

Variable de session PHP et APPLICATION

Tache 1 :

Une variable de session PHP est une variable stockée

C'est une variable temporaire qui a une durée limitée et est détruite à la déconnexion (fermeture du navigateur).

Les variables de session sont partagées par toutes les pages PHP d'une session (accès depuis un même navigateur). Elles permettent donc le passage d'information entre pages.

Exemple d'utilisation

- Sauvegarde d'identifiants de connexion
- Sauvegarde d'un panier d'achat

C'est ce morceau de code qui permet l'utilisation de celle-ci. (! à toujours placé en haut des pages)

```
<?php session_start(); ?>
```

Et on peut faire appel aux variables de session de cette façon :

```
$_SESSION['variable'] = valeur;
```

Tache 2 :

Nous allons tout d'abord créer une page de connexion qui permettra à l'Admin de se connecter.



CONNEXION

Nom d'Utilisateur :

Mot de Passe :

SE CONNECTER

Les commentaires sont fournis sur les Screens

```
<?php
session_start();

//Récupération du pseudo et MDP
$pseudo = $_POST['pseudo'];
$pass = $_POST['pass'];

// connexion BD
$bdd = new PDO('mysql:host=localhost;dbname=client', 'root', '');
//Requête
$sql = "SELECT * FROM user WHERE pseudo = '$pseudo'";
$result = $bdd->prepare($sql);
$result->execute();

var_dump($result);
// Si le résultat est positif alors on vérifie si mdp correspond à celui de la BD
if($result->rowCount() > 0)
{
    var_dump($pass);
    $data = $result->fetchAll();

    if(password_verify($pass, $data[0]['password']))
    {
        // On affiche un message comme quoi il est bien connecté |
        var_dump($pass);
        echo "Connexion effectué";
        $_SESSION['pseudo'] = $pseudo;
    }
}
```

La fonction « password_verify » permet de vérifier le MDP saisi et celui dans la BD

Les var_dump sont des message système pour savoir ce qu'il se passe (suppression possible)

Si la connexion est réussie alors on initialise la variable de SESSION au pseudo saisi

Avec cette ligne « **\$_SESSION['pseudo'] = \$pseudo** »

Ajout d'un Admin :

Si le pseudo et MDP ne correspondent pas à la BD alors on crée un utilisateur admin avec les informations saisi

```
else // Sinon on créé un Admin avec le pseudo et le MDP entrée
{
    $pass = password_hash($pass, PASSWORD_DEFAULT);
    $sql = "INSERT INTO user (pseudo, password) VALUES ('$pseudo','$pass')";
    $req = $bdd->prepare($sql);
    $req->execute();
    echo "Enregistrement Effectué ";
}
```

J'entre en NOM D'UTILISATEUR : Admin et en MDP : test

Enregistrement Effectué

La BD étant vide il crée donc un utilisateur Admin, voici donc la BD

+ Options			
← T →	▼	id	pseudo password
<input type="checkbox"/>	✎ Éditer	✂ Copier	🗑 Supprimer
1	admin	\$2y\$10\$dild4JRzFaeX7YSRsrHxc.BDRHHKlwnM6yjjbSD8X1q...	

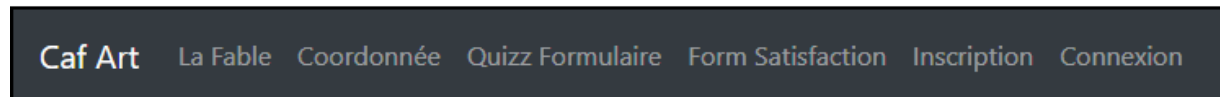
Le mot de passe est crypté (hashé) c'est grâce à la fonction « password_hash » qui prend le MDP saisi et le crypte pour une sécurité accrue

Maintenant que l'utilisateur est créé je peux me connecter à celui-ci en saisissant les mêmes identifiants

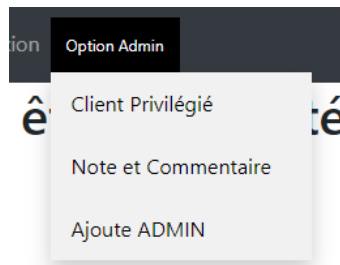
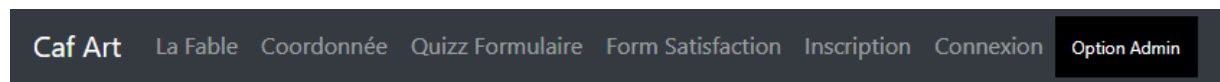
Connexion effectué

Je suis connecté en Admin j'ai donc accès à des fonctionnalités que les autres utilisateurs n'ont pas

Personnes classiques :



Admins :



Voici mon menu avec une nouvelle case que personne ne voit sauf les Admins qui permet l'accès aux clients, aux notes etcetc...

C'est bien évidemment 2 Menu différents qui s'affiche en fonction de qui est connecté sur le site (Admin ou une personne Lambda)

```

<?php
if(isset($_SESSION['pseudo']))
{
    ?>
    <div name="menu">
        <?php include "menuAdmin.php";?>
    </div>
    <?php
}
else
{
    ?>
    <div name="menu">
        <?php include "menu.php";?>
    </div>
    <?php
}
?>

```

On check si une variable de **SESSION** est en cours d'utilisation si oui alors on affiche le Menu Admin, on peut bien sûr vérifier si la personne connectée est un Admin et pas juste un membre qui possède un compte (à améliorer)

Suppression :

Il peut ainsi aller sur la page client et effectués des manipulations comme supprimé un client

Liste des Inscrits : Résultat avec PDO	
•	Caron Sylvain 0102030405 abcd@gmail.com Supprimer

BD :

+ Options						
	id	nom	prenom	tel	mail	
<input type="checkbox"/>  Éditer  Copier  Supprimer	1	Caron	Sylvain	0102030405	abcd@gmail.com	

Code effectué lors du clic sur « supprimer » (Commentaire sur screen)

supprimer.php

```

<?php

//Connexion
$objectPDO = new PDO('mysql:host=localhost;dbname=client','root','');

//Requete de suppression
$pdoStat = $objectPDO->prepare('DELETE FROM contact WHERE id=:num LIMIT 1');

//Liaison du parametre

$pdoStat->bindValue(':num', $_GET['numContact'], PDO::PARAM_INT);

//Execution
$executeIsOk = $pdoStat->execute();

if($executeIsOk)
{
    $message = 'Supprimer !';
}
else
{
    $message = 'Echec';
}

?>

```

inscrit.php

```

<?php foreach($contacts as $contact): ?>

    <li>
        <?= $contact['nom'] ?> <?= $contact['prenom'] ?> <?= $contact['tel'] ?> <?= $contact['mail'] ?>
        //Lors de l'affichage on fourni l'id en numContact afin de pouvoir supprimer une personne juste en trouvant l'id voulu
        <a href='supprimer.php?numContact=<?= $contact['id'] ?>'>Supprimer</a>
    </li>

<?php endforeach; ?>

```

Affichage du message :

Suppression

Supprimer !

Plus de client dans la liste :

Liste des Inscrits : Résultat avec PDO

BD vide :

id	nom	prenom	tel	mail	