

#### Objectifs :

- Mutualiser les composants graphiques d'un formulaire (ajout et modification)
- Passer une valeur à un script en HTTP GET
- Gérer un script qui contient plusieurs traitements (récupération des données de l'employé, affichage des données sur le formulaire, modification des caractéristiques dans la table)
- Utiliser un champ caché dans un formulaire (input type=hidden)
- Modifier des données dans une table

## Sommaire

I.	cinématique des échanges client-serveur lors de la modification d'un employé .....	2
II.	Mutualisation des composants graphiques.....	4
	1) Création du nouveau script employe_composant_graph.php .....	4
	2) Modification du script employe_ajout.php .....	5
III.	Mutualisation des contrôles de saisie .....	5
IV.	Passage de l'id de l'employé à modifier.....	5
V.	Réalisation du script employe_modif.php .....	5
	1) Traitements 1 et 3 (récupération et affichage des données).....	5
	2) Traitement 2 (modification des données dans la table).....	8

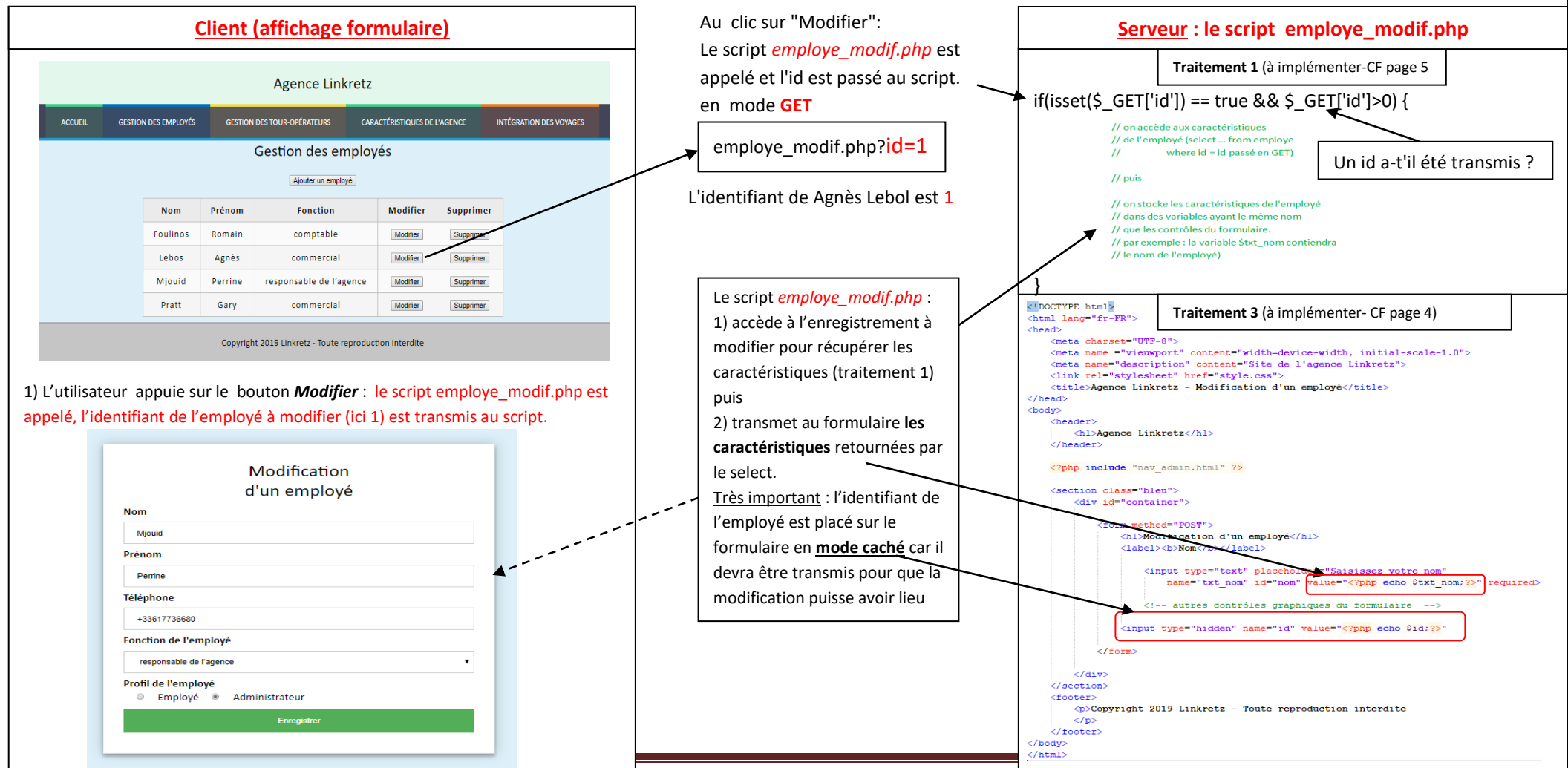
# I. CINEMATIQUE DES ECHANGES CLIENT-SERVEUR LORS DE LA MODIFICATION D'UN EMPLOYE

Sur la page "Gestion des employés" accessible à l'administrateur du site, l'ensemble des employés est listé et pour chaque employé un bouton Modifier permet de modifier les caractéristiques d'un employé. Lors du clic sur ce bouton, c'est le script **employe\_modif.php** qui est appelé. Pour l'instant, le code qui génère le bouton Modifier est le suivant :

```
echo "<td><a href='employe_modif.php'><input type='button' name='Modifier' value='Modifier'/> </a> </td>";
```

- Vous devrez modifier cette instruction pour transmettre au script l'identifiant de l'enregistrement à modifier de cette manière :

```
echo "<td><a href='employe_modif.php?id=$enreg->id' ><input type='button' name='Modifier' value='Modifier'/></a></td>";
```



1) L'utilisateur appuie sur le bouton **Modifier** : le script **employe\_modif.php** est appelé, l'identifiant de l'employé à modifier (ici 1) est transmis au script.

## Modification d'un employé

Nom  
Mjoud

Prénom  
Perrine

Téléphone  
+33617736680

Fonction de l'employé  
responsable de l'agence

Profil de l'employé  
☐ Employé ☒ Administrateur

Enregistrer

Après avoir modifié les caractéristiques de l'employé, l'utilisateur appuie sur le bouton "Enregistrer" pour valider ses modifications.

Le script **employe\_modif.php** est appelé  
Les données du formulaire sont passées en **POST**

Le script :

- 1) contrôle que les données transmises en POST sont valides
- 2) si les données sont valides : il met à jour l'employé dans la base de données et fait une redirection sur le script de consultation **avec un message** informant que la modification a été réalisée
- 3) si les données ne sont pas valides : le formulaire qui contient les données saisies est affiché ainsi que le **message d'erreur**

Si la **modification est valide**, on affiche un message indiquant que la modification a eu lieu et on affiche la liste des employés.

Si la **modification est invalide**, un message d'erreur est affiché et les valeurs saisies sont fournies sur le formulaire pour correction.

2) L'utilisateur modifie les informations du formulaire et appuie sur le bouton **Enregistrer** : le script **employe\_modif.php** est appelé. Les données du formulaire sont transmises en **POST**

Agence Linkretz

ACCUEIL GESTION DES EMPLOYÉS GESTION DES TOUR-OPÉRATEURS CARACTÉRISTIQUES DE L'AGENCE INTÉGRATION DES VOYAGES

Gestion des employés

Ajouter un employé

Nom	Prénom	Fonction	Modifier	Supprimer
Foulinos	Romain	comptable	Modifier	Supprimer
Lebos	Agnès	commercial	Modifier	Supprimer
Mjoud	Perrine Alice	responsable de l'agence	Modifier	Supprimer
Pratt	Gary	commercial	Modifier	Supprimer

La modification a été enregistrée

Copyright 2019 Linkretz - Toute reproduction interdite

Modification d'un employé

Nom  
Mjoud

Prénom

Téléphone  
+33617736680

Fonction de l'employé  
responsable de l'agence

Profil de l'employé  
☒ Employé ☐ Administrateur

Enregistrer

Le prénom est obligatoire

## Serveur : le script employe\_modif.php

Traitement 2

```
if (count($_POST) != 0) {
    // on contrôle que les données transmises sont
    // valides (même contrôles que pour l'ajout)

    // si les données sont valides, on exécute la
    // requête update avec les données du formulaire
    // et on fait un header sur le script de consultation
}
```

Traitement 3

```
<!DOCTYPE html>
<html lang="fr-FR">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta name="description" content="Site de l'agence Linkretz">
  <link rel="stylesheet" href="style.css">
  <title>Agence Linkretz - Modification d'un employé</title>
</head>
<body>
  <header>
    <h1>Agence Linkretz</h1>
  </header>

  <?php include "nav_admin.html" ?>

  <section class="bleu">
    <div id="container">

      <form method="POST">
        <h1>Modification d'un employé</h1>
        <label><b>Nom</b></label>

        <input type="text" placeholder="Saisissez votre nom"
          name="txt_nom" id="nom" value="<?php echo $txt_nom;?>" required>

        <!-- autres contrôles graphiques du formulaire -->

        <input type="hidden" name="id" value="<?php echo $id;?>"

      </form>

    </div>
  </section>
  <footer>
    <p>Copyright 2019 Linkretz - Toute reproduction interdite</p>
  </footer>
</body>
</html>
```

## II. MUTUALISATION DES COMPOSANTS GRAPHIQUES

Comparons les composants graphiques du formulaire d'ajout et ceux du formulaire de modification souhaité :

Ajout d'un employé	Modification d'un employé
<p><b>Nom</b></p> <input type="text" value="Saisissez votre nom"/>	<p><b>Nom</b></p> <input type="text" value="Mjoud"/>
<p><b>Prénom</b></p> <input type="text" value="Saisissez votre mot prénom"/>	<p><b>Prénom</b></p> <input type="text" value="Perrine"/>
<p><b>Téléphone</b></p> <input type="text" value="Saisissez votre téléphone"/>	<p><b>Téléphone</b></p> <input type="text" value="+33617736680"/>
<p><b>Fonction de l'employé</b></p> <div>commercial ▼</div>	<p><b>Fonction de l'employé</b></p> <div>responsable de l'agence ▼</div>
<p><b>Profil de l'employé</b></p> <p><input type="radio"/> Employé <input type="radio"/> Administrateur</p>	<p><b>Profil de l'employé</b></p> <p><input type="radio"/> Employé <input checked="" type="radio"/> Administrateur</p>
<p>Enregistrer</p>	<p>Enregistrer</p>

Ces deux formulaires contiennent les mêmes composants graphiques... c'est tout-à-fait normal car les caractéristiques **ajoutées** sont les mêmes que les caractéristiques **modifiées**.

- Pour optimiser le travail à faire, vous allez **créer un nouveau script *employe\_composant\_graph.php*** qui ne contiendra que les composants graphiques communs aux 2 fonctionnalités puis vous inclurez ce fichier avec un ***include*** dans les scripts ***employe\_ajout.php*** et ***employe\_modif.php*** au lieu d'y coder tous les composants graphiques.

Voici comment faire... mais vous pouvez choisir de réaliser le travail sans être guidé et sauter l'étape 1) !

### 1) Création du nouveau script *employe\_composant\_graph.php*

- Créer un nouveau script ***employe\_composant\_graph.php***
- Placez-y les composants graphiques contenus dans le script ***employe\_ajout.php***  
Voici ce que vous devez avoir dans ce nouveau script ***employe\_composant\_graph.php*** :

```
<label>Nom</label>
<input type="text" placeholder="Saisissez votre nom" name="txt_nom" id="nom" value="<?php echo $txt_nom; ?>" required>

| suite des composants

<input type="submit" id='valider' name="btn_valider" value='Enregistrer'>
<?php
echo $msg;
?>
```

## 2) Modification du script `employe_ajout.php`

- Supprimez tous les composants contenus dans le formulaire du script `employe_ajout.php` et ajoutez l'instruction `include` pour ajouter le script `employe_composant_graph.php` de cette manière :

```
<form method="POST">
  <h2>Ajout d'un employé</h2>
  <?php include 'employe_composant_graph.php'; ?>
</form>
```

## III. MUTUALISATION DES CONTRÔLES DE SAISIE

Les contrôles de saisie sont identiques pour la fonctionnalité d'ajout et celle de modification. Je pense que vous avez une petite idée de ce que je vais vous demander...

- Créez un nouveau script `employe_controle_saisie` dans lequel vous allez placer l'appel à la fonction `extract` et tous les contrôles à réaliser
- Utilisez ce nouveau script (`include`) dans le script `employe_ajout.php`
- Vous ferez de même dans le script `employe_modif.php` que vous allez bientôt réaliser.\*

## IV. PASSAGE DE L'ID DE L'EMPLOYÉ À MODIFIER

Si vous êtes passé à côté de cette modification indiquée page 2, vous devez modifier le script `employe_consult_admin.php` qui permet d'obtenir la liste des employés afin :

- de passer l'id de l'employé à modifier au script `employe_modif.php` lorsque l'on clique sur le bouton "Modifier" pour demander la modification d'un employé (attention : n'oubliez pas de modifier la requête SQL afin de récupérer l'id...)

**Voici comment faire pour passer l'id au script :**

```
<a href='employe_modif.php?id=$enreg->id' >
  <input type='button' name='Modifier' value='Modifier' />
```

id est passé en méthode GET et pas en méthode POST

## V. RÉALISATION DU SCRIPT `EMPLOYE_MODIF.PHP`

Voici comment réaliser les traitements 1, 2 et 3 décrits dans le paragraphe I) pages 2 et 3

### 1) Traitements 1 et 3 (récupération et affichage des données)

Vous allez réaliser les instructions qui vont permettre :

- de récupérer les caractéristiques de l'employé à modifier qui sont enregistrées dans la table `employe`
- d'alimenter le formulaire qui contient les caractéristiques retournées par le `select` et l'identifiant de l'employé passé en caché

Voici comment faire :

➤ Le traitement 1 (récupération des caractéristiques de l'employé à modifier)

```
// -----  
// le script est-il appelé lors du clic sur le bouton Modifier situé sur la page  
// d'affichage des employés (script employe_consult_admin.php)  
// -----  
  
if(isset($_GET['id']) == true && $_GET['id']>0){  
    // -----  
    // on récupère l'identifiant de l'employé passé avec la méthode GET dans la variable $id  
    // -----  
      
    // -----  
    // connexion à la base de données  
    // -----  
      
    try {  
        // -----  
        // on récupère les caractéristiques de l'employé  
        // dont l'identifiant est contenu dans la variable $id  
        // -----  
  
        // -----  
        // on stocke les caractéristiques de l'employé  
        // dans des variables ayant le même nom  
        // que les contrôles du formulaire.  
        // par exemple : la variable $txt_nom contiendra le nom de l'employé)  
        // -----  
          
    } catch(PDOException $e) {  
          
    }  
}
```

➤ Le traitement 3 (la génération du formulaire)

```
<!DOCTYPE html>
<html lang="fr-FR">

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta name="description" content="Site de l'agence Linkretz">
  <link rel="stylesheet" href="/css/style.css">
  <title>Agence Linkretz - Modification d'un employé</title>
</head>

<body>
  <?php include $_SERVER['DOCUMENT_ROOT'] . '/include/header.html'; ?>
  <?php include $_SERVER['DOCUMENT_ROOT'] . '/include/menu_admin.html'; ?>

  <section class="comm">
    <div id="container">
      <!-- formulaire de modification d'un employé -->
      <form method="POST">
        <h2>Modification d'un employé</h2>
        <input type="hidden" name="id" value = "<?php echo $id; ?>">
      </form>
    </div>
  </section>

  <?php
    include $_SERVER['DOCUMENT_ROOT'] . '/include/footer.html'
  ?>
</body>

</html>
```

On ajoute un contrôle graphique qui ne sera pas visible (**pas affiché**) sur le formulaire. Il contient l'identifiant de l'employé à modifier. Cette information est indispensable pour pouvoir modifier l'employé

Inclure ici le script *employe\_composant\_graph.php* qui contient les composants graphiques du formulaire

## 2) Traitement 2 (modification des données dans la table)

Vous allez réaliser les instructions qui vont permettre :

- de contrôler que les données transmises en POST sont valides
- de mettre à jour les caractéristiques de l'employé dans la base de données et faire une redirection sur le script de consultation **avec un message** informant que la modification a été réalisée : ce traitement ne sera réalisé que si les données sont valides

```
// -----  
// si le tableau $_POST contient le bouton valider  
// alors cela signifie que le formulaire a été soumis,  
// le traitement de mise à jour peut avoir lieu  
// -----  
if(isset($_POST['btn_valider']) == true){
```

// contrôle des autres données du formulaire à placer ici

```
// -----  
// s'il n'y a pas d'erreur de saisie on va modifier l'enregistrement  
// -----  
if ($msg == "")  
{  
    // connexion à la base de données  
      
    try {  
        // -----  
        // on réalise la requête de mise à jour (update)  
        // -----  
          
        // on indique dans la variable $msg que tout s'est bien passé  
          
        // -----  
        // on se redirige vers l'affichage des employés en fournissant le message d'information  
        // -----  
        header('Location: employe_consult_admin.php?msg='.$msg);  
    } catch(PDOException $e) {  
          
    }  
}
```

La variable \$msg qui contient le message d'information indiquant que la modification a été réalisée est passée au script en GET.

**Adaptez le script *employe\_consult\_admin.php* pour que le message transmis soit affiché après le tableau contenant les employés.**