**Requêtes SQL – MachineACafe**

Véronique ROUAULT

|  |  |  |
| --- | --- | --- |
| **Demande** | **Code SQL** | **Résultat affiché** |
| Liste des boissons | [SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) nomboisson  FROM boisson; |  |
| Liste des ingrédients en manque (dont la quantité est nulle) | [SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) nomingredients  FROM ingredients  WHERE stockingredients = 0 |  |
| Liste des boissons dont le libellé contient le mot « café » | [SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) nomboisson  FROM boisson  WHERE nomboisson  LIKE "%Café%" | % remplace n’importe quel caractère |
| Liste des boissons dont le libellé contient la lettre « o » | [SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) nomboisson  FROM boisson  WHERE nomboisson  LIKE "%o%" |  |
| Liste des boissons dont le prix est entre 0.50 et 0.70 euros | [SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) nomboisson  FROM boisson  WHERE prixboisson <=70 AND prixboisson >=50 |  |
|  | [SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) nomboisson  FROM boisson  WHERE prixboisson  BETWEEN 50 [AND](file:///P:/Developpement/_transferts/mysql_doc) 70 | *Autre méthode* |
| Liste des ventes d’aujourd’hui classées par n° décroissant | [SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) \*  FROM vente  WHERE datevente = '2018-01-04'  ORDER BY codevente DESC |  |
|  | [SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) boisson\_codeboisson, datevente,heurevente  FROM vente  WHERE datevente = [CURRENT\_DATE](file:///P:/Developpement/_transferts/mysql_doc)()  ORDER BY codevente DESC |  |
| Liste des ingrédients (nom et qte nécessaire) d’une boisson donnée | [SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) nomboisson, nomingredients, qteboisson  FROM boisson\_ingredients  INNER JOIN boisson ON boisson.codeboisson = boisson\_ingredients.boisson\_codeboisson  INNER JOIN ingredients ON ingredients.codeingredients = boisson\_ingredients.ingredients\_codeingredients  WHERE nomboisson LIKE "%Café%" |  |
| Liste des boissons vendues aujourd’hui | [SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) nomboisson, datevente  FROM boisson  JOIN vente  ON boisson.codeboisson = vente.boisson\_codeboisson  WHERE datevente = [CURRENT\_DATE](file:///P:/Developpement/_transferts/mysql_doc) () |  |
| Prix de la dernière boisson vendue | [SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) nomboisson, prixboisson, datevente, heurevente FROM boisson  JOIN vente  ON boisson.codeboisson=vente.boisson\_codeboisson WHERE datevente = ([SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) [MAX](file:///P:/Developpement/_transferts/mysql_doc)(datevente)FROM vente)  ORDER BY heurevente DESC LIMIT 1 |  |
| Prix de la dernière boisson vendue **du jour** | SELECT nomboisson, datevente, heurevente, prixboisson  FROM boisson  JOIN vente ON boisson.codeboisson=vente.boisson\_codeboisson  WHERE datevente=CURRENT\_DATE  ORDER BY heurevente DESC LIMIT 1 |  |
| Nombre de ventes de la boisson « Café Long » **(Additionner les Commandes)** | [SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) nomboisson, [COUNT](file:///P:/Developpement/_transferts/mysql_doc)(\*)  FROM boisson  JOIN vente  ON boisson.codeboisson = vente.boisson\_codeboisson  WHERE nomboisson [LIKE](file:///P:/Developpement/_transferts/mysql_doc) "Café Long" |  |
|  | SELECT nomboisson, COUNT(\*)  FROM boisson,vente  WHERE boisson.codeboisson = vente.boisson\_codeboisson  AND nomboisson = "Café Long" | *Autre méthode*  *que INNER JOIN* |
| Rajouter la boisson  « Café au lait » | INSERT INTO boisson  (codeboisson, nomboisson, prixboisson)  VALUES (cla, 'Café au Lait', 70) |  |
| Rajouter l’ingrédient  « Lait » | [INSERT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/insert.html) INTO ingredients  (codeingredients, nomingredients, stockingredients) [VALUES](file:///P:/Developpement/_transferts/mysql_doc) (006,'Lait',0) |  |
| Modifier le code de l’ingrédient Lait de « 006 » à « lai » | [UPDATE](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/update.html) ingredients  [SET](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/set.html) codeingredients = ‘lai’  WHERE ingredients.codeingredients = 6 |  |
| Rajouter 100 à la quantité en stock de l’ingrédient « Lait » | [UPDATE](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/update.html) ingredients  [SET](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/set.html) stockingredients = stockingredients + 100  WHERE nomingredients = "Sucre" |  |
| Augmenter de 10 centimes d’euros le prix de toutes les boissons | [UPDATE](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/update.html) boisson  [SET](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/set.html) prixboisson = prixboisson + 10 |  |
| Créer une nouvelle vente d’expresso avec 2 sucres | [INSERT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/insert.html) INTO vente(codevente, datevente, heurevente, nbsucres, boisson\_codeboisson)  [VALUES](file:///P:/Developpement/_transferts/mysql_doc) (NULL,[CURRENT\_DATE](file:///P:/Developpement/_transferts/mysql_doc),[CURRENT\_TIME](file:///P:/Developpement/_transferts/mysql_doc),2,"exp") |  |
| Supprimer cette vente | [DELETE](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/delete.html) FROM `vente` WHERE codevente = 11 |  |
| Liste des boissons disponibles (pour lesquelles les ingrédients sont dispo) | [SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) nomboisson, qteboisson, stockingredients  FROM boisson\_ingredients  INNER JOIN boisson ON boisson.codeboisson = boisson\_ingredients.boisson\_codeboisson  INNER JOIN ingredients ON ingredients.codeingredients = boisson\_ingredients.ingredients\_codeingredients  GROUP BY nomboisson  HAVING [MIN](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/group-by-functions.html" \l "function_min)(qteboisson <= stockingredients) = 1 |  |

**GLOSSAIRE**

# SQL WHERE

La commande WHERE dans une requête SQL permet d’extraire les lignes d’une base de données qui respectent une condition. Cela permet d’obtenir uniquement les informations désirées.

## Syntaxe

La commande WHERE s’utilise en complément à une requête utilisant SELECT. La façon la plus simple de l’utiliser est la suivante :

SELECT nom\_colonnes FROM nom\_table WHERE condition

## Opérateurs de comparaisons

Il existe plusieurs opérateurs de comparaisons. La liste ci-jointe présente quelques-uns des opérateurs les plus couramment utilisés.

| Opérateur | Description |
| --- | --- |
| = | Égale |
| <> | Pas égale |
| != | Pas égale |
| > | Supérieur à |
| < | Inférieur à |
| >= | Supérieur ou égale à |
| <= | Inférieur ou égale à |
| IN | Liste de plusieurs valeurs possibles |
| BETWEEN | Valeur comprise dans un intervalle donnée (utile pour les nombres ou dates) |
| LIKE | Recherche en spécifiant le début, milieu ou fin d'un mot. |
| IS NULL | Valeur est nulle |
| IS NOT NULL | Valeur n'est pas nulle |
| LIMIT 5, 10 | Permet de ne sélectionner qu'une partie des résultats (ici de 5 à 10) |
| INNER JOIN | jointure interne pour retourner les enregistrements quand la condition est vrai dans les 2 tables. C’est l’une des jointures les plus communes. |
| INSERT INTO |  |
| GROUP BY | La commande GROUP BY est utilisée en SQL pour grouper plusieurs résultats et utiliser une fonction de totaux sur un groupe de résultat. Sur une table qui contient toutes les ventes d’un magasin, il est par exemple possible de liste regrouper les ventes par clients identiques et d’obtenir le coût total des achats pour chaque client. |

|  |  |  |
| --- | --- | --- |
| HAVING | Similaire à WHERE à la seule différence que HAVING permet de filtrer en utilisant des fonctions telles que SUM(), COUNT(), AVG(), MIN() ou MAX(). |  |
| [MIN](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/group-by-functions.html" \l "function_min) | Permet de retourner la plus petite valeur d’une colonne sélectionnée. Cette fonction s’applique aussi bien à des données numériques qu’à des données alphanumériques. |  |
| SELECT CURRENT\_DATE | Récupère la date courante |  |
| SELECT CURRENT\_DATE AS 'DATE', CURRENT\_TIME AS 'TIME' | Récupère la date et l’heure courante en renommant les entêtes de colonnes « Date » et « Time » |  |

INSERT INTO `vente` (`codevente`, `datevente`, `heurevente`, `nbsucres`, `boisson\_codeboisson`)

VALUES (NULL, '2018-01-05', '08:34:49', '0', 'exp');

Ajoute une vente avec des commandes SQL

INSERT INTO `boisson\_ingredients` (`boisson\_codeboisson`, `ingredients\_codeingredients`, `qteboisson`) VALUES ('cla', 'caf', '1'), ('cla', 'lai', '1'), ('cla', 'eau', '2');

Ajoute la recette du café au lait

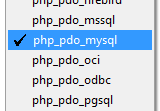
### **Se connecter à la base de données en PHP avec PDO**

**L'extension PDO**: c'est un outil complet qui permet d'accéder à n'importe quel type de base de données. On peut donc l'utiliser pour se connecter aussi bien à MySQL que PostgreSQL ou Oracle.

#### **Se connecter à MySQL avec PDO :**

Vérifiez que l'extension PDO est activée

Normalement, PDO est activé par défaut. Pour le vérifier (voir la figure suivante), faites un clic gauche sur l'icône de WAMP dans la barre des tâches, puis allez dans le menuPHP / Extensions PHPet vérifiez quephp\_pdo\_mysqlest bien coché.



Pour nous connecter à MySQL. Nous allons avoir besoin de quatre renseignements :

* le nom de l'hôte : l'adresse de l'ordinateur où MySQL est installé (comme une adresse IP). Le plus souvent, MySQL est installé sur le même ordinateur que PHP : dans ce cas, mettre la valeurlocalhost(cela signifie « sur le même ordinateur »). Néanmoins, il est possible que l’hébergeur web indique une autre valeur à renseigner (qui ressemblerait à : sql.hebergeur.com). Dans ce cas, il faud modifier cette valeur lorsque l’on envoie le site sur le Web .
* la base : le nom de la base de données à laquelle on veut se connecter… machineacafe.
* le login : il permet l’identification. Le plus souvent, chez un hébergeur gratuit, c'est le même login que celui utilisé pour le FTP ;
* le mot de passe : le mot de passe est souvent le même que celui utilisé pour accéder au FTP.

<?php

// Sous WAMP (Windows)

$bdd = new PDO('mysql:host=localhost;dbname=test;charset=utf8', 'root', '');

?>

Cette ligne de code crée un objet $bdd qui représente la connexion à la base de données avec les paramètres dans l’ordre :

* le nom d'hôte (localhost) ;
* la base de données (machineacafe) ;
* le login (vero) ;
* le mot de passe (s’il n'y a pas de mot de passe, mettre une chaîne vide).

Exemple d’un site en ligne :

<?php

$bdd = new PDO('mysql:host=sql.hebergeur.com;dbname=mabase;charset=utf8', 'pierre.durand', 's3cr3t');

?>

Le premier paramètre (qui commence par mysql) s'appelle le DSN : Data Source Name.

C'est généralement le seul qui change en fonction du type de base de données auquel on se connecte.

Exemple avec fonction de test :

<?php

try

{

$bdd = new PDO('mysql:host=localhost;dbname=machineacafe;charset=utf8', 'vero', 'MotDePasse38');

}

catch (Exception $e)

{

die('Erreur : '. $e -> getMessage());

}

?>

Le paramètre qui commence par mysql s'appelle le DSN : Data Source Name.

C'est généralement le seul qui change en fonction du type de base de données auquel on se connecte.

Le bloc try permet de tester s'il y a une erreur dans les données d’accès à la base de données.

Si il y a une erreur, il rentre dans le bloc catch et fait ce qu'on lui demande (ici, on arrête l'exécution de la page en affichant un message décrivant l'erreur).

Si au contraire tout se passe bien, PHP poursuit l'exécution du code et ne lit pas ce qu'il y a dans le bloc catch.

**Récupérer les données :**

<?php

$reponse = $bdd -> query('Tapez votre requête SQL ici');

?>

*quéry* en anglais signifie « requête »

On récupère ce que la base de données nous a renvoyé dans un autre objet que l'on a appelé ici

$reponse.

<?php

$donnees = $reponse->fetch();

?>

fetch en anglais signifie « va chercher ».

$donnees est un array qui contient champ par champ les valeurs de la première entrée.

$reponse contient toute la réponse de MySQL en vrac, sous forme d'objet.  
$donnees est un array renvoyé par le fetch().

Chaque fois qu'on fait une boucle, fetch va chercher dans $reponse l'entrée suivante et organise les champs dans l'array $donnees.

<?php

while ($donnees = $reponse→fetch())

?>

Cette ligne fait deux choses à la fois :

* elle récupère une nouvelle entrée et place son contenu dans $donnees ;
* elle vérifie si $donnees vaut vrai ou faux.

Le fetch renvoie faux (false) dans $donnees lorsqu'il est arrivé à la fin des données, c'est-à-dire que toutes les entrées ont été passées en revue.

Dans ce cas, la condition du while vaut faux et la boucle s'arrête.

<?php

$reponse->closeCursor();

?>

Termine le traitement de la requête. A placer à la fin de chaque requête.