

Morgado-Samagaio Jonathan

2A

**PWS / TP4 PHP**

## Table des matières

Table des matières .....	2
1. Introduction.....	3
2. Amélioration des pages précédentes.....	3
3. Espace sécurisé : Session et Cookie.....	3
4. Conclusion .....	5

## 1. Introduction

Dans ce TP, nous allons voir comment utiliser les sessions et les cookies en PHP pour faire des espaces sécurisés.

## 2. Amélioration des pages précédentes

Nous commençons par améliorer les pages ConsultPrix.php et ConsultCat.php pour que, quand on affiche le tableau, la catégorie ou le prix sélectionné ne reprenne pas sa valeur par défaut. Le code pour ConsultPrix.php :

```
<?php
    echo '-500€ <input type="radio" name="prix" value="-500"';
    if (isset($_POST['Valider']) && $_POST['prix'] == "-500")
    {
        echo 'checked';
    }
    echo '><br><br>';
    echo '+500 <input type="radio" name="prix" value="+500"';
    if (isset($_POST['Valider']) && $_POST['prix'] == "+500")
    {
        echo 'checked';
    }
    echo '><br><br>';
?>
```

On vient vérifier si le formulaire a déjà été validé et la case qui a été choisie pour rajouter la valeur checked à l'un des deux boutons radio.

Pour ConsultCat.php :

```
<?php
include("InitTableaux.php");
for ($i = 0; $i < sizeof($tabCat); $i++)
{
    echo '<option value="'. $tabCat[$i]['idCategorie'] . '";';
    if (isset($_POST['afficher']) && $_POST['cat'] == $tabCat[$i]['idCategorie'])
    {
        echo 'selected';
    }
    echo '>'. $tabCat[$i]['nomCategorie'] . '</option>';
}
?>
```

On vient donc rajouter selected si la case a été sélectionnée précédemment.

## 3. Espace sécurisé : Session et Cookie

Nous allons faire un espace sécurisé avec nos différentes pages. La connexion se fera par la page FormConnexion.php qui enverra les données à TraitConnexion.php pour vérifier les identifiants. Si les

identifiants sont corrects, une session est alors créée et il devient possible d'accéder aux autres pages. S'il n'y a pas de session et que l'on essaie d'accéder à l'une des autres pages, nous sommes renvoyés sur la page de connexion.

```
<?php
    if (isset($_GET['msgError']))
    {
        echo '<h1>'.$_GET['msgError'].'</h1>';
    }
?>
<h1>Veuillez entrer les identifiants pour accéder aux données</h1>
<form method="POST" action="traitConnexion.php">
    <?php
        echo 'Login : <input type="text" name="login" ';
        if (isset($_COOKIE['CmorgadoJonathan']))
        {
            echo 'value="'.$_COOKIE['CmorgadoJonathan'].'"';
        }
        echo ' /> <br><br>';
    ?>
    Mot de passe : <input type="password" name="password"><br><br>
    <input type="checkbox" name="cb"> Se souvenir de moi <br><br>
    <input type="submit" name="afficher" value="Valider">
</form>
```

Nous vérifions en premier s'il y a un message d'erreur dans l'URL pour l'afficher. Nous regardons ensuite si un cookie existe pour afficher sa valeur dans le champ login.

TraitConnexion.php :

```
<?php
session_start();
if (isset($_POST['afficher']))
{
    if (htmlentities($_POST['login']) != "Achille" || htmlentities($_POST['password']) != "Talon")
    {
        header('Location: FormConnexion.php?msgError=Erreur de connexion, mauvais identifiants');
    }
    else
    {
        $_SESSION['SmorgadoJonathan'] = $_POST['login'];
        if (isset($_POST['cb']))
        {
            setcookie('CmorgadoJonathan', $_POST['login'], time()+5*60);
        }
        header("Location: index.php");
    }
}
?>
```

On active la gestion des sessions au tout début. On regarde ensuite si les identifiants saisis sont corrects. S'ils le sont, on crée la session et on vérifie si la checkbox a été cochée pour pouvoir créer le cookie et rediriger vers index.php. S'ils ne le sont pas, on redirige vers la page de connexion avec un message d'erreur.

Sur chacune des pages, nous venons rajouter ce bloc php au tout début pour vérifier que l'utilisateur est bien connecté :

```
<?php
session_start();
if (isset($_SESSION['SmorgadoJonathan']))
{
    if ($_SESSION['SmorgadoJonathan'] != 'Achille')
    {
        header('Location: FormConnexion.php?msgError=Acces Interdit');
    }
}
else
{
    header('Location: FormConnexion.php?msgError=Acces Interdit');
}
?>
```

On affiche dans index.php les valeurs de \$\_SESSION et \$\_COOKIE :

```
<?php
echo 'On affiche la session :<br>';
var_dump($_SESSION);
echo '<br><br>';
echo 'On affiche le possible cookie :<br>';
var_dump($_COOKIE);
?>
```

On ajoute dans le menu un bouton « Se déconnecter » qui permet de détruire la session et de rediriger vers FormConnexion.php :

```
<?php
session_start();
session_destroy();
header('Location: FormConnexion.php');
?>
```

On ajoute aussi un bouton « Détruit le cookie » qui permet de supprimer le cookie contenant le nom d'utilisateur :

```
<?php
setcookie('SmorgadoJonathan', "", time()-3600);
header('Location: '.$_GET['lien']);
?>
```

## 4. Conclusion

Dans ce TP, nous avons vu comment utiliser les sessions pour créer des espaces sécurisés et les cookies pour stocker des informations comme des identifiants.