# TP PHP base de donnée, Partie 3

# gestion de la base nounou à l'aide de scripts PHP

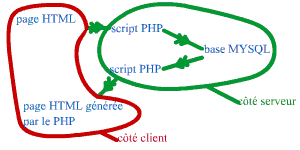
# Introduction

Nous allons nous mettre à la place de la personne qui va accéder à la base nounou, on ne peut pas la laisser taper des requêtes dans phpmyadmin, le plus simple est de lui écrire une série de petits scripts qui vont lancer les "bonnes" requêtes.

Nous allons donc devoir créer deux pages HTML, l'une à destination de la personne chargée d'alimenter la base de donnée, l'autre à destination de parents chargés de la consulter.

Chacune de ces pages va interroger la base mysql en lançant des scripts PHP.

Les scripts PHP sont des petits fichiers de commande qui s'exécute sur le serveur, le script PHP peut renvoyer au travers de la commande echo "uhgu..." du code HTML.



## Quelques rappels sur le langage PHP

Supposons que l'on veuille que, dans un fichier HTML, des instructions soient interprétées ou écrites par l'interpréteur PHP

### Il va falloir faire plusieurs choses,

* renommer “monfichier.html” en “monfichier.php”, pour que l'interpréteur php soit appelé.
* insérer le code PHP que l'on veut faire exécuter entre des balises <?php et ?>
* respecter la syntaxe du PHP (en gros une variable se nomme $quelqchose, un point virgule termine les lignes, début de bloc { et fin de bloc }, le signe point est la marque de concaténation et pour créer du code HTML, on dispose de l'instruction echo" le code HTML qu'on veut intégrer" .

### Un exemple, le fichier HTML suivant:

<!DOCTYPE html>

<html>

<head>

<title>test</title>

<meta http-equiv="Content-Type" content="text/HTML; charset=utf-8" />

</head>

<body>

<p>Ceci est du code html classique</p>

<?php echo "<p> ceci est du code html généré par le php</p>

<p>le calcul 7+9 a pour solution: ".7+9."</p>";

?>

</body>

</html>

est affiché ainsi:

| Ceci est du code html classique  ceci est du code html généré par le php  le calcul 7+9 a pour solution: 16 |
| --- |

## Activité 1: Réaliser l'interface HTML et un premier échange HTML <-> PHP

On commence par créer un petit fichier html...

<!DOCTYPE html>

<html>

<head>

<title>test</title>

<meta http-equiv="Content-Type" content="text/HTML; charset=utf-8" />

</head>

<body>

...

</body>

</html>

Pour passer une variable de script en script deux solutions sont possibles:

* dans un formulaire <input type="hidden" name="mavariable" value="toto">
* dans un lien < a href="monscript.php? mavariable=toto"> mon lien </a>

C'est ce que nous allons réaliser.

La première solution: cela consiste à lancer le script comme on lancerait un fichier HTML

<a href="monscript.php?nom=toto"> ce lien lance un script php</a>

l'autre solution que l'on utilise quand on veut faire traiter des données au script consiste à créer un formulaire:

<form method="POST" action="monscript.php">

<p>Quel le nom du client à ajouter</p>

<p><input type="text" name="nom" size="20"></p>

<p><input type="submit" value="Envoyer" ></p>

</form>

| Ce qu'il faut faire: Copier ces lignes dans le fichier HTML. Le contenu du formulaire est alors envoyé à "monscript.php" qui récupère sous forme de variable les champs nommés (ici selon le type de formulaire le contenu du champ de type texte sera exporté vers le script et récupérable dans le tableau  $\_POST["nom"] ou $\_GET["nom"]). |
| --- |

Créons ce fichier monscript.php

<!DOCTYPE html>

<html>

<head>

<title>test</title>

<meta http-equiv="Content-Type" content="text/HTML; charset=utf-8" />

</head>

<body>

<p>Test du nom du client! </p>

<?php

$nom=$\_POST["nom"];

 echo "Le nom est: ".$nom; ?>

<p><a href="index.html">Retour</a>

</body>

</html>

Il est bien évident que ces fichiers doivent être copiés dans un répertoire de votre serveur et qu'ils doivent être appelés avec la ligne de commande <http://localhost/monrepertoire/> .....

#### Ce qu'il faut remarquer

La variable nom de index.html devient dans monscript.php: $\_POST["nom"] (si on utilise la méthode POST) ou $\_GET["nom"] (si on utilise la méthode GET)

pour les curieux c'est une des conséquences du positionnement dans le fichier php.ini, de la commande registerglobal à off

Si on passe en paramètre un mot avec des apostrophes on obtient pas le même résultat avec les deux méthodes, voir l'option de php.ini: magic gpc

## Quelques remarques à lire sur les différents types de champs intervenant dans un formulaire:

-  type texte:

<p><input type="text" name="nom" size="20"></p>

- type radio: le contenu du champ de type radio sera exporté vers le script dans la variable $genre, cette variable pourra prendre les valeurs "informatique", "sciences" ou "maths"  
 <input type="radio" value="informatique" name="genre">informatique<br>

<input type="radio" value="sciences" name="genre">sciences<br>

<input type="radio" value="maths" checked name="genre">math

- type liste déroulante: le contenu du champ de type liste déroulante sera exporté vers le script dans la variable $genre, cette variable pourra prendre les valeurs "informatique", "sciences" ou "maths"  
informatique maths sciences

<select size="3" name="genre">

<option selected>informatique</option>

<option>maths</option>

<option>sciences</option>

</select>

- type case à cocher: le formulaire envoie les variables cochées avec la valeur true, ici seules les variables rugby et tennis sont envoyées.

<input type="checkbox" name="foot" value=true>foot<br>

<input type="checkbox" name="rugby" checked value=true>rugby<br>

<input type="checkbox" name="tennis" checked value=true>tennis

## Activité 2

| Ce qu'il faut faire Nous allons utiliser les fonctions : mysqli\_connect(), mysqli\_query(), mysqli\_num\_rows(), mysqli\_close()  Ces fonctions sont le minimum que vous utiliserez à chaque fois que vous interrogerez une base de données MySQL, Insérer le code ci-dessous (nous avons repris notre table nounou) dans un script php. |
| --- |

#### 

se connecter à sa base de donnée distante cela signifie bien comprendre comment fonctionne son environnement, dans le script qui suit je travaille:

#### en local sur mon propre serveur:

serveur: “localhost”  
login: “root”  
password: *vide*  
base de donnée: “nounou”

la ligne de connection: $con=mysqli\_connect("localhost","root","","nounou");

#### en distant sur free:

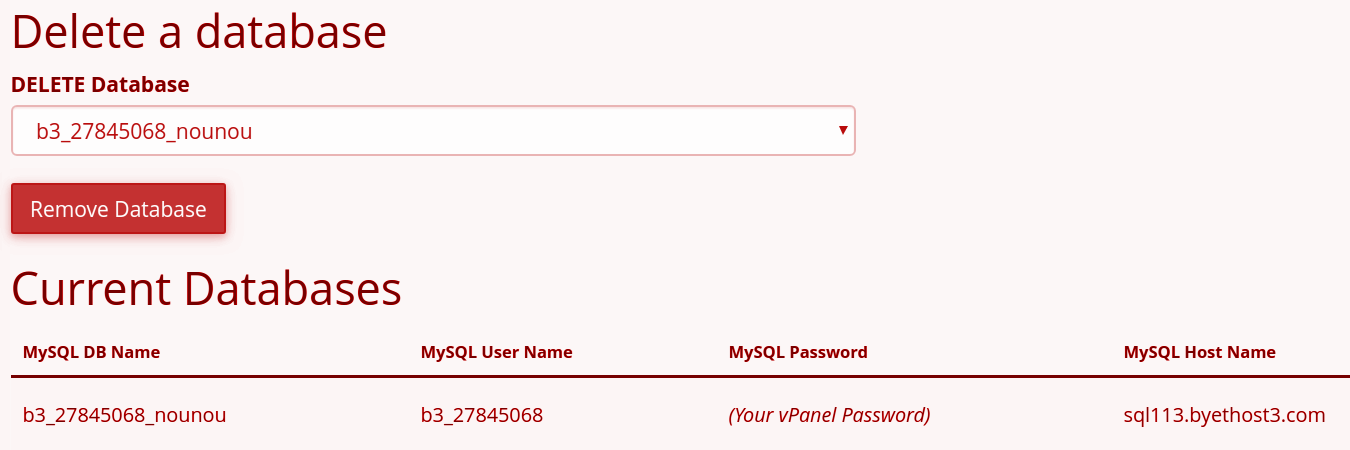
serveur: “sql.free.fr”  
login: *monloginfree*  
password: *monmotdepassefree*  
base de donnée: *monloginfree*

la ligne de connection: $con=mysqli\_connect("sql.free.fr","pacaron210","RtsZZe56!lu","pacaron210");

#### en distant sur byethost

dans ce cas dans le cpanel de byethost, je crée une base de donnée nounou, le cpanel crée quant à lui une base b3\_27845068\_nounou (b3\_27845068 c’est mon identifiant)

je peux alors lire les infos:



du coup mes données sont

serveur: “sql113.byethost3.com”  
login: “b3\_27845068”  
password: *monmotdepasse*  
base de donnée: “b3\_27845068\_nounou”

et ma ligne:

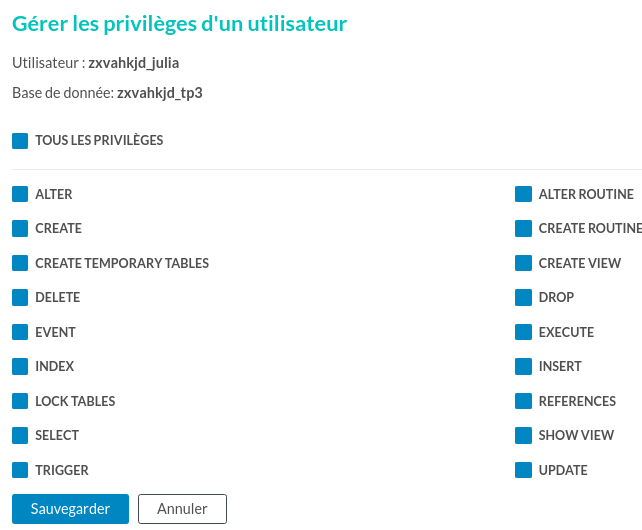
$con=mysqli\_connect("sql113.byethost3.com","b3\_27845068",monmotdepasse,"b3\_27845068\_nounou");

#### en distant sur planet hoster

dans ce cas dans le cpanel de planethoster, (si mon identifiant est par exemple “zxvahkjd”) je crée une base de donnée tp3, le cpanel crée quant à lui une base zxvahkjd\_tp3 je lui crée un utilisateur zxvahkjd\_julia . (et un mot de passe)



Je clique ensuite sur l’utilisateur et je lui donne les droits (je coche tout)



du coup mes données sont

serveur: “locahost”  
login: “zxvahkjd\_julia”  
password: *motdepassedemoncompte*  
base de donnée: “zxvahkjd\_tp3”

et ma ligne:

$con=mysqli\_connect("locahost","zxvahkjd\_julia",monmotdepasse,"zxvahkjd\_tp3");

## SUITE du TP

<?php

echo "test";

$con=mysqli\_connect("localhost","root","","nounou");//1

if (mysqli\_connect\_errno())

{

echo "probleme de connexion à MySQL: " . mysqli\_connect\_error();

}

$resultat= mysqli\_query($con,"SELECT \* FROM nounou"); //2

$nombre = mysqli\_num\_rows($resultat);//3

echo "Il y a $nombre enregistrement(s) dans la table nounou."; //4

mysqli\_close($con); // 5

?>

### vous devez obtenir

Il y a 2 enregistrement(s) dans la table nounou.

***Attention: ne pas confondre la base nounou et la table nounou!!***

### **Explication du code ci-dessus :**

1. On se connecte à la base de données :  
   Host : Dans notre cas le HOST est "localhost".  
   Login : Ensuite vous devez mettre le "login" pour accéder à la base en localhost c'est root  
   Password : Ensuite vous devez mettre le "mot de passe",  en localhost c'est rien du tout, à moins que vous ayez défini un password.  
   bd : Ensuite vous devez mettre la base de donnée que vous souhaitez interroger
2. La fonction mysqli\_query() permet de passer une requête SQL vers la base de données, c'est évidemment l'un des attraits intéressant de PHP (notez que nous définissons au passage la variable $resultat qui contient le résultat de la requête).
3. La fonction mysqli\_num\_rows() permet de compter le nombre d'enregistrements que retourne la requête "$nombre" dans notre cas : 2 puisqu'il s'agit d'un simple "select " sur la table sans aucune condition, nous initialisons donc une variable $nombre qui contient la valeur 2.
4. Il ne reste plus qu'à afficher le nombre de résultat avec un echo() de $nombre.
5. Et pour finir on referme la connexion avec la fonction mysqli\_close(). Cette fonction n'est pas vraiment obligatoire car par défaut la connexion sera coupée automatiquement à la fin de l'exécution du script.

## Activité 3 Interroger la base avec un script php

| Ce qu'il faut faire Maintenant que nous nous sommes connectés à la base de données, nous allons interroger une table pour en extraire les résultats |
| --- |

.

### le fichier php

<!DOCTYPE html>

<html>

<head>

<title>test</title>

<meta http-equiv="Content-Type" content="text/HTML; charset=utf-8" />

</head>

<body>

<?php

include("config.php"); //1

$con=mysqli\_connect($serveur, $login, $password, $base);//1

if (mysqli\_connect\_errno())

{

echo "probleme de connexion à MySQL: " . mysqli\_connect\_error();

}

$result = mysqli\_query($con, "select \* from enfants");

while($donnees = mysqli\_fetch\_assoc($result))

{

echo $donnees['id']." <b>".$donnees['nom']." </b><i>".$donnees['prenom']."</i>: ".$donnees['date\_naissance']."<br>";

}

?>

<a href="index.html">retour</a>

</body>

</html>

*affichera les différents enfants gardés*

### **Explication du code ci-dessus :**

Remarque 1 et 1 bis: On stocke dans le fichier config.php, les instructions de connection, cela permet de changer dans la totalité des scripts ces instructions, lorsque l'on désire publier le site chez un hébergeur.

Ici le fichier config.php est:

<?php

$serveur="localhost";

$login="root";

$password="";

$base="nounou";

?>

*Remarque: en publiant chez un autre hébergeur ce fichier deviendrait:*

*chez free*

<?php

$serveur="sql.free.fr";

$login="monloginfree";

$password= *monpasswordfree*;

$base="monloginfree";

?>

*chez byesthost*

<?php

$serveur="sql113.byethost3.com";

$login="b3\_27845068";

$password= *motdepassedemoncompte*;

$base="b3\_27845068\_nounou";

?>

*chez planethoster*

<?php

$serveur="localhost";

$login="zxvahkjd\_julia";

$password= *lepassworddecetutilisateur*;

$base="zxvahkjd\_tp3";

?>

etc…..

## Activité 4: Bases et formulaire

| Ce qu'il faut faire Dans cet exercice nous allons voir comment alimenter une table avec les données qui proviennent d'un même formulaire.  Nous allons ajouter une nounou dans la base. |
| --- |

 le fichier ajout .html

<!DOCTYPE html>

<html>

<head>

<title>ajout nounou</title>

<meta http-equiv="Content-Type" content="text/HTML; charset=utf-8" />

</head>

<body>

<form method="POST" action="ajout.php">

<p>Ajouter une nounou</p>

<p>Quel est le nom de la nounou à ajouter</p>

<p><input type="text" name="nom" size="20"></p>

<p>Quel est le prénom de la nounou</p>

<p><input type="text" name="prenom" size="20"></p>

<p>Quel est le code postal de la nounou</p>

<p><input type="text" name="code" size="5"></p>

<p><input type="submit" value="Envoyer"></p>

</form>

</body>

</html>

le fichier ajout.php:

<!DOCTYPE html>

<html>

<head>

<title>ajout nounou php</title>

<meta http-equiv="Content-Type" content="text/HTML; charset=utf-8" />

</head>

<body>

<?php

$nom=$\_POST["nom"];

$prenom=$\_POST["prenom"];

$code=$\_POST["code"];

if(empty($nom) OR empty($prenom)){

echo "Attention, il faut remplir tous les champs!";

}

else{

echo "la nounou que vous venez de saisir dans la base est : <br>

<b>$nom </b><i> $prenom </i>";

include("config.php");

$con=mysqli\_connect($serveur, $login, $password, $base);//1

if (mysqli\_connect\_errno())

{

echo "probleme de connexion à MySQL: " . mysqli\_connect\_error();

}

mysqli\_set\_charset($con, "utf8");

if (!(mysqli\_query($con,"insert into nounou values (0, '$nom','$prenom','$code')"))) {

echo "erreur: ". mysqli\_error($con);

}

}

mysqli\_close($con); // 5

?>

<a href="ajout.html">retour</a>

</body>

</html>

| **LE TP à me rendre!! compléter les fichiers HTML et les fichiers PHP** |
| --- |

## pour:

1. Lister les enfants.
2. Afficher les enfants actuellement gardés.
3. Afficher les nounous libres  entre 2 dates spécifiques.
4. Insérer un nouvel enfant et l'inscrire à une nounou
5. Arrêter la garde d'un enfant

| aparté:  pour cette question 3, il vous faudra trouver une nounou libre et donc prendre en compte à la fois les nounous qui n’ont jamais gardées d’enfants et les nounous pour lesquelles la date de fin d’un enregistrement dans la table planning est dépassée.  Cette prise en compte est moins innocente qu’elle n’y paraît en effet lorsque vous faites une jointure avec la condition INNER (voir partie précédente) vous faites *“l’hypothèse du monde clos”* vous considérez que si une nounou n'apparaît pas dans la table planning c’est qu’elle n’existe pas.  Sans le savoir, nous faisons assez systématiquement l'hypothèse du monde clos. C'est à dire que nous considérons que l'absence d'information, n'est pas une information. Si vous demandez à une secrétaire de vous communiquer les coordonnées des clients qui sont domiciliés à Paris, elle vous donnera une liste où figurera autant de fois le nom PARIS qu'il y a de clients dans la liste, et ceci paraît bien normal ! Sauf que... comme l'aurait fait tout un chacun, votre secrétaire a fait l'hypothèse du monde clos sans le savoir en présumant que les clients pour lesquels l'adresse n'est pas renseignée ne sont pas domiciliés à PARIS !  C'est cela l'hypothèse du monde clos : considérer que l'absence d'information doit être synonyme de critère de discrimination... La jointure externe permet de contrer l'hypothèse du monde clos en considérant qu'en cas d'absence de jointure entre une table et l'autre, on ne supprime pas pour autant l'information.  dans ce cas précis il vous faudra faire **une jointure externe**:  "SELECT  nounou.nom,  nounou.prenom,  planning.date\_debut,  planning.date\_fin  FROM  nounou  **LEFT OUTER JOIN**  planning  ON  planning.id\_nounou = nounou.id  WHERE  planning.date\_debut > '$date\_fin' OR planning.date\_fin < '$date\_debut' OR ISNULL(planning.date\_debut)" |
| --- |

Attention: Pour simuler les caractères "" à l'intérieur d'un echo il faut utiliser \" et \"

C'est ce travail que vous devez réaliser:

| vous devez donc me rendre un fichier zip contenant vos fichiers:   * 1 fichier sql contenant la description de votre base de donnée (faire un export dans phpmyadmin) * vos fichiers html, php, css constituant votre site. * + une adresse où je puisse consulter votre travail! |
| --- |

