

TP n° 3 DAW : Site dynamique avec PHP (Mini Boutique virtuelle)

Nous souhaitons créer un site Web dynamique qui permet de présenter en ligne le catalogue de produits.

- Le catalogue doit présenter des produits.
- Un produit est défini par :
 - Sa référence
 - Sa désignation
 - Sa quantité en stock
 - Son prix
 - Sa photo
 - Un champ qui indique s'il est disponible
 - Un champ qui indique s'il est en promotion
 - Un champ qui indique s'il est sélectionné ou non
- Un client peut consulter le catalogue du site et obtenir ainsi toutes les références des produits vendus.
- Un client, après authentification, peut effectuer ses propres commandes. Il doit alors créer un panier virtuel et y ajouter les produits qu'il souhaite acheter. Lorsqu'il le voudra, il pourra valider sa commande.
- Le paiement et la livraison de la commande n'entrent pas dans le cadre de notre étude.
- Un administrateur peut gérer le catalogue du site (ajouter de nouvelles références ou supprimer des références existantes) ainsi que le stock (acheter de nouveaux produits et faire l'inventaire).

Annexe (Base de données en PHP) :

1. Insérer des données à partir d'un formulaire

Dans un site dynamique, il est intéressant de prévoir la possibilité d'insérer des données en ligne. Par exemple dans votre TP, l'un de vos formulaires va servir à insérer les données concernant les anciens clients.

2. – Création de la base de données et des tables

Avant d'insérer les données, il faut avant tout créer une base de données puis une table (ou des tables) qui va accueillir les données. Par exemple pour créer une table CLIENT, on peut exécuter les instructions SQL suivantes :

```
drop table if exists CLIENT;

create table CLIENT (
    id    int not null auto_increment,
    login varchar(15) not null,
    password varchar(55) not null, primary
    key (id)
) ;
```

Ce formulaire va permettre d'envoyer les données figurant dans des les champs <input ... avec la méthode « GET », c'est à dire que les données vont transiter par l'URL vers un fichier **insert.php**

formulaire.html

```
Pour vous inscrire :<br>
<form method="get" action="insert.php">
  LOGIN : <input type="text" name="login"><br>
  PASSWORD : <input type="text" name="pass"><br>
```

3. Insertion des données

insert.php

```
<?php
$login = $_GET['login'];
$mdp = $_GET['pass'];

$connect = mysql_connect('localhost','root','') or die ("erreur de
connexion");//connexion au serveur

mysql_select_db('Boutique') or die ("erreur de connexion
base");//sélection de la base de données

$req = " INSERT INTO CLIENT (login, password) VALUES
('$login','$mdp');"; echo $req; // tester la requete sur
phpmyadmin

mysql_query($req);

echo "Client ajouté"; // vérifier avec phpmyadmin
?>
```

4. Affichage du résultat d'une requête SQL

listeClient.php

```

<?php
$connect = mysql_connect('localhost','root','') or die ("erreur de connexion");

mysql_select_db('CLIENT') or die ("erreur de connexion base");//selection de la base de données

$req = "SELECT * FROM CLIENT ";
$result = mysql_query($req);

while ( $ligne=mysql_fetch_assoc($result)){
    $id = $ligne['id'];
    $login = $ligne['login'];
    $passwd = $ligne['password'];
    echo $id.' : ' . $login.' : ' . $passwd.'  


```

5. Mysql : INSERT – UPDATE – DELETE

INSERT

Pour insérer des données dans cette table, il ne faut pas oublier que le champ ID est de type **auto_increment**. On n'a donc pas besoin d'insérer une valeur dans ce champ. Voici la requête qui va permettre d'insérer un nouvel utilisateur, portant le pseudo "toto", le mot de passe "toto" .

```
mysql_query("INSERT INTO CLIENT VALUES('','toto','toto')");
```

UPDATE

Pour modifier des données qui existent déjà dans une table, on utilise une requête de type UPDATE. Nous souhaitons modifier dans la table UTILISATEUR le mot de passe de l'utilisateur "titan" et le login pour le remplacer par **tris20** et **azerty**. Nous allons utiliser la requête suivante :

```
mysql_query("UPDATE CLIENT SET login='tris20',password='azerty' WHERE id=1");
```

DELETE

Si nous souhaitons supprimer l'utilisateur **titan**. Nous allons utiliser la requête suivante :

```
mysql_query("DELETE FROM CLIENT WHERE id=1");
```

Remarque :

Pour mettre à jour ou supprimer un enregistrement il est préférable de sélectionner l'identifiant, car des CLIENT peuvent avoir le même login.

6. Démarche à suivre :

Il est fréquent dans une application WEB que l'administrateur doit réaliser ce type d'opération sur une table. Dans ce cas vous devez proposer une interface à l'administrateur, qui généralement propose les fonctionnalités suivantes :

1 – Afficher les clients (utilisateurs) et les différentes opérations : fichier → afficheUtil.php

id	login	password	mise a jour	delete
2	toto	toto	maj	delete
3	tutu	123456	maj	delete

[ajout utilisateur](#)

ajout.php

maj.php?id=2

delete.php?id=2

Pour supprimer ou mettre à jour un utilisateur vous devez indiquer dans l'URL l'**identifiant** de l'utilisateur.

2 – Créer un formulaire pour ajouter un utilisateur : fichier → ajout.php

Login :

Mot de passe :

3 – Créer un script PHP (delete.php) pour récupérer l'id de l'utilisateur à supprimer et effectuer la requête de suppression .

4 – La mise à jour est plus délicate, vous devez dans un premier temps récupérer l'id pour effectuer une requête de sélection du login et du password sur la table. On utilise ensuite le résultat de la requête pour remplir les champs du formulaire, que l'on modifiera par la suite.

maj.php

id :

Login :

Mot de passe :