

Lors de la séance précédente, vous avez finalisé la fonctionnalité « consultation des horaires d’une agence ».
 Dans ce TP, vous devez réaliser la version **V6 consistant à compléter la fonctionnalité « Faire une location »**.

Fonctionnalité à intégrer dans la version V6
Faire une location dans l’agence sélectionnée.

1. Analyse de l’existant

Lorsque l’on clique sur le bouton « Louer un véhicule » de la première agence (La nantaise), le site est redirigé vers l’url ci-dessous :

<http://projetphpnetcarmvc/index.php?controleur=gestionLocation&action=faireUneLocation&agence=1>

1. Analyser l’url et répondre aux questions :
2. Quelle est la page de destination (page appelée dans l’url) :
3. Quel est le contrôleur désigné ?
4. Quelle est l’action concernée ?
5. Quelle agence est spécifiée ? Quelle information est fournie pour la désigner ?
6. Consulter les fichiers de l’application dans Netbeans et répondre aux questions :
 - 6.1. Est-ce que ce contrôleur existe déjà comme fichier dans le site fourni ?
 - 6.2. Est-ce que l’action indiquée dans l’url existe déjà dans le fichier du contrôleur ?
 - 6.3. Le site comporte-t-il une vue déjà codée qui serait utile pour la fonctionnalité de location ? Si oui :
 - quel est le rôle de cette vue ?
 - quelles variables PHP utilise cette vue ?

2. Créer une nouvelle branche du projet nommée « Finalisation Location »

7. Synchroniser les dépôts local et distant : **Git > Remote > Pull**
8. Créer une nouvelle branche du projet : **Clic-droit** sur le projet : **Git > Branch/Tag > Create Branch**

2.1. Compléter le contrôleur principal de manière à appeler le contrôleur adéquat

9. Ouvrir le fichier **index.php** et rajouter le cas ci-dessous :

```

29
30
31
32
33
34
35
36
37
38
39
40
41
??

case 'gestionAgence' :
{ include("contrôleurs/c_gestionAgence.php");
break;
}

case 'gestionLocation' :
{ include("contrôleurs/c_gestionLocation.php");
break;
}

//import des instructions permettant de construire le pied de page
include("vues/v_layout_pied.php") ;

```

10. Enregistrer

2.2. Créer et compléter le sous-contrôleur `c_gestionLocation.php`

11. Faire une copie du fichier `c_gestionAgence.php` et le renommer `c_gestionLocation.php`

12. Modifier le code de manière à obtenir ceci :

```
<?php
if(!isset($_GET['action']))
    $action = 'consulterLesLocations';
else
    $action = $_GET['action'];

switch($action)
{
    case 'consulterLesLocations':
    {
        $message = "Cette action n'est pas encore développée ;|";
        $stylePresentation = "text-warning";
        break;
    }

    case 'faireUneLocation':
    {
        $leCodeAgence = $_GET["agence"];

        include("vues/v_faireUneLocation.php");

        $message = "Netcar met à disposition une large gamme de véhicules "
            . "pour répondre au mieux à vos besoins.";
        $stylePresentation = "text-success";
        break;
    }
}
```

13. Analyser le code fourni :

13.1. Combien d'actions sont prévues dans le sous-contrôleur à ce stade ?

13.2. Combien d'actions sont réellement créées ?

13.3. Quelle est l'action qui est définie par défaut (si l'url n'a pas de valeur pour ce paramètre) ?

13.4. Que contient la variable `$leCodeAgence` après son affectation ? D'où vient la valeur ?

13.5. Quel est le lien entre cette variable et la vue `v_faireUneLocation.php` ?

14. Enregistrer et tester la fonctionnalité en cliquant sur « **Louer un véhicule** ».

14.1. Que se passe-t-il ?

14.2. Y-a-t-il des informations déjà saisies ? D'où proviennent-elles ?

14.3. Le résultat obtenu est-il conforme à l'écran ci-dessous :

14.4. A quelle vue correspond cet affichage ?

2.3. Analyse du formulaire de saisie de la location

15. Ouvrir le fichier **v_faireUneLocation.php** dans Netbeans et répondre aux questions ci-dessous :

- 15.1. Quel est le nom de chaque zone de texte à saisir ?
- 15.2. Quelle est la méthode choisie pour transmettre les informations saisies dans ce formulaire ?
- 15.3. Quel est le fichier spécifié comme destinataire de ces données ?
- 15.4. Quels paramètres sont fournis en complément au fichier destinataire ?
Par quelle méthode sont transmis ces paramètres complémentaires ?

2.4. Codage de l'action EnregistrerUneLocation

16. Ouvrir, dans Netbeans, le fichier **Contrôleurs/c_gestionLocation.php**

17. Rajouter le code de la nouvelle action.



```
<?php
include 'Modeles/LocationDAO.php';

if (!isset($_GET['action']))
{
    // ...
}
else
{
    // ...
}
switch($action)
{
    case 'consulterLesLocations':
        $message = "Cette action n'est pas encore développée.";
        $stylePresentation = "text-warning";
        break;

    case 'faireUneLocation': // ... 11 lines ...
    case 'enregistrerUneLocation':
        $codeAgence = $_POST["codeAgence"];
        $dateDebut = $_POST["dateDebut"];
        $dateFin = $_POST["dateFin"];
        $categorieVehicule = $_POST["categorieVehicule"];

        $objLocation = new LocationDAO();
        $resultat = $objLocation->enregistrerLaLocation($codeAgence, $dateDebut, $dateFin, $categorieVehicule);
        if ($resultat == 1) {
            $message = "La location a été enregistrée.<br/>Netcar vous remercie pour votre confiance.";
            $stylePresentation = "text-success";
        }
        else {
            $message = "La location n'a pu être enregistrée.";
            $stylePresentation = "text-danger";
        }
        break;
}
```

18. Analyser le code fourni :

- 18.1. Quel est le fichier inclus ? Existe-t-il ?
- 18.2. Dans le code de la nouvelle action :
 - Quelles variables PHP sont initialisées ? D'où proviennent les valeurs utilisées pour l'initialisation ?
 - Est-ce que toutes les données ont été récupérées ?
 - Quel objet est créé ? A quelle classe appartient-il ? Est-ce que cette classe existe à ce stade ?
 - Quelle fonction est appelée sur cet objet ? Est-ce que cette fonction existe à ce stade ?
 - Comment est utilisé le résultat retourné par cette fonction ? Quels sont les différents résultats possibles ?

2.5. Codage de la classe LocationDAO

19. Faire une copie de la classe **AgenceDAO.php** que vous renommerez **LocationDAO.php**

20. Modifier le code ainsi :

```
<?php
class LocationDAO extends ConnexionBD
{
    //cette classe hérite de la propriété $connexion
    // la propriété $connexion contient un objet PDO permettant de se connecter et d'interagir
    // avec une base de données

    // cette classe a accès à l'accessor getConnexion() qui retourne la propriété $connexion
    // grâce à cet objet $connexion de la classe PDO, la fonction query() sera accessible
    // la fonction query() permet de demander l'exécution d'une requête au SGBD

    public function enregistrerLaLocation($codeA, $debut, $fin, $categV)
    {
        $req = "insert into Location (CODECATEGORIE, CODEAGENCE, DATEDEBUT, DATEFIN)"
        . " values ('".$categV."', '".$codeA."', '".$debut."', '".$fin."')";

        $resultatDeLaRequete = $this->getConnexion()->exec($req);

        return $resultatDeLaRequete;
    }
}
```

21. Enregistrer.

2.6. Test de la fonctionnalité « Louer un véhicule »

22. Saisir les informations suivantes :

Faire une location de véhicule

Agence sélectionnée
1

Date de prise du véhicule
2025-05-21

Date de retour du véhicule
2025-05-20

Catégorie de véhicule souhaitée
A

Civilité
M

Nom devant figurer sur la location
M. Dupont

Email pour l'envoi de la confirmation
m.dupont@email.com

23. Cliquer ensuite sur le bouton « **Réserver** ». Vous devriez obtenir le résultat suivant :

La location a été enregistrée.
Netcar vous remercie pour votre confiance.

24. Aller vérifier dans la base de données si la ligne a été effectivement ajoutée.

2.7. Tests complémentaires à réaliser (démarche utile pour les futurs développements)

25. Tester en premier lieu la requête dans l'onglet SQL de **PhpMyAdmin** :



26. Ensuite, lorsque la requête est mise au point (fourni le résultat attendu), tester la fonction dans un fichier test.

2.8. Enregistrer la version V6 sur la branche et fusionner avec la branche principale

27. Enregistrer les changements dans Git (**Git add**)
28. Enregistrer une version de cette branche (**Git commit** « V6.... »)
29. Fusionner ensuite cette nouvelle branche avec la branche principale (**Git Merge**).

2.9. Transférer cette version V6 sur le dépôt distant (Gitlab)