

Interroger une table MySQL

Maintenant que nous nous sommes connectés à la base de données (exercice 14), nous allons interroger une table pour en extraire les résultats puis les ranger dans un ordre précis. Créons d'abord une table comme ci-dessous.

```
Requête SQL : CREATE TABLE famille_tbl ( id int(11) NOT NULL auto_increment, nom varchar(255) NOT NULL, prenom varchar(255) NOT NULL, statut varchar(255) NOT NULL, date date DEFAULT '0000-00-00' NOT NULL, PRIMARY KEY (id), KEY id (id), UNIQUE id_2 (id) );
```

```
INSERT INTO famille_tbl VALUES( '', 'Dupond', 'Grégoire', 'Grand-père', '1932-05-17');
INSERT INTO famille_tbl VALUES( '', 'Dupond', 'Germaine', 'Grand-mère', '1939-02-15');
INSERT INTO famille_tbl VALUES( '', 'Dupond', 'Gérard', 'Père', '1959-12-22');
INSERT INTO famille_tbl VALUES( '', 'Dupond', 'Marie', 'Mère', '1961-03-02');
INSERT INTO famille_tbl VALUES( '', 'Dupond', 'Julien', 'Fils', '1985-05-17');
INSERT INTO famille_tbl VALUES( '', 'Dupond', 'Manon', 'Fille', '1990-11-29');
```

Structure de la table (PhpMyAdmin) :

Champ	Type	Null	Defaut	Extra
id	int(11)	Non	0	auto_increment
nom	varchar(255)	Non		
prenom	varchar(255)	Non		
statut	tinyint(255)	Non	0	
date	date	Non	0000-00-00	

Contenu de la table "famille_tbl"

id	nom	prenom	statut	date
1	Dupond	Grégoire	Grand-père	1932-05-17
2	Dupond	Germaine	Grand-mère	1939-02-15
3	Dupond	Gérard	Père	1959-12-22
4	Dupond	Marie	Mère	1961-03-02
5	Dupond	Julien	Fils	1985-05-17
6	Dupond	Manon	Fille	1990-11-29

Affichage des résultats tels qu'ils sont dans la table sans condition.

Code PHP

```
<?php
// on se connecte à MySQL
$db = mysql_connect('localhost', 'login', 'password');

// on sélectionne la base
mysql_select_db('nom_de_la_base',$db);

// on crée la requête SQL
$sql = 'SELECT nom,prenom,statut,date FROM famille_tbl';

// on envoie la requête
$req = mysql_query($sql) or die('Erreur SQL !<br>'.$sql.'<br>'.mysql_error());

// on fait une boucle qui va faire un tour pour chaque enregistrement
while($data = mysql_fetch_assoc($req))
{
    // on affiche les informations de l'enregistrement en cours
```

<pre> echo ''.\$data['nom'].' '.\$data['prenom'].' ('.\$data['statut'].')'; echo ' <i>date de naissance : '.\$data['date'].'</i>
'; } // on ferme la connexion à mysql mysql_close(); ?> </pre>
Donne à l'écran
<p>Dupond Grégoire (Grand-père), <i>date de naissance : 1932-05-17</i></p> <p>Dupond Germaine (Grand-mère), <i>date de naissance : 1939-02-15</i></p> <p>Dupond Gérard (Père), <i>date de naissance : 1959-12-22</i></p> <p>Dupond Marie (Mère), <i>date de naissance : 1961-03-02</i></p> <p>Dupond Julien (Fils), <i>date de naissance : 1985-05-17</i></p> <p>Dupond Manon (Fille), <i>date de naissance : 1990-11-29</i></p>

Explication :

Voici donc notre première boucle sur une table, champagne ! :). Plus sérieusement, vous vous apercevez que les résultats qui s'affichent sont exactement dans le même ordre que la table et pour cause, nous n'avons pas spécifié de condition dans notre requête (2), donc dans ce cas, la requête scanne la table de haut en bas.

Remarquez que la fonction `mysql_fetch_assoc()` renvoie un tableau dont les clés sont les noms des champs sélectionnés. On a aussi rajouté après `mysql_query()` ceci: `or die('Erreur SQL !
'.$sql.'
mysql_error());`. Cela veut dire qu'en cas d'erreur dans la requete vers mysql, ce qui arrive fréquemment, php va afficher un message indiquant l'erreur renvoyée par mysql (grace à `mysql_error()`) ce qui fournit une aide précieuse pour comprendre le problème.

Vous notez également que les dates de naissance sont au format US, ceci est normal puisque nous avons défini un type DATE dans notre table, nous verrons plus bas comment convertir les dates US au format FR.

Nous allons maintenant faire plusieurs tests en ne changeant uniquement que la requête SQL (2). **Le reste du code ne change pas.**

Affichage des résultats par ordre alphabétique de prénom.

Le code PHP de la requête
<pre> <?php // Gardez le code ci-dessus, changez juste la requête SQL ! \$sql = 'SELECT nom,prenom,statut,date FROM famille_tbl ORDER BY prenom'; // L'opérateur ORDER BY permet de classer soit alphabétiquement // soit numériquement suivant le type du champ. // Si l'on souhaite classer en décroissant (ex. de Z à A), nous // y ajouterons DESC soit : ORDER BY prenom DESC ?> </pre>
Donne à l'écran
<p>Dupond Gérard (Père), <i>date de naissance : 1959-12-22</i></p> <p>Dupond Germaine (Grand-mère), <i>date de naissance : 1939-02-15</i></p> <p>Dupond Grégoire (Grand-père), <i>date de naissance : 1932-05-17</i></p>

Dupond Julien (Fils), *date de naissance : 1985-05-17*
Dupond Manon (Fille), *date de naissance : 1990-11-29*
Dupond Marie (Mère), *date de naissance : 1961-03-02*

Affichage des résultats par comparaison de date.

Le code PHP de la requête
<pre><?php // Gardez le code ci-dessus, changez juste la requête ! \$sql = 'SELECT nom,prenom,statut FROM famille_tbl WHERE date<1960-01-01'; // L'avantage d'avoir un type DATE dans notre base de données, c'est que // nous pouvons comparer des dates dans la requête SQL. // Ici nous ne souhaitons afficher que les membres de la famille qui sont // nés avant le 1er janvier 1960, soit : WHERE date<1960-01-01 ?></pre>
Donne à l'écran
<p>Dupond Grégoire (Grand-père), <i>date de naissance : 1932-05-17</i> Dupond Germaine (Grand-mère), <i>date de naissance : 1939-02-15</i> Dupond Gérard (Père), <i>date de naissance : 1959-12-22</i></p>

Affichage des résultats avec le commande LIKE.

La commande LIKE en SQL permet de fouiller le contenu de chaque champ. Je m'explique, je recherche tous les enregistrements dont le champ **"prenom"** commence par la lettre **"G"**, voyons la syntaxe ci-dessous.

Le code PHP de la requête
<pre><?php // Gardez le code ci-dessus, changez juste la requête ! \$sql = "SELECT nom,prenom,statut,date FROM famille_tbl WHERE prenom LIKE 'G%'"; // Le signe pourcentage "%" placé après le G, indique que la lettre G peut // être suivie, mais pas précédée, d'autres caractères ! // Notez aussi que LIKE n'est pas sensible à la casse, cela veut dire que // la requête cherchera aussi bien des G majuscules que minuscules. ?></pre>
Donne à l'écran
<p>Dupond Grégoire (Grand-père), <i>date de naissance : 1932-05-17</i> Dupond Germaine (Grand-mère), <i>date de naissance : 1939-02-15</i> Dupond Gérard (Père), <i>date de naissance : 1959-12-22</i></p>

Maintenant voyons la même chose mais cette fois nous allons chercher la syllabe **"ma"** dans le champ **"prenom"**, qu'elle soit placée au début ou au milieu d'autres caractères.

Le code PHP de la requête
<pre><?php // Gardez le code ci-dessus, changez juste la requête ! \$sql = "SELECT * FROM famille_tbl WHERE prenom LIKE '%MA%'";</pre>

```
// Le signe pourcentage "%" placé avant et après MA indique que la syllabe
// peut-être précédée ou suivie de caractères.
// Une fois de plus notez que LIKE n'est pas sensible à la casse, la requête
// cherchera aussi bien des MA majuscules que des ma en minuscules.

?>
```

Donne à l'écran

Dupond Marie (Mère), *date de naissance : 1961-03-02*

Dupond Manon (Fille), *date de naissance : 1990-11-29*

Bien sûr, il est possible de mettre plusieurs conditions dans la même requête, exemple :

```
• SELECT * FROM famille_tbl WHERE prenom LIKE '%MA%' AND date<'1960-01-01' ORDER BY prenom
```

Pour terminer voici comment vous allez pouvoir convertir la date du format US au format FR, une fois que vous avez récupéré l'information depuis la table. Nous vous donnons deux façons de faire, la deuxième est bien meilleure !

Le code PHP (déconseillé pour ce cas précis)

```
<?php
// on se connecte à MySQL
$db = mysql_connect('localhost', 'login', 'password');

// on sélectionne la base
mysql_select_db('nom_de_la_base',$db);

// on crée la requête SQL
$sql = 'SELECT nom,prenom,statut,date FROM famille_tbl';

// on envoie la requête
$req = mysql_query($sql) or die('Erreur SQL !<br>'.$sql.'<br>'.mysql_error());

// on fait une boucle qui va faire un tour pour chaque enregistrement
while($data = mysql_fetch_array($req))
{
    $a = substr($data['date'], 0, 4);    // conversion
    $m = substr($data['date'], 5, 2);    // de la date
    $j = substr($data['date'], 8, 2);    // au format
    $date = $j.'-'. $m.'-'. $a;          // Français

    // on affiche les informations de l'enregistrement en cours
    echo '<b>'.$data['nom'].' '.$data['prenom'].'</b> ('.$data['statut'].')';
    echo ' <i>date de naissance : '.$date.'</i><br>';
}

// on ferme la connexion à mysql
mysql_close();
?>
```

Le code PHP (bien meilleur, utilisant les ressources de MySQL)

```
<?php
// on se connecte à MySQL
$db = mysql_connect('localhost', 'login', 'password');

// on sélectionne la base
mysql_select_db('nom_de_la_base',$db);
```

```
// on crée la requete SQL
$sql = 'SELECT nom,prenom,statut,DATE_FORMAT(date, '%d-%m-%Y') as datefr FROM
famille_tbl';

// on envoie la requête
$req = mysql_query($sql) or die('Erreur SQL !<br>'.$sql.'<br>'.mysql_error());

// on fait une boucle qui va faire un tour pour chaque enregistrement
while($data = mysql_fetch_array($req))
{
    // on affiche les informations de l'enregistrement en cours
    echo '<b>'.$data['nom'].' ' . $data['prenom'].'</b> ('.$data['statut'].')';
    echo ' <i>date de naissance : '.$data['datefr'].'</i><br>';
}

// on ferme la connexion à mysql
mysql_close();
?>
```

Donne à l'écran, dans les deux cas

Dupond Grégoire (Grand-père), *date de naissance : 17-05-1932*
Dupond Germaine (Grand-mère), *date de naissance : 15-02-1939*
Dupond Gérard (Père), *date de naissance : 22-12-1959*
Dupond Marie (Mère), *date de naissance : 02-03-1961*
Dupond Julien (Fils), *date de naissance : 17-05-1985*
Dupond Manon (Fille), *date de naissance : 29-11-1990*