

Objectifs

Ecrire et manipuler des JSP avec Beans.

1 Préambule

1. Créer un répertoire `mvc` dans `WEB-INF/classes`
2. Créer dans `mvc` un objet `Message.java` qui servira de Bean

```
package mvc;
public class Message
{
    private int i;
    public Message() {i=0;}
    public String toString() {i++; return "C'est le " + i + "ème Hello World !";}
}
```

3. Ne pas oublier de compiler cet objet. Par ailleurs, après tout changement de ce code, le contexte doit être redémarré
4. Créer à la racine du contexte une page `Test.jsp` qui contient juste les deux lignes:

```
<jsp:useBean id="p" class="mvc.Message" scope="session" />
<%=p%>
```

5. Testez avec deux navigateurs simultanés
6. Modifiez `scope` en `request` et retestez
7. Modifiez `scope` en `application` et retestez

2 Exercice

1. Construire **sans aucun Bean** une application WEB de présentation de l'annuaire d'une société. Un utilisateur doit pouvoir rechercher une personne sur une partie de son nom et retrouver ses coordonnées. Cette application contient donc une page HTML contenant le formulaire de recherche et une JSP qui se connecte à la base de données et récupère les renseignements sur la (ou les) personnes recherchées. Si la personne n'existe pas un message adéquat sera affiché. Si il y a plusieurs personnes vérifiant le critère, seule la première sera affichée. Cette application nécessite bien sûr une base annuaire contenant les champs (`num`, `nom`, `prenom`, `sexe`, `tel`, `fonction`) par exemple.
2. Modifiez votre application pour que tout ce qui concerne la base de données soit maintenant placé dans un Bean.
 - Créer tout d'abord un objet POJO `Personne` avec autant d'attributs que de colonnes dans la table. Il constitue le "mapping" objet du contenu de la table. Donnez lui une méthode `getHTML()` qui renvoie une représentation HTML de cet Objet.
 - Construire maintenant un objet `DAOPersonne` avec notamment une méthode `find` qui se connecte à la base et permet de rechercher la personne sur une partie de son nom. Cette méthode renvoie une instance de `Personne` ou `null` si elle n'existe pas. Placez un `main` dans cet objet afin qu'il soit "testable" indépendamment du WEB.
 - Construire enfin la JSP `vuePersonne.jsp` qui utilise cet objet `DAOPersonne` en tant que Bean et affiche le résultat de la recherche quand la personne existe. Rappel : Les beans sont à placer dans le répertoire `WEB-INF/classes/package` du contexte d'application.
3. Ajoutez à votre objet une méthode `findAll(String)` pour permettre de retrouver plusieurs personnes répondant au critère. Le Bean donne alors accès à une `List` de `Personne`.
4. Modifiez votre application pour permettre une recherche sur différents critères conjugués (`Tel`, `fonction`, `nom` etc ...) et permettre le tri sur chacune des colonnes affichées.

Annuaire de la société

Name

Prenom

Sexe

N° Telephone

Fonction

Rechercher

Informations				
Nom	Prenom	Sexe	Tel	Fonction
mathieu	philippe	M	XXX	XXX
durand	jean	M	XXX	XXX
lefebvre	paul	M	XXX	XXX