

Sujet du TP2.

! on utilise html lorsque la page est statique et JSP lorsqu'on veut faire un traitement sur la page

index.html

```
<body>
    < a href = "ajoutPersonne.html"> ajouter personne </a>
    <a href = "ajouterAdresse.html"> ajouter Adresse </a>
    < ahref = "associer.jsp"> associer </a>
    < a href ="lister.jsp"> lister </a>
</body>
```

```
Public Class Personne {
    String nom;
    String prenom;
    Collection <Adresse> adresses = new ArrayList <Adresse>( );
    int id;

    public Collection<Adresse> getAdresses( ){
        return adresses;
    }

    public void setAdresses(Collection <Adresses> a){
        adresses = a;
    }
    ...
}
```

```
Public Class Adresse{
    int id;
    String rue;
    String ville;
    /* remember Léa les bdd associatives... pour ne pas ajouter un
    attribut personnes...*/
    //Getter et Setter
}
```

ajouterPersonne.html

```
<body>
  <form method="post" action="Serv">
    nom: <input type="text" name="nom" required>
    prenom: <input type="text" name="prenom" required>
    <input type="Submit" value="ok">
    <input type="hidden" name="op" value="ajoutPersonne">
  </form>
</body>
```

ajouterAdresse.html

```
<body>
  <form method="post" action="Serv">
    rue: <input type="text" name="rue" required>
    ville: <input type="text" name="ville" required>
    <input type="Submit" value="ok">
    <input type="hidden" name="op" value="ajoutAdresse">
  </form>
</body>
```

```
Public class Serv extends HttpServlet{
  public void doPost(HttpServlet request, HttpServlet response) throws
  ServletException, IOException{
    String op=request.getParameter("op");

    if (op.equals("ajoutPersonne")){
      String nom=request.getParameter("nom");
      String prenom = request.getParameter("prenom");
      facade.ajouterPersonne(nom, prenom);

      request.getRequestDispatcher("index.html") forward(request,
      response);
    }
    if (op.equals("ajoutAdresse")){
```

```

    ...
}
if (op.equals("choix")){
    request.setAttribute("listePersonnes", facade.getPersonne( ));
    request.setAttribute("listeAdresse", facade.getAdresses)( );
    request.getRequestDispatcher("choix.jsp") forward(request,
    response)
}

if (op.equals("associer")){
    int ida = request.getParameter("ida");
    int idp = request.getParameter("idp");
    facade.associer(idp, ida);
    request.getRequestDispatcher("index.html") forward(request,
    response);
}

if (op.equals("lister")){
    request.setAttribute("listePersonnes", facade.getPersonnes( ));
}
}

```

choix.jsp

```

<body>
    <form method="post" action="Serv">
        choisir la personne
        <%Collection<Personne> personnes =
        request.getAttribute("listePersonnes");
        for (Personne p:personnes){
            int idp = p.getId( );
            String np = p.getNom( ) + " " + p.getPrenom();
        >
        <input type="radio" name="idpersonne" value=<%=idp> <%=np>>
        <%=>

        <%Collection<Adresse> adresses =

```

```
request.getAttribute("listeAdresses");
for (Adresse a:adresses){
    int ida = p.getId( );
    String rv= a.getRue( ) + " " + a.getVille();
    >
    <input type="radio" name="idadresse" value=<%=ida> <%=rv>>
    <%=}>

    <input type="Submit" value="ok">
    <input type="hidden" name="op" value="choix">
</form>
</body>
```

lister.jsp

```
<body>
    <%Collection<Personne personnes =
request.getAttribute("ListePersonnes");
for (Personnes p:personnes){
    out.println(p.getnNom( ) + " " + p.get.Prenom( ));
    for (Adresse a:p.getAdresses( )){
        out.println("it" + a.getRue( ) + " " +a.getVille());
    }>
    <%=}>
</body>
```

```
Public Class Facade{
    int indexP = 0;
    int indexA = 0;
    HashMap<Integer, Personne> personnes = new HashMap<Integer,
    Personne> ( );
    HashMap<Integer, Adresse> adresses = new HashMap<Integer,
    Adresse> ( );

    public void ajouterPersonne(String nom, String prenom){
        Personne p=new Personne( );
```

```
    p.setId(index p);  
    p.setNom(nom);  
    p.setPrenom(prenom);  
    personnes.put(indexP, p);  
    indexP ++;  
}
```

```
Public Collection<Personne> ListePersonne( ){  
    return personnes.values( );  
}
```

```
Public Collection<Adresse> ListeAdresse( ){  
    return adresses.values( );  
}
```

```
Public void associer(int idp, int ida){  
    Personne p=personnes.get(idp);  
    p.adresses.add(adresses.get(ida));  
}
```

```
}
```