

DES PETITS TRUCS

```
<?php  
    echo date("H:i:s");  
?>
```

→ 10 :30 :45

```
<?php  
    $variable = "Hello World !";  
    echo gettype($variable);  
    if(is_string($variable))  
        echo "<br /> Mais je t'ai dit que c'est un string !";  
?>
```

→ string

Mais je t'ai dit que c'est un string !

- Les doubles cotes interprètent les variables alors que les simples cotes ne le fait pas.

Accéder à un caractère d'une chaîne

```
<?php  
    $chaine = "Hello World !";  
    echo $chaine{1};  
?>
```

→ e

Déclaration des constantes

```
<?php  
    define("mon_nom", "Hamza Bounasseh");  
    echo mon_nom; // Les constantes ne comportent pas de $ devant leur nom  
    echo "<br />";  
    if(define("mon_nom"))  
        echo "OUI";  
    else  
        echo "NON";  
?>
```

→ Hamza Bounasseh

OUI

Exemple de constantes prédéfinies

```
<?php  
    echo __FILE__; // double underscore  
    echo "<br>";  
    echo __LINE__; // double underscore  
    echo "<br>";  
    echo PHP_VERSION;  
    echo "<br>";  
    echo PHP_OS;  
    echo "<br>";  
?>
```

→ C:\wamp64\www\php\index.php

4

7.2.10

WINNT

Test d'existence

```
<?php
    $variable = 10;
    if(isset($variable))
        echo "Cette variable existe effectivement !";
    echo "<br />";
    if(define($variable)) // defined ne marche qu'avec les constantes
        echo "Mais je t'ai dis que cette variable existe effectivement !";
    else
        echo "Cette variable n'existe pas !";
?>
```

→ Cette variable existe effectivement !

Cette variable n'existe pas !

Destruction

```
<?php
    $variable = 10;
    unset($variable);
    if(isset($variable))
        echo "Cette variable existe effectivement !";
    else
        echo "Cette variable n'existe pas !";
?>
```

→ Cette variable n'existe pas!

Traitements de base

```
<?php
    $variable = "Hello ";
    $variable .= "World !";
    echo $variable;
?>
```

→ Hello World !

```
<?php
    $a="1";
    $b=1;
    if($a==$b)
        echo "a et b ont la même valeur !";
    echo "<br />";
    if($a!=$b)
        echo "a et b n'ont pas la même valeur !";
    echo "<br />";
    if($a===$b)
        echo "a et b ont la même valeur et le même type !";
    echo "<br />";
    if($a!==$b)
        echo "a et b n'ont pas la même valeur ou le même type !";
?>
```

→ a et b ont la même valeur !

a et b n'ont pas la même valeur ou le même type !

```
<?php
    if()
    {}
    elseif()
    {}
    elseif()
    {}
    elseif()
    {}
    else
    {}
?>
```

Switch

```
<?php
    $mois = 2;
    switch ($mois)
    {
        case 1:
            echo "Je suis Janvier !";
            break;
        case 2:
            echo "Je suis Février !";
            break;
        case 3:
            echo "Je suis Mars !";
            break;
    }
?>
```

→ Je suis Février !

Les structures répétitives //exactement comme le langage C.

LES TABLEAUX (ARRAY) ET LES FONCTIONS

Les types de tableaux :

1. Tableau indicé
2. Tableau associatif

1. Tableau indicé

- Les clés sont numériques.
- Les valeurs sont de type quelconque.
- Accès aux éléments par l'intermédiaire d'indices numériques.

```
<?php
    $jour=array("Lundi","Mardi","Mercredi");
    echo $jour[0];
    echo " ";
    $jour=["Lundi","Mardi","Mercredi"];
    echo $jour[1];
?>
```

→ Lundi Mardi

- On peut assigner les valeurs aux éléments du tableau.

```
<?php
    $jour=array("Lundi","Mardi","Mercredi");
    $jour[0]="Hamza";
    echo $jour[0];
?>
```

→ Hamza

Le nombre d'éléments dans un tableau

```
<?php
    $jour=array("Lundi","Mardi","Mercredi");
    echo count($jour);
?>
```

→ 3

Parcourir un tableau

```
<?php
    $tab[] = 1;
    $tab[] = 2;
    $tab[] = 3;
    $tab[] = 4;
    $tab[] = 5;
    for($i=0;$i<count($tab);$i++)
    {
        echo $tab[$i]." ";
    }
?>
```

→ 1 2 3 4 5

- On peut affecter un tableau à un autre. **\$tab1=\$tab2**

Quelques fonctions de manipulation des tableaux

```
<?php
    $tab=[1,2,3,4];
    print_r($tab);
    array_push($tab,5);
    echo "<br />";
    print_r($tab);
    array_push($tab,6,7);
    echo "<br />";
    print_r($tab);
    array_unshift($tab,0);
    echo "<br />";
    print_r($tab);
?>
```

→ Array ([0] => 1 [1] => 2 [2] => 3 [3] => 4)
 Array ([0] => 1 [1] => 2 [2] => 3 [3] => 4 [4] => 5)
 Array ([0] => 1 [1] => 2 [2] => 3 [3] => 4 [4] => 5 [5] => 6 [6] => 7)
 Array ([0] => 0 [1] => 1 [2] => 2 [3] => 3 [4] => 4 [5] => 5 [6] => 6 [7] => 7)

- **array_pop(\$tab)** : Retourne et supprime le dernier élément du tableau.
- **array_shift(\$tab)** : Retourne et supprime le premier élément du tableau.
- **sort(\$tab)** : trie le tableau.

```
<?php
    $tab=array("e","d","c","b","a");
    sort($tab);
    print_r($tab);
?>
```

→ Array ([0] => a [1] => b [2] => c [3] => d [4] => e)
 Ou même :

```
<?php
    $tab=array("e","d","c","b","a");
    sort($tab);
    $tab1=$tab;
    print_r($tab1);
?>
```

→ Array ([0] => a [1] => b [2] => c [3] => d [4] => e)
 Mais pas :

```
<?php
    $tab=array("e","d","c","b","a");
    $tab1=sort($tab);
    print_r($tab1);
?>
```

→ 1

- On trouve d'autres fonctions de tri comme : rsort(), asort(), arsort(), ksort(), krsort()
- **explode("separateur",\$chaine);** // \$tab = explode(":", "rouge:vert:bleu");
- **implode("separateur",\$tab);** // \$chaine = implode(":", \$tab);

2. Les tableaux associatifs

- Les clés d'index : des chaînes de caractères.
- \$tableau = array(clé0 => val0, clé1 => val1,..., cléN => valN);
- Les éléments sont référencés par ces indices_chaîne :

\$tableau["indice_chaîne"] = valeur;

```
<?php  
    $tab= ["nom"=>"Bounasseh", "prenom"=>"Hamza", "age"=>20];  
    print_r($tab);  
?>
```

→ Array ([nom] => Bounasseh [prenom] => Hamza [age] => 20)

```
<?php  
    $tab= ["nom"=>"Bounasseh", "prenom"=>"Hamza", "age"=>20];  
    echo "<br />Je m'appelle ".$tab["nom"]." ".$tab["prenom"]." et j'ai  
".$tab["age"]." ans.";  
?>
```

→ Je m'appelle Bounasseh Hamza et j'ai 20 ans.

```
echo "<br />Je m'appelle $tab[nom] $tab[prenom] et j'ai $tab[age] ans";
```

→ Erreur ; Les éléments d'un tableau ne s'interprètent pas comme les variables.

3. Tableau à plusieurs dimensions

```
<?php  
    $tab1=array(1,2,3);  
    $tab2=array(4,5,6);  
    $tab3=array("nom"=>"Bounasseh", "prenom"=>"Hamza", "age"=>20);  
    $tab=array($tab1,$tab2,$tab3);  
    echo $tab[2]["prenom"];  
?>
```

→ Hamza

```
<?php  
    $tab =  
    [  
        [1,2,3],  
        [4,5,6],  
        ["nom"=>"Bounasseh", "prenom"=>"Hamza", "age"=>20]  
    ];  
    echo $tab[2]["prenom"];  
?>
```

→ Hamza

```
<?php  
    $tab[0][0]=1;  
    $tab[0][1]=2;  
    $tab[0][2]=3;  
    $tab[1][0]=4;  
    $tab[1][1]=5;  
    $tab[1][2]=6;  
    $tab[2]["nom"]="Bounasseh";  
    $tab[2]["prenom"]="Hamza";  
    $tab[2]["age"]=20;  
    echo $tab[2]["prenom"];  
?>
```

→ Hamza //c'est trois derniers exemples sont équivalents.

- **current(\$tab)** : Retourne la valeur du dernier élément manipulé.
- **key(\$tab)** : Retourne la clé du dernier élément manipulé.
- **reset(\$tab)** : se positionne sur le premier élément et renvoie sa valeur.
- **end(\$tab)** : se positionne sur le dernier élément et renvoie sa valeur.
- **next(\$tab)** : avance d'un élément et renvoie la valeur de l'élément suivant.
- **prev(\$tab)** : recule d'un élément et renvoie la valeur de l'élément précédent.

4. La fonction Foreach()

- **each(\$tab)** : Retourne l'élément courant et avance le pointeur du tableau de 1.

```
<?php
$tab = ["nom"=>"Bounasseh", "prenom"=>"Hamza", "age"=>20];
foreach ($tab as $var)
{
    echo $var."  ";
}
echo "<br />";
foreach ($tab as $x=>$y)
{
    echo $x."=".$y."  ";
}
?>
```

→ Bounasseh Hamza 20
 nom=Bounasseh prenom=Hamza age=20

5. Passage par référence

```
<?php
$tab = [1,2,3];
foreach ($tab as $var)
{
    $var--;
}
echo $tab[0]+$tab[1]+$tab[2];
echo "<br />";
foreach ($tab as &$var)
{
    $var--;
}
echo $tab[0]+$tab[1]+$tab[2];
?>
```

→ 6
 3

6. Modifier la portée des variables locales d'une fonction (Global)

```
<?php
$a=1;
$b=2;
function somme()
{
    $a=0;
    $b=0;
    global $a, $b;
    return $a+$b;
}
echo somme();
?>
```

→ 3

```
<?php
    $a=1; $b=2;
    function somme()
    {
        return $GLOBALS["a"]+$GLOBALS["b"];
    }
    echo somme();
?>
```

→ 3 //c'est deux derniers exemples sont équivalents.

7. Modifier la portée des variables locales d'une fonction (Static)

```
<?php
    function test()
    {
        static $a=0;
// La variable $a est initialisée uniquement lors du premier appel à la fonction
        $a++;
        echo $a." ";
    }
test();
test();
test();
?>
```

→ 1 2 3

8. Retourner plusieurs valeurs d'une fonction

```
<?php
    function trigo($angle)
    {
        return array(cos($angle),sin($angle),tan($angle));
    }
    list($a,$b,$c)=trigo(pi()/4);
    echo "cos(pi/4)=$a      sin(pi/4)=$b      tan(pi/4)=$c";
?>
```

→ cos(pi/4)=0.7071 sin(pi/4)=0.7071 tan(pi/4)=1

LES CHAINES DE CARACTERES

Traitements de chaines

```
<?php
    echo ord("a");
    echo "<br />";
    echo chr(97);
?>
```

→ 97 //String to ASCII
a //ASCII to String

```
<?php
    $chaine="Hello World !";
    echo strlen($chaine);
?>
```

→ 13 //13 caractères

```
<?php
    $chaine="Hello World, How are You ?";
    echo str_word_count($chaine);
?>
```

→ 5 //5 mots

```
<?php
    $chaine="Hello World, How are You ?";
    $tab=str_word_count($chaine,1);
    print_r($tab);
?>
```

→ Array ([0] => Hello [1] => World [2] => How [3] => are [4] => You)
//Lister les mots d'une chaîne dans un tableau

- **addslashes()** :

- Permet de protéger automatiquement les guillemets, apostrophes et barres obliques inverses en les préfixant automatiquement.
- Cette fonction est particulièrement utile dans la préparation de requêtes SQL.

```
<?php  
    $chaine=addslashes("J'aime l'argent ^^' !");  
    echo $chaine;  
?>
```

→ J'aime l'argent ^^' !

- **str_replace(\$cherche,\$remplace,\$texte)** : Pour remplacer un motif dans une chaîne.

```
<?php  
    $chaine="Mon nom est Bounasseh, mais vous pouvez m'appeler Hamza !";  
    echo str_replace("Hamza","Monsieur Hamza",$chaine);  
?>
```

→ Mon nom est Bounasseh, mais vous pouvez m'appeler Monsieur Hamza !

- On trouve aussi : **str_ireplace()**, **substr_replace()**.

- **trim()** : Retire les caractères blancs avant et après la chaîne de référence.

```
<?php  
    $chaine=" Hamza      Bounasseh      !";  
    echo trim($chaine);  
?>
```

→ Hamza Bounasseh !

- **strtoupper()** & **strtolower()** : Convertissent respectivement les chaînes de caractères en majuscules et minuscules.
- **ucfirst()** & **ucwords()** : Convertissent en majuscules respectivement le premier caractère de la chaîne et le premier caractère de chaque mot.

```
<?php  
    $chaine="hAmzA bOUUnAssEh";  
    echo strtoupper($chaine);  
    echo "<br />";  
    echo strtolower($chaine);  
    echo "<br />";  
    echo ucfirst($chaine);  
    echo "<br />";  
    echo ucwords($chaine);  
?>
```

→ HAMZA BOUNASSEH

hamza bounasseh

HAmzA bOUUnAssEh

HAmzA BOUnAssEh

D'autres fonctions peu fréquentes

- **strpos()** :

- Permet de connaître la position d'une chaîne dans une autre.
- La valeur FALSE est renvoyée si la chaîne n'est pas trouvée.
- On trouve aussi : **stripos()**, **strrpos()**, **strrstr()**, **stristr()**, **strrchr()**.

- **substr()** : Pour récupérer une sous-chaîne si on en connaît la position.

LE TRAITEMENT DES FORMULAIRES

Méthode d'envoi d'un formulaire

- **\$_GET et \$_POST sont des tableaux de données associatifs et superglobaux.**
- **Méthode GET (transmission par URL) :** Une des façons les plus simples pour faire transiter des données de page en page est l'insertion de ces données dans l'URL.

```
<html lang="fr">
    <form action="index.php" method="GET">
        <label>Entrer votre prenom :</label>
        <input type="text" name="prenom"><br />
        <label>Entrer votre nom :</label>
        <input type="text" name="nom"><br />
        <input type="submit" name="send" value="valider">
    </form>
</html>
<?php
    if(isset($_GET["send"]))
        echo "Votre prenom est ".$_GET["prenom"]." et votre nom est ".
    ".$_GET["nom"];
?>
```

- **Méthode POST :** elle cache les informations transmises. De plus, elle permet d'envoyer des données importantes en taille et assure la gestion de l'envoi de fichiers.

```
<form action="index.php" method="POST">
</form>
<?php
if(isset($_POST["send"]))
    echo "Votre prenom est ".$_POST["prenom"]." et votre nom est ".$_POST["nom"];
?>
```

Récupération des données saisie dans un formulaire

- Elle diffère selon le type de la réponse (unique, multiple).
 - Réponse unique : Zone de texte ; Bouton Radio ;
Zone de texte multiligne ; Liste de sélection à choix unique.
 - Réponses multiples : Cases à cocher ; Liste de sélection à choix unique.

➤ Réponse unique :

L'élément choisi est récupéré du tableau \$_POST par son nom.

Il est récupéré dans une variable (**\$x=\$_POST["nom_du_champs"]**).

➤ Réponses multiples :

Le traitement d'une liste déroulante

```
<html lang="fr">
    <form action="index.php" method="POST">
        <select name="couleur">
            <option value="R">Couleur Rouge</option>
            <option value="G" selected="selected">Couleur Verte</option>
            <option value="B">Couleur Bleue</option>
        </select> <br />
        <input type="submit" name="send" value="Envoyer">
    </form>
</html>
<?php
    if(isset($_POST["send"]))
        echo "Vous avez sélectionné : ".$_POST["couleur"];
?>
```

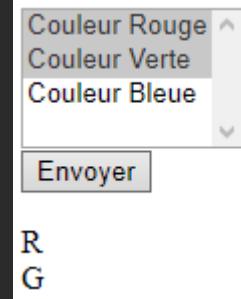
value

↓

Vous avez sélectionné : G

Le traitement d'une liste défilante

```
<html lang="fr">
<form action="index.php" method="POST">
    <select name="couleur[]" multiple="multiple">
        <option value="R">Couleur Rouge</option>
        <option value="G" selected="selected">Couleur Verte</option>
        <option value="B">Couleur Bleue</option>
    </select> <br />
    <input type="submit" name="send" value="Envoyer">
</form>
</html>
<?php
if(isset($_POST["send"]))
{
    $tab = $_POST["couleur"];
    foreach($tab as $var)
    {
        echo $var."<br>";
    }
}
?>
```



Le traitement de cases à cocher

```
<html lang="fr">
<form action="index.php" method="POST">
    <input type="checkbox" name="couleur[]" value="R">Rouge
    <input type="checkbox" name="couleur[]" value="G">Vert
    <input type="checkbox" name="couleur[]" value="B">Bleu
    <br />
    <input type="submit" name="send" value="Envoyer">
</form>
</html>
<?php
if(isset($_POST["send"]))
{
    $tab = $_POST["couleur"];
    foreach($tab as $var)
    {
        echo $var."<br>";
    }
}
?>
```



LES COOKIES

- **Un cookie** : un petit fichier texte stocké sur le disque dur du visiteur du site. On peut y sauvegarder plusieurs informations concernant ce visiteur et les réutiliser lors de sa prochaine visite.

Créer un cookie

```
<?php
    //On définit une durée de vie de notre cookie (en secondes), donc un an dans
    //notre cas
    $duree_cookie=365*24*60*60;
    //On envoie un cookie de nom 'monNom' et de valeur 'Hamza'
    setcookie("monNom", "Hamza", time()+$duree_cookie);
?>
```

Exemple

```
<?php
    if(isset($_COOKIE["monNom"]))
    {
        echo "<h1> Bonjour ".$_COOKIE["monNom"]."</h1>";
    }
    else
    {
        ?>
        <form action="index.php" method="GET">
            <!-- Comme on peut faire le traitement dans un deuxième fichier
"traitement.php" et retourner avec un header --&gt;
            &lt;input type="text" name="name" /&gt;
            &lt;input type="submit" name="send" value="Envoyer"&gt;
        &lt;/form&gt;
        &lt;?php
        if(isset($_GET["name"]))
        {
            header("location:index.php"); // Sinon, on sera amenés à actualiser la page
            setcookie("monNom", $_GET["name"], time() + 365 * 24 * 3600);
        }
    }
?&gt;</pre>

```

Bonjour Hamza

LES SESSIONS

- **Les sessions** : sont un moyen de stocker des types de données simples (texte, nombres, ...) mais pas de ressources comme des images ou BDD.
- Les informations sont stockées dans des fichiers, sur le serveur, à chaque session correspond un fichier.
- Chaque session est désignée par un nom et un identifiant.
- Lorsque le visiteur accepte les cookies, l'identifiant de la session est stocké dans un cookie.
- La session est détruite à la fermeture du navigateur, ou au bout de 30 minutes.

Quand est ce qu'on utilise les sessions ?

- Les espaces membres et accès sécurisés avec authentification.
- Gestion d'un caddie sur un site de vente en ligne.
- Formulaires éclatés sur plusieurs pages.
- Stockage d'informations relatives à la navigation de l'utilisateur (thème, langues ...).

Initialisation (et restauration) d'une session

- **session_start()** : PHP introduit nativement cette unique fonction permettant de démarrer ou de continuer une session.
- Il faut appeler **session_start()** sur chaque page utilisant le système de session AVANT d'écrire votre code HTML.

Sauvegarder et récupérer une variable

```
<?php
    session_start();
    $_SESSION["name"] = "Hamza";
?>
    <a href="temp.php">Cliquer ICI !</a>
<?php
?>
```

Cliquer ICI !

```
<?php
    session_start();
    echo $_SESSION["name"];
?>
```

Hamza

temp.php

Vérifier l'existence d'une variable

```
<?php // Pour jouer le tour, il faut actualiser la page à chaque fois
session_start();
$i=&$_SESSION["i"]; // Un alias
if(!isset($i)) // Savoir si la variable appartient à la session
    $i=0;
if($i==5)
{
    echo "OK OK tu as gagné, cette variable existe bel et bien !";
    unset($_SESSION["i"]); // Destruction de la variable
}
else
{
    echo "Je te dis que cette variable n'est pas définie !";
    $i++;
}
?>
```

Exemple d'un formulaire de connexion

Exemple du cours

```
<?php
if(isset($_POST["submit"]))
{
    $login=isset($_POST["login"]) ? $_POST["login"] : "";
    $pass=isset($_POST["pass"]) ? $_POST["pass"] : "";
    if($login=="Hamza" && $pass=="123")
    {
        $_SESSION["login"]="Hamza";
        $_SESSION["pass"]="123";
        echo '<a href="Acceuil.php"> Acceuil </a>';
    }
    else
        echo '<h1> Erreur de Connexion ! </h1>';
}
else
    echo '
    <center>
        <fieldset>
            <legend> Formulaire de connexion</legend>
            <form action="#" method="POST">
                <label>ENTRER VOTRE LOGIN : </label>
                <input type="text" name="login"/> <br /><br />
                <label>ENTRER VOTRE MOT DE PASSE : </label>
                <input type="password" name="pass" /><br /><br />
                <input type="submit" name="submit" value="CONNEXION" />
            </form>
        </fieldset>
    </center> ';
```

Le mien

```
<?php
    session_start();
    if(isset($_SESSION["login"]) && isset($_SESSION["pass"]))
        header("location:acceuil.php");
    else
    {
?>
<center>
    <fieldset>
        <legend> Formulaire de connexion</legend>
        <form action="#" method="POST">
            <label>ENTRER VOTRE LOGIN : </label>
            <input type="text" name="login"/> <br /><br />
            <label>ENTRER VOTRE MOT DE PASSE : </label>
            <input type="password" name="pass" /><br /><br />
            <input type="submit" name="send" value="CONNEXION" />
        </form>
    </fieldset>
</center>
<?php
    if(isset($_POST["login"]) && isset($_POST["pass"]))
    {
// Attention, "" est une valeur possible et considérée pour $_POST["..."]
        if($_POST["login"]!="" && $_POST["pass"]!="")
        {
            $_SESSION["login"]=$_POST["login"];
            $_SESSION["pass"]=$_POST["pass"];
            header("location:acceuil.php");
        }
    }
?>
```

index.php

```
<?php
    session_start();
    if(isset($_SESSION["login"]) && isset($_SESSION["pass"]))
    {
        echo "<h1>Bienvenue ".$_SESSION["login"]."</h1>";
    ?>
        <form action="#" method="POST">
            <button name="logout"> DECONNEXION </button>
        </form>
    <?php
        if(isset($_POST["logout"]))
        {
            unset($_SESSION["login"]);
            unset($_SESSION["pass"]);
            session_destroy();
            header("location:index.php");
        }
    }
    else
    {
        echo "<h1> Erreur de connexion ! </h1>";
    ?>
        <a href="index.php">Page d'authentification</a>
    <?php
    }
?>
```

acceuil.php

Supprimer une variable d'une session

unset(\$_SESSION['prenom']);

Détruire une session

session_destroy();

LA CONNEXION A UNE BASE DE DONNEES MYSQL

- MySQL : est un serveur de bases de données relationnelles SQL.

Moteurs de stockage inclus

- MyISAM : Utilise des fichiers qui grandissent au fur et à mesure que la base grossit.
Il ne supporte pas les transactions, ni les clefs étrangères.
- InnoDB : moteur créé et maintenu par InnoDB (Filiale d'Oracle).
Il gère les transactions et les clefs étrangères. En contrepartie, les bases qui l'utilisent occupent bien plus d'espace sur le disque.

Les API PHP-Mysql

- Pour manipuler les bases de données, PHP possède plus d'une vingtaine d'extensions et des couches d'abstraction.

Pour la base de données MySQL, trois API différentes :

- mysql : API d'origine, aujourd'hui obsolète.
- mysqli : mysql improved, avec de nouvelles fonctionnalités.
- PDO_MySQL : couche d'abstraction d'accès aux données à travers des PHP Data Objects.

LES OPERATIONS CRUD SUR UNE BDD MYSQL ET LA GESTION DES FICHIERS

Manipulation d'une BDD

- Connexion au SGBD et sélection d'une base.
- Exécution d'une requête.
- Récupération et utilisation du résultat.
- Fermeture de la connexion.

Connexion à une BDD MySQL

- Au cours de la communication PHP-MySQL, il faut spécifier :

- Le nom du serveur.
- Le nom de l'utilisateur.
- Le mot de passe.
- Le nom de la BDD.

La fonction de connexion

\$link = mysqli_connect(\$server,\$user,\$password,\$base) ;

- ✓ Cette fonction retourne une ressource \$link, qui représente le lien de connexion vers la base de données sélectionnée.

```
<?php  
    $link = mysqli_connect("localhost", "root", "12345678", "biblio")  
    or die("Echec de connexion à la base");  
?>
```

```
<?php  
    $link = mysqli_connect("ent.uit.ac.ma", "user_ent", "12345678", "ENT")  
    or die("Echec de connexion à la base");  
?>
```

Fermer une connexion

mysqli_close(\$link) ;

- ✓ Permet de fermer la connexion.

Interrogation de la BD

- **mysqli_query(\$requête)** : Pour envoyer une requête à une base de données. Elle prend pour paramètre une chaîne de caractères qui contient la requête SQL et retourne un identificateur de résultat ou FALSE si échec.

Exemple

```
<?php
    $requête = "SELECT * FROM etudiant" ;
    $result = mysqli_query($link,$requête) ;
?>
```

- ✓ L'identificateur de résultat \$result peut contenir un ou plusieurs résultats. Il permettra à d'autres fonctions d'extraire ligne par ligne les données retournées.

Dénombrer les résultats

- **mysqli_num_rows(\$result)** : Elle retourne le nombre de lignes du résultats. Elle permet de remplacer le while par un for.

```
<?php
    $nbr = mysqli_num_rows($result);
    echo "Le nombre total des étudiants est $nbr";
?>
```

Extraction des données

- **mysqli_fetch_assoc (\$result)** : Elle retourne une ligne de résultat sous forme d'un tableau associatif dont :

- Les clés étant les noms des attributs.
- Les valeurs associées sont leurs valeurs respectives.

```
<?php
    while($data=mysqli_fetch_assoc($result))
    {
        echo $data['id_etudiant']."<br />";
        echo $data['nom']."<br />";
        echo $data['prenom']."<br />";
        echo $data['date_naissance']."<br />";
        echo "<hr>";
    }
?>
```

Formulaire de connexion (Exemple)

- La table "user" de la base "inscription" :

#	Nom	Type
1	LOGIN	varchar(20)
2	PASSWORD	varchar(20)
3	NOM	varchar(40)
4	PRENOM	varchar(40)
5	VILLE	varchar(40)
6	DATE_NAISSANCE	date

- **LOGIN** ici est une clé primaire.

```

<center>
    <fieldset>
        <legend> Formulaire de connexion</legend>
        <form action="verifier_connexion.php" method="POST">
            <label>ENTRER VOTRE LOGIN : </label>
            <input type="text" name="login"/> <br /><br />
            <label>ENTRER VOTRE MOT DE PASSE : </label>
            <input type="password" name="pass" /><br /><br />
            <input type="submit" name="submit" value="CONNEXION" />
        </form>
    </fieldset>
</center>

```

index.php

```

<?php
    if(isset($_POST["login"]) && isset($_POST["pass"]))
    {
        $login=$_POST["login"];
        $pass=$_POST["pass"];
        if(!empty($login) && !empty($pass))
        {
            $link = mysqli_connect("localhost","root","","inscription");
            $query = "SELECT * FROM USER WHERE login='$login' AND password='$pass'";
            $result = mysqli_query($link,$query);
            $data = mysqli_fetch_assoc($result);
            if($data != FALSE)
            {
                session_start();
                $_SESSION["login"]=$data["login"];
                $_SESSION["nom"]=$data["nom"];
                $_SESSION["prenom"]=$data["prenom"];
                $_SESSION["ville"]=$data["ville"];
                header("location:acceuil.php");
            }
            else
            {
                echo "<h1> Votre LOGIN ou votre mot de passe n'est pas correct ! </h1>";
                echo '<a href="index.php"> Connectez Vous par ICI ! </a>';
            }
        }
        else
        {
            echo "<h1> Veuillez entrer votre LOGIN et votre MOT DE PASSE ! </h1>";
            echo '<a href="index.php"> Connectez Vous par ICI ! </a>';
        }
    }
    else
        header("location:index.php");
?>

```

verifier_connexion.php

```

<?php
    session_start();
    if(isset($_SESSION["login"]))
    {
        echo "Login : ".$_SESSION["login"]."<br />";
        echo "Nom : ".$_SESSION["nom"]."<br />";
        echo "Prenom : ".$_SESSION["prenom"]."<br />";
        echo "Ville : ".$_SESSION["ville"]."<br />";
    ?>
        <form action="#" method="POST">
            <input type="submit" name="submit" value="DECONNEXION" />
        </form>
        <?php
            if (isset($_POST["submit"]))
            {
                session_destroy();
                header("location:index.php");
            }
        ?>
        else
            header("location:index.php");
?>

```

acceuil.php

Formulaire d'inscription

```
<center>
    <fieldset>
        <legend> Formulaire d'Inscription </legend>
        <form action="traitement.php" method="POST">
            <label> LOGIN : </label>
            <input type="text" name="login"/> <br /><br />
            <label> MOT DE PASSE : </label>
            <input type="password" name="pass" /><br /><br />
            <label> NOM : </label>
            <input type="text" name="nom"/> <br /><br />
            <label> PRENOM : </label>
            <input type="text" name="prenom"/> <br /><br />
            <label> VILLE : </label>
            <input type="text" name="ville"/> <br /><br />
            <label> DATE DE NAISSANCE : </label>
            <input type="date" name="date_naissance"/> <br /><br />
            <input type="submit" name="submit" value="INSCRIPTION" />
        </form>
    </fieldset>
</center>
```

[index.php](#)

```
<?php
    $link = mysqli_connect("localhost", "root", "", "inscription");
?>
```

[connexion.php](#)

```
<?php
    include("connexion.php");
    $login = $_POST["login"];
    $password = $_POST["pass"];
    $nom = $_POST["nom"];
    $prenom = $_POST["prenom"];
    $ville = $_POST["ville"];
    $date_naissance = $_POST["date_naissance"];
    $query = "INSERT INTO user (login,password,nom,prenom,ville,date_naissance)
VALUES ('$login','$password','$nom','$prenom','$ville','$date_naissance')";
    $result = mysqli_query($link,$query);
    if($result == TRUE)
        echo "<h1> Félicitation ! Votre compte a été créé avec succès ! </h1>";
    else
        echo "<h1> Une erreur est survenue lors de la création de votre compte !
</h1>";
?>
```

[traitement.php](#)

Le remplissage d'une liste à partir de la base de données

#	Nom	Type
1	<u>LOGIN</u>	varchar(20)
2	<u>PASSWORD</u>	varchar(20)
3	<u>NOM</u>	varchar(40)
4	<u>PRENOM</u>	varchar(40)
5	<u>VILLE</u>	int(11)
6	<u>DATE_NAISSANCE</u>	date

Nom	Type
<u>id_ville</u>	int(11)
<u>lib_ville</u>	varchar(20)

Créer un compte

Login :

Mot de passe:

Nom:

Prénom:

Ville

 RABAT ▾

Date Naissance

```

<?php
    include("connexion.php");
    $query = "SELECT * FROM ville";
    $result = mysqli_query($link,$query);
    // $data = mysqli_fetch_assoc($result); // Faux
/* A ne jamais commettre cette erreur d'affecter cette fonction à une variable
sauf avec une boucle ou une condition if else */
?>
<center>
    <fieldset>
        <legend> Formulaire d'Inscription </legend>
        <form action="traitement.php" method="POST">
            <label> LOGIN : </label>
            <input type="text" name="login"/> <br /><br />
            <label> MOT DE PASSE : </label>
            <input type="password" name="pass" /><br /><br />
            <label> NOM : </label>
            <input type="text" name="nom"/> <br /><br />
            <label> PRENOM : </label>
            <input type="text" name="prenom"/> <br /><br />
            <label> VILLE : </label>
            <select name="ville">
<?php
    while($data = mysqli_fetch_assoc($result)) // Vrai
    {
        echo '<option value="'.$data["id_ville"].'">'.$data["lib_ville"].'</option>';
    }
?>
            </select>
            <br /><br />
            <label> DATE DE NAISSANCE : </label>
            <input type="date" name="date_naissance"/> <br /><br />
            <input type="submit" name="submit" value="INSCRIPTION" />
        </form>
    </fieldset>
</center>

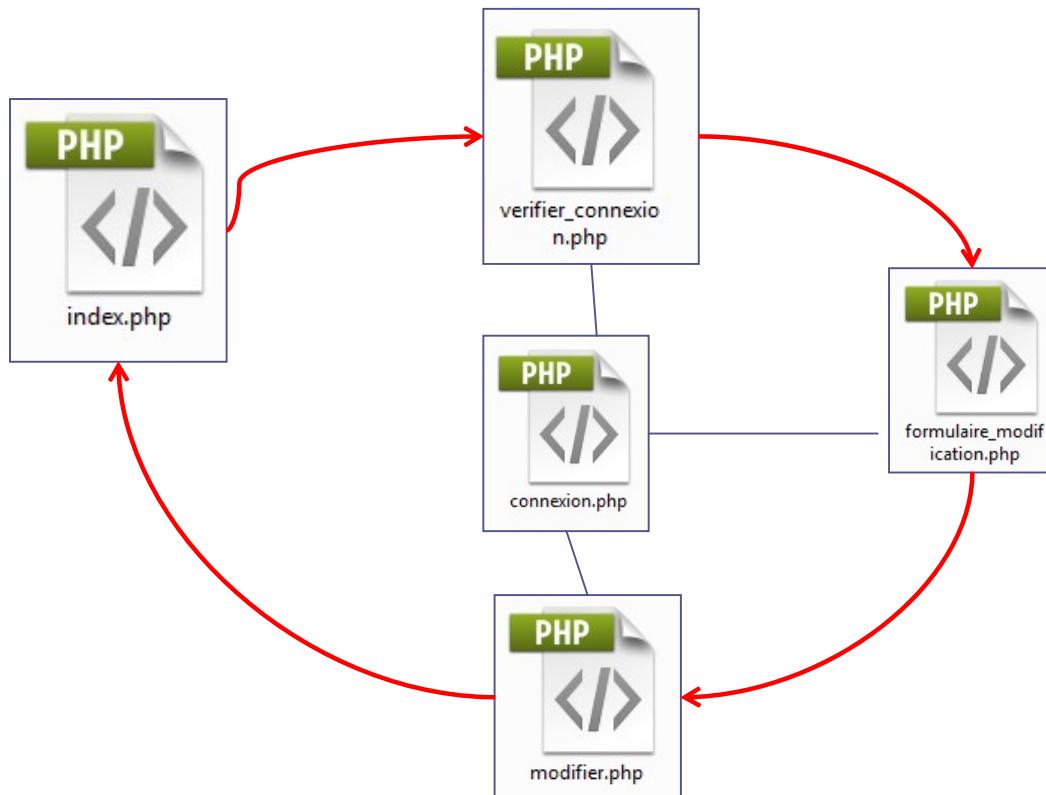
```

index.php

- On préserve les mêmes fichiers *connexion.php* & *traitement.php*.

Modification des données

- Maintenant, on va changer de philosophie, les nominations des fichiers auront d'autres sens.



```
<?php  
    $link = mysqli_connect("localhost", "root", "", "inscription");  
?>
```

connexion.php

```
<?php  
session_start();  
if(isset($_SESSION["connect"]))  
{  
header("location:acceuill.php");  
}  
else  
{  
?>  
<center>  
<fieldset>  
<legend> <h2> FORMULAIRE DE CONNEXION </h2> </legend>  
<form action="" method="POST">  
<label>ENTRER VOTRE LOGIN : </label>  
<input type="text" name="login"/> <br /><br />  
<label>ENTRER VOTRE MOT DE PASSE : </label>  
<input type="password" name="pass" /><br /><br />  
<input type="submit" name="connexion" value="CONNEXION" />  
<input type="submit" name="inscription" value="INSCRIPTION" />  
</form>  
</fieldset>  
</center>  
<?php  
if(isset($_POST["connexion"]))  
{  
if(!empty($_POST["login"]) && !empty($_POST["pass"]))  
{  
$_SESSION["connect"] = 1;  
$_SESSION["login"] = $_POST["login"];  
$_SESSION["pass"] = $_POST["pass"];  
header("location:acceuill.php");  
}  
}  
if(isset($_POST["inscription"]))  
{  
header("location:inscription.php");  
}  
}  
?>
```

index.php

```

<?php
session_start();
if(!isset($_SESSION["connect"]))
header("location:index.php");
else{
include("connexion.php");
$login = $_SESSION["login"];
$pass = $_SESSION["pass"];
$query = "SELECT * FROM user WHERE login = '$login' AND password = '$pass'";
$result = mysqli_query($link, $query);
$data = mysqli_fetch_assoc($result);
if ($data != FALSE) {
?>
<center>
<fieldset>
<legend> VOS INFORMATIONS :</legend>
<table>
<tr>
<th> LOGIN :</th>
<td> <?php echo $data["login"]; ?> </td>
</tr>
<tr>
<th> NOM :</th>
<td> <?php echo $data["nom"]; ?> </td>
</tr>
<tr>
<th> PRENOM :</th>
<td> <?php echo $data["prenom"]; ?> </td>
</tr>
<tr>
<th> VILLE :</th>
<td> <?php echo $data["ville"]; ?> </td>
</tr>
<tr>
<th> DATE DE NAISSANCE :</th>
<td> <?php echo $data["date_naissance"]; ?> </td>
</tr>
<form action="" method="POST">
<tr>
<th><input type="submit" name="modifier" value="MODIFIER VOS INFORMATIONS"></th>
<td><input type="submit" name="deconnexion" value="DECONNEXION"></td>
</tr>
</form>
</table>
</fieldset>
</center>
<?php
if (isset($_POST["modifier"]))
header("location:modification.php");
if (isset($_POST["deconnexion"])) {
session_destroy();
header("location:index.php");
}
?>
<?php
}
else{
echo "<h1> Vérifier vos informations !</h1>";
?>
<form method="POST">
<input type="submit" name="back" value="PAGE D'AUTHENTICATION"/>
</form>
<?php
if (isset($_POST["back"]))
{
session_destroy();
header("location:index.php");
}
}
}
?>

```

```

<?php
session_start();
include("connexion.php");
$query = "SELECT * FROM ville";
$result = mysqli_query($link, $query);
?>
<center>
<fieldset>
<legend> Formulaire d'Inscription</legend>
<form action="traitement_inscription.php" method="POST">
<label> LOGIN : </label>
<input type="text" name="login"/> <br/><br/>
<label> MOT DE PASSE : </label>
<input type="password" name="pass"/><br/><br/>
<label> NOM : </label>
<input type="text" name="nom"/> <br/><br/>
<label> PRENOM : </label>
<input type="text" name="prenom"/> <br/><br/>
<label> VILLE : </label>
<select name="ville">
<?php
while ($data = mysqli_fetch_assoc($result)) // Vrai
{
echo '<option value="' . $data["id_ville"] . '">' . $data["lib_ville"] .
'</option>';
} ?>
</select>
<br/><br/>
<label> DATE DE NAISSANCE : </label>
<input type="date" name="date_naissance"/> <br/><br/>
<input type="hidden" name="insc" value="1" />
<input type="submit" name="inscription" value="INSCRIPTION"/>
</form>
<form method="POST">
<input type="submit" name="annuler" value="ANNULER"/>
</form>
</fieldset>
</center>
<?php
if (isset($_POST["annuler"]))
header("location:index.php");
?>

```

inscription.php

```

<?php
session_start();
if(!isset($_POST["insc"]))
header("location:index.php");
else {
include("connexion.php");
$login = $_POST["login"];
$password = $_POST["pass"];
$nom = $_POST["nom"];
$prenom = $_POST["prenom"];
$ville = $_POST["ville"];
$date_naissance = $_POST["date_naissance"];
$query = "INSERT INTO user (login,password,nom,prenom,ville,date_naissance) VALUES
('$login','$password','$nom','$prenom','$ville','$date_naissance')";
$result = mysqli_query($link, $query);
if ($result == TRUE)
{
echo "<h1> Félicitation ! Votre compte a été créé avec succès ! </h1>";
echo "<a href='index.php'>Connectez Vous d'ici !</a>";
}
else
{
echo "<h1> Une erreur est survenue lors de la création de votre compte ! </h1>";
echo "<a href='index.php'>Page d'Accueil</a><br /><br />";
echo "<a href='inscription.php'>Réessayer encore une fois !</a>";
}
?>

```

traitement_inscription.php

```

<?php
session_start();
if(!isset($_SESSION["connect"]))
header("location:index.php");
else
{
$_SESSION["connect"] = "ON";
include("connexion.php");
$login = $_SESSION["login"];
$pass = $_SESSION["pass"];

$query_1 = "SELECT * FROM user WHERE login='".$login' AND password='".$pass."'";
$result_1 = mysqli_query($link, $query_1);
$data_1 = mysqli_fetch_assoc($result_1);

$query_2 = "SELECT * FROM ville";
$result_2 = mysqli_query($link, $query_2);

$_SESSION["ancien_login"] = $data_1['login'];
?>
<center>
<fieldset>
<legend> MODIFIER VOS INFORMATIONS</legend>
<form action="traitement_modification.php" method="POST">
<label> LOGIN : </label>
<input type="text" name="login" value="<?php echo $data_1['login'] ?>"/>
<br/><br/>
<label> MOT DE PASSE : </label>
<input type="text" name="pass" value="<?php echo $data_1['password'] ?>"/><br/><br/>
<label> NOM : </label>
<input type="text" name="nom" value="<?php echo $data_1['nom'] ?>"/> <br/><br/>
<label> PRENOM : </label>
<input type="text" name="prenom" value="<?php echo $data_1['prenom'] ?>"/>
<br/><br/>
<label> VILLE : </label>
<select name="ville">
<?php
while ($data_2 = mysqli_fetch_assoc($result_2))
{
if ($data_1["ville"] == $data_2["id_ville"])
{
echo '<option value="'. $data_2["id_ville"].'" selected="selected"> ' .
$data_2["lib_ville"].'</option>';
}
else
echo '<option value="'. $data_2["id_ville"].'"> ' . $data_2["lib_ville"] . '
</option>';
}
?>
</select>
<br/><br/>
<label> DATE DE NAISSANCE : </label>
<input type="date" name="date_naissance" value="<?php echo
$data_1['date_naissance'] ?>"/>
<br/><br/>
<input type="submit" name="modifier" value="MODIFIER"/>
</form>
<form method="POST">
<input type="submit" name="annuler" value="ANNULER"/>
</form>
</fieldset>
</center>
<?php
if (isset($_POST["annuler"]))
header("location:acceuil.php");
?
?>

```

modification.php

```

<?php
session_start();
if(!isset($_SESSION["connect"]))
header("location:index.php");
else
{
if (isset($_POST["deconnexion"])) {
session_destroy();
header("location:index.php");
}
$ancien_login = $_SESSION["ancien_login"];
include("connexion.php");
$login = $_POST["login"];
$password = $_POST["pass"];
$nom = $_POST["nom"];
$prenom = $_POST["prenom"];
$ville = $_POST["ville"];
$date_naissance = $_POST["date_naissance"];
$query = "UPDATE user SET login='$login', password='$password',
nom='$nom', prenom='$prenom', ville='$ville', date_naissance='$date_naissance'
WHERE login='$ancien_login'";
$result = mysqli_query($link, $query);
if ($result == TRUE) {
echo "<h1> Votre compte a été modifier correctement ! </h1>";
?>
<form method="POST">
<input type="submit" name="deconnexion" value="CONNECTEZ VOUS AVEC LES NOUVELLES
INFORMATIONS !"/>
</form>
<?php
}
else
{
echo "<h1> Une erreur est survenue lors de la modification de vos informations !
</h1>";
echo "<a href='acceuil.php'>Page d'Acceuil</a><br /><br />";
echo "<a href='modification.php'>Réessayer encore une fois !</a>";
}
?>

```

traitement_inscription.php

- Rassemblez tous ces fichiers dans un même dossier dans le chemin : C:\wamp64\www et explorer par vous-même.

ALLEZ PLUS LOIN

- **L'extension PDO** : c'est un outil complet qui permet d'accéder à n'importe quel type de base de données, il vient pour remplacer l'extension **mysqli_**.
- Pour pratiquer la PDO, Veuillez Importer cette BDD dans phpMyAdmin : oc.cm/tablejeux

```

<?php
$bdd = new PDO("mysql:host=localhost;dbname=test","root","");
$result = $bdd->query("SELECT * FROM jeux_video WHERE console='PC'");

while($data = $result->fetch())
    echo $data["nom"]."<br />";                                // VERSION 1
?>

```

```

<?php
$bdd = new
PDO("mysql:host=localhost;dbname=test","root","",array(PDO::ATTR_ERRMODE =>
PDO::ERRMODE_EXCEPTION));
// grace à ce dernoer parametre, on recevera lors des erreurs des messages plus
précise e plus significatifs.

$result = $bdd->query("SELECT * FROM jeux_video WHERE console='PC'");

while($data = $result->fetch())
    echo $data["nom"]."<br />";                                // VERSION 2
?>

```

```

<?php
    $bdd = new PDO("mysql:host=localhost;dbname=test","root","",array(PDO::ATTR_ERRMODE =>
PDO::ERRMODE_EXCEPTION));
    $query = $bdd->prepare('SELECT * FROM jeux_video WHERE console=?');
    $query->execute(array("PC"));
    // PC est un enregistrement du champ 'console' dans la BDD
    while($data = $query->fetch())
    {
        echo $data["nom"]."<br />"; // VERSION 3
    }
?>

```

```

<form action="" method="POST" >
    <input type="text" name="console" />
    <input type="submit" name="validate" />
</form>
<?php
if(isset($_POST["console"]))
{
    $bdd = new PDO("mysql:host=localhost;dbname=test","root","",array(PDO::ATTR_ERRMODE =>
PDO::ERRMODE_EXCEPTION));
    $query = $bdd->prepare('SELECT * FROM jeux_video WHERE console=?');
    $query->execute(array($_POST["console"]));
    while($data = $query->fetch())
        echo $data["nom"]."<br />"; // VERSION 4
}
?>

```

```

<form action="" method="GET">
    <center>
        <label> Console : </label>
        <input type="text" name="console" /><br /><br />
        <label> Nombre de Joueurs: </label>
        <input type="text" name="nbre" /><br /><br />
        <input type="submit" name="submit" value="Valider">
    </center>
</form>
<?php
if(isset($_GET["console"]) && isset($_GET["nbre"]))
{
    $db = new PDO("mysql:host=localhost;dbname=test","root","");
    $query = $db->prepare("SELECT * FROM jeux_video WHERE console=? AND
nbre_joueurs_max=?");
    $query->execute(array($_GET["console"], $_GET["nbre"]));
    while ($data = $query->fetch())
        echo $data["nom"]."-----".$data["console"]."-----".
        ".$data["nbre_joueurs_max"]."<br />"; // VERSION 5
}
?>

```

Autrement :

```

<?php
if(isset($_GET["console"]) && isset($_GET["nbre"]))
{
    $db = new PDO("mysql:host=localhost;dbname=test","root","");
    $query = $db->prepare("SELECT * FROM jeux_video WHERE console=? AND
nbre_joueurs_max=?");
    $query->bindValue(1, $_GET["console"], PDO::PARAM_STR); // console est un STRING
    // 1 cad le premier '?' rencontré
    $query->bindValue(2, $_GET["nbre"], PDO::PARAM_INT); // console est un INT
    // 2 cad le deuxième '?' rencontré
    $query->execute();
    while ($data = $query->fetch())
        echo $data["nom"]."-----".$data["console"]."-----".
        ".$data["nbre_joueurs_max"]."<br />"; // VERSION 1
}
?>

```

```

<?php
if(isset($_GET["console"]) && isset($_GET["nbre"]))
{
    $db = new PDO("mysql:host=localhost;dbname=test", "root", "");
    $query = $db->prepare("SELECT * FROM jeux_video WHERE console=:Hamza AND
nbre_joueurs_max=:Bounasseh");
    $query->bindValue(":Hamza", $_GET["console"], PDO::PARAM_STR);
    $query->bindValue(":Bounasseh", $_GET["nbre"], PDO::PARAM_INT);
    $query->execute();
    while ($data = $query->fetch())
        echo $data["nom"]."-----".$data["console"]."-----".
" ".$data["nbre_joueurs_max"]."<br />"; // VERSION 2
}
?>

```

Ecriture des données

INSERT

```

<form action="" method="GET">
    <center>
        <label> nom : </label>
        <input type="text" name="nom" /><br /><br />
        <label> prix : </label>
        <input type="text" name="prix" /><br /><br />
        <label> nombre de joueurs : </label>
        <input type="text" name="nbre" /><br /><br />
        <input type="submit" name="submit" value="Valider">
    </center>
</form>
<?php
    if(isset($_GET["nom"]) && isset($_GET["prix"]) && isset($_GET["nbre"]))
    {
        $db = new PDO("mysql:host=localhost;dbname=test", "root", "");
        $query = $db->prepare("INSERT INTO `jeux_video`(`nom`, `possesseur`,
`console`, `prix`, `nbre_joueurs_max`, `commentaires`) VALUES
(?, 'x', ?, ?, 'x')");
        // Puisque nos champs dans la BDD sont de valeurs NOT NULL, alors on est obligé de
        // les remplir tous ; c'est pourquoi le 'x' , L'ID s'incrémente automatiquement !
        $query->execute(array($_GET["nom"], $_GET["prix"], $_GET["nbre"]));
    }
?>

```

UPDATE

```

<form action="" method="GET">
    <center>
        <label> id : </label>
        <input type="text" name="id" /><br /><br />
        <label> nom : </label>
        <input type="text" name="nom" /><br /><br />
        <label> prix : </label>
        <input type="text" name="prix" /><br /><br />
        <input type="submit" name="submit" value="Valider">
    </center>
</form>
<?php
    if(isset($_GET["id"]) && isset($_GET["nom"]) && isset($_GET["prix"]))
    {
        $db = new PDO("mysql:host=localhost;dbname=test", "root", "");
        $query = $db->prepare("UPDATE `jeux_video` SET `nom`=? , `prix`=? WHERE
id=?");
        $query->execute(array($_GET["nom"], $_GET["prix"], $_GET["id"]));
    }
?>

```

DELETE

```
<form action="" method="GET">
    <center>
        <label> id : </label>
        <input type="text" name="id" /><br /><br />
        <input type="submit" name="submit" value="Valider">
    </center>
</form>
<?php
    if(isset($_GET["id"]))
    {
        $db = new PDO("mysql:host=localhost;dbname=test", "root", "");
        $query = $db->prepare("DELETE FROM `jeux_video` WHERE id = ?");
        $query->execute(array($_GET["id"]));
    }
?>
```

Ou bien sur :

```
<form action="" method="GET">
    <center>
        <label> id : </label>
        <input type="text" name="id" /><br /><br />
        <input type="submit" name="submit" value="Valider">
    </center>
</form>
<?php
    if(isset($_GET["id"]))
    {
        $db = new PDO("mysql:host=localhost;dbname=test", "root", "");
        $query = $db->prepare("DELETE FROM `jeux_video` WHERE id = ?");
        $query->bindValue(1, $_GET["id"], PDO::PARAM_INT);
        $query->execute();
    }
?>
```