

Accès aux données PHP

1 – Exploitation des données d'un formulaire

Création d'un formulaire statique

On souhaite à présent afficher uniquement les employés appartenant à un service. La première étape est de permettre à l'utilisateur de sélectionner le service. Pour cela, nous allons concevoir un formulaire contenant une **liste déroulante des services** (en statique dans un 1^{er} temps) :

Résultat à obtenir

Liste des employés par service

Service :

Voici la première version **statique** de ce formulaire (fichier **formEmplService.html**).

→ **Ajoutez** cette page **formEmplService.html** dans la couche "présentation" de votre projet :

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8" />
    <link rel="stylesheet" href="../design.css" />
    <title>Sélection du service</title>
  </head>
  <body>
    <h1>Liste des employés par service</h1>
    <form class="formulaire" action="emplService_post.php" method="post">
      <ul>
        <!-- Liste déroulante des services -->
        <li><label for="champService">Service :</label>
          <select name="service" id="champService">
            <option value="s04">Administration</option>
            <option value="s03">Commercial</option>
            <option value="s02">Emballage</option>
            <option value="s01">Fabrication</option>
          </select>
        </li>
        <li><input type="submit" value="Afficher les employés"></li>
      </ul>
    </form>
  </body>
</html>
```

N'oubliez pas le fichier CSS que nous compléterons ensuite

On associe le label au contrôle identifié par son id

Bien voir qu'on affiche les noms des services, et qu'on passera le code sélectionné dans `$_POST['service']`

Ajoutez dans la feuille de style (fichier **design.css**) les classes suivantes :

```
.formulaire ul {
    list-style-type: none;
    list-style-position: outside;
    margin: 0px;
    padding: 0px;
}

.formulaire li {
    padding: 8px;
    position: relative;
}

.formulaire label {
    width: 80px;
    display: inline-block;
    text-align: right;
    padding: 3px;
}
```

- Testez votre formulaire : la liste déroulante doit fonctionner.
- Le bouton de validation déclenche une erreur qui est pour l'instant normale : il nous reste à écrire le script **emplService_post.php** qui va afficher les employés du service sélectionné.

Exécution d'une requête SQL contenant des paramètres

Le script **emplService_post.php** va afficher les employés du service sélectionné dans la liste déroulante :

Sur la validation du formulaire, on va récupérer la variable **\$_POST['service']** qui contiendra **le code du service sélectionné** (valeur du "value" dans la liste déroulante).

La requête SQL à écrire correspond à : "sélectionner tous les employés dont le code service vaut une certaine valeur. Il s'agit donc d'une **requête contenant un paramètre** : le code d'un service.

- Dans la couche "**Persistence**", ajoutez une fonction **getEmployesParService(\$service)** dans le fichier **dialogueBD.php**
(On ajoute une méthode à notre classe DialogueBD.php).

```
public function getEmployesParService($service) {

    try {
        $conn = Connexion::getConnexion();
        $sql = "SELECT * FROM employe WHERE ServEmpI=?";
        $sql = $sql." ORDER BY NomEmpI";
        $sth = $conn->prepare($sql);
        $sth->execute(array($service));
        $tabEmployesService = $sth->fetchAll(PDO::FETCH_ASSOC);
        return $tabEmployesService;
    } catch (PDOException $e) {
        $erreur = $e->getMessage();
    }
}
```

Ici 1 seul paramètre
matérialisé par le ?

Explications :

```
$sql = "SELECT * FROM employe WHERE ServEmpl=?";
```

```
$sql = $sql." ORDER BY NomEmpl";
```

⇒ Ecriture de la requête sur 2 lignes : le ? correspond au paramètre attendu

```
$sth = $conn->prepare($sql);
```

⇒ Préparation de la requête

```
$sth->execute(array($service));
```

⇒ Exécution de la requête préparée sur l'objet \$sth, en lui passant le(s) paramètre(s) nécessaire(s) sous la forme d'un tableau (array) (ici un seul élément : le code du service qui est passé en paramètre à la fonction)

A RETENIR :

Le tableau des paramètres doit contenir autant d'éléments qu'il y a de ? dans la requête préparée (*dans notre cas, 1 seul*).


Si on a plusieurs ? **l'ordre doit également être respecté.**

```
$tabEmployesService = $sth->fetchAll(PDO::FETCH_ASSOC);
```

⇒ Récupération des lignes (tableau résultat de la requête)

→ Dans la couche "**Présentation**", réalisez le script **emplService_post.php** qui va afficher les employés du service sélectionné.

- Dans la partie "Données" du script, il faut récupérer le code du service sélectionné :
\$codeService = \$_POST['service'];
- Et vous appelez la fonction avec ce paramètre : **getEmployesParService(\$codeService)**
- Une fois les données des employés récupérées, il reste à les afficher dans un tableau : Reprenez votre code précédent **tableauEmployes.php** pour réaliser l'affichage du tableau des employés du service sélectionné dans formulaire.

Voici le résultat à obtenir : dans l'exemple ci-dessous, l'utilisateur a saisi dans le formulaire le service "Commercial" (dans la liste déroulante)  code transmis = s03

Liste des employés du service s03


E008	FAUVEL	Anne
E009	NOUVION	Patrick

→ **Améliorez votre affichage** : on aimerait avoir le nom du service plutôt que son code dans le titre.

- Faites une nouvelle fonction dans le fichier dialogueBD.php, avec une requête qui va récupérer le nom d'un service avec son code **passé en paramètre**, puis modifiez le code de l'affichage pour obtenir le résultat ci-dessous :

Liste des employés du service Commercial

E008	FAUVEL	Anne
E009	NOUVION	Patrick

Aide : on récupère un objet **\$nomService** dont les propriétés sont constituées par les colonnes renvoyées par la requête. 
On récupère directement la valeur de la colonne "DesiServ" avec : **\$nomService->DesiServ**;