInt(1, nums)

ps.setString(2, jour)

ps.setInt(3, getIde())

ps.setInt(4, getIdm)

ResultSet res = ps.execute();

while(res.next())

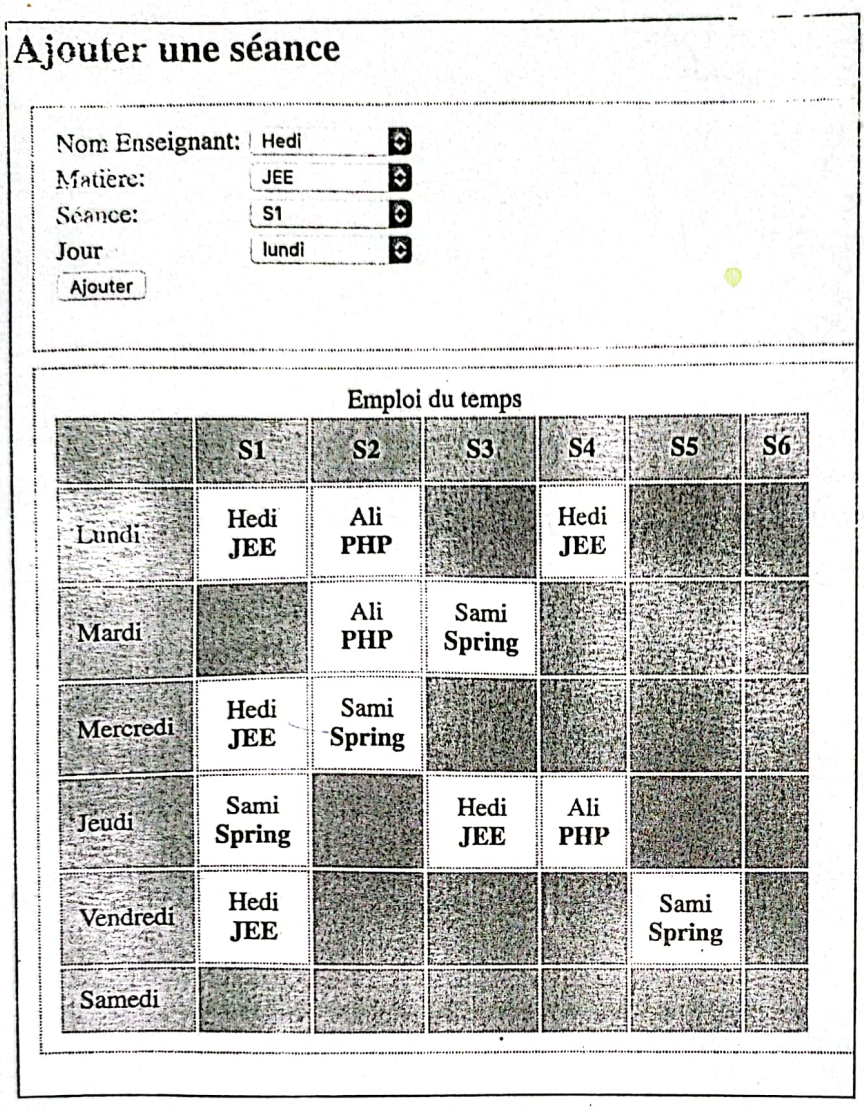
{ Seance s = new seance ((res.getInt(1), res.getString(2), res.getInt(3), res.getInt(4))

ls.add(s) }

catch (exception) { printStactrace(e))

Return ls

Pages web Principale : « gestion.jsp »



Code JSP

```
<html> <head> <title>Gestion Emploi</title>
</head> <body>
<h2> Ajouter une séance</h2>
<div> <form action= … method= … >

<table> <tr> <td> Nom Enseignant: </td>
<td><select name="…" class="select">
      <c:...>
        <option value= … > … </option>
      </c:… >
    </select></td> </tr>
<tr><td> Matière: </td>
```


CamScanner

```
<td><select name= … >
      <c:… >
       <option value= … >… </option>
      </c:… >
     </select></td> </tr>
<tr><td> Séance: </td>
<td><select name="seanc">
       ….
     </select> </td> </tr>
<tr><td>Jour</td>
<td><select name="jour">
      ….
     </select></td></td> </tr>
<tr><td>  <input type="submit" value="Ajouter"/></td></tr>
</table> </form> </div>
```

```
<div> <table class="table1">
      <CAPTION> Emploi du temps </CAPTION>
<tr><th></th><th>S1</th><th>S2</th><th>S3</th><th>S4</th><th>S5
</th><th>S6</th></tr>

<c:… >
 <tr><td style="background-color:pink;"> … </td>
      <c:… >
        …
      </c:… > </tr>
</c:… >
</table></div></body></html>
```

CamScanner

TD3 JEE

Etape 1 : Package Model

Public class Enseignant {

Private int ide ;
Private String Nom ;

}

Public Enseignat ( ) ;
Public Enseignat (int ide, String Nom) {
this.ide = ide ;
this.Nom Nom }

// les getters, setters

Public class Matiere {
Private int idmat ;
String Nom ;
Enseignante ;
}
Public Matière (int idmat, String Nom
Enseignant e)
{ this.idmat = idmat
this.e;
}

Public class Seance {
int ids, nums ;
String jour ;
Matière m ;
enseignat e ; }

Package DAO :

AO

- les interfaces

Public Interface IGestionEnsei {
déclaration
// les méthodes
list <Enseignant> getEnseignantsParMotCle (String mc);
<Matière> Matière getEnseignantById (int id);

- Implémentation de l'interface (les méthodes)

Public class GestionEnseiImp
Implements IGestion
Enseignant {
Public GestionEnseiImp
{ // connection
// à la base
// de donnée }
list <Enseignat> getAllEnseignat()
list <Ensignat> le = new ArrayList();
try {
PreparedStatement ps = null
conn.prepareStatement ("Select
* from Enseignant)
resultSet res = ps.executeQuery();
while

```
try {
Prepare Statement ps = con.prepared
Statement ("select * from Enseignant")
Result set res = ps.execute Query();
While (res.next) {
Enseignate = new Enseignat (res.get
Int(1), res.getString(2));
le.add(e);
} catch (Exception e)
{ e.printStackTrace();}
return le; }
list<Enseignd) getEnseignat by Id
list<Ens(int id)> le = new array list();
Connection con
try {
prepared Statement ps = con.prepared statement
("select * From Enseignat where id=?)
ps.setInt(1,id);
Result Set res = ps.Execute Query();
While (res.next()) {
{ Ensagnat e = new Ensagnat (res.getint(1),
res.getString(2))
le.add(e);
} catch (exception e)
{ e.printStackTrace();}
Return le;
```

```
list (Matière) getAllMatiere ();
{ list <Matiere> ln = new array
Enseignante; list();
try {
prepare Statement ps = con.prepared
Statement ("select * from Matière
ResultSet res = ps.executeQuery()
While (res.next()) {
Enseignat e = res.getEnseignatById(getInte
Matiere M = new Matiere (res.getInt(1)
res.getInt(2), res.getEnseignat(3) by
e Id
ln.add(M)) }
catch (exception e)
{ printStackTrace(e)
} return ln;
list <Stagiaire> getMatiere
By Id (int idm) {
list <Matiere> lm = New array
list ();
try {
prepare Statement ps = con.prepared
Statiments ("select * From
Matiere where id=?)
res ps.setInt(id,1)
Result Set res = ps.executeQuery();
While (res.next()) {
Matiere M = new Matiere (res.getInt(1),
res.getString(2), res.getInt
LM.add(M); } catch (exception
Retur lm; } { print Stack trace
```