SQL
et
introduction à PDO

Sommaire

WEBFORCE3
★★★<HELLO WORLD>

- Syntaxe SQL.
- SELECT, INSERT, UPDATE, DELETE.
- La clause WHERE.
- Les opérateurs 'AND' et 'OR'.
- Limites.
- Trier les résultats.
- Les erreurs SQL.
- Les classes PDO.
- Connexion PDO.
- Exécution de requêtes avec PDO.

Syntaxe SQL

WEBFORCE3
★★★<HELLO WORLD>

- SQL (Structured Query Langage) est le langage utilisé par les bases de données. Il permet :
  - L'ajout d'informations dans une table.
  - La modification d'informations dans une table.
  - La suppression d'informations dans une table.
  - La lecture d'informations dans une ou plusieurs tables.
  - La modification des tables et des champs.
  - La création et la suppression de bases de données.
  - La gestion des droits.

Syntaxe SQL

WEBFORCE3 ★★★<HELLO WORLD>

- La commande `INSERT INTO` permet d'ajouter des informations dans une table:

```
INSERT INTO bdd.table(champs1, champs2)
values(val1, val2)
```

Exemple:

```
INSERT INTO webforce3.eleve(nom, prenom)
values('DURAND', 'Paul');
```

Syntaxe SQL

- La commande `UPDATE` permet de modifier des informations dans une table. Cette commande doit toujours être accompagnée d'une clause "WHERE" afin d'éviter de modifier tout le fichier.

```
UPDATE bdd.table set champs1=val1, champs2 = val2 where …

Exemple:
UPDATE webforce3.eleve set nom='DUPOND' , prenom = 'Paul'
where idEleve = 8
```

Syntaxe SQL

WEBFORCE3 ★★★<HELLO WORLD>

- La commande `DELETE FROM` permet de supprimer des informations dans une table. Cette commande doit toujours être accompagnée d'une clause "WHERE" afin d'éviter de vider tout le fichier.

```
DELETE FROM bdd.table where …

Exemple:
DELETE FROM webforce3.eleve where idEleve = 8
```

Exercice pratique:

Ajouter, modifier, supprimer des enregistrements dans les tables de la base webforce3

Syntaxe SQL

- La commande `SELECT FROM` permet de lire une ou plusieurs informations dans une table. Cette commande peut être accompagnée de clauses complémentaires permettant de choisir ce que l'on veut récupérer, de trier les résultats, de compléter certaines informations, …

```
SELECT * FROM bdd.table

Exemple:
SELECT nom, prenom FROM webforce3.eleve where idEleve = 7
```

Syntaxe SQL

WEBFORCE3 ★★★<HELLO WORLD>

- La clause `WHERE` permet de définir la quantité d'informations à retourner.

```
SELECT FROM bdd.table WHERE …
```

Exemple:

- `SELECT * FROM webforce3.eleve where idEleve = 7`
- `SELECT a, b, c FROM bdd.fichier where champs < 10`
- `SELECT a,b FROM bdd.fichier where champs between 0 AND 10`
- `SELECT a,b from bdd.fichier where champs IN(1,2,3)`

Syntaxe SQL

WEBFORCE3 ★★★<HELLO WORLD>

- La clause `WHERE` permet de définir la quantité d'informations à retourner.

```
% est un joker qui peut remplacer tous les autres caractères.
_  est un joker qui peut remplacer un unique caractère.
Exemple:
• SELECT * FROM bdd.fichier where champs like %a
• SELECT a, b, c FROM bdd.fichier where champs like a%
• SELECT a,b FROM bdd.fichier where champs like %a%
• SELECT a,b from bdd.fichier where champs PA%IS
• SELECT a,b from bdd.fichier where champs like PA_IS
```

Syntaxe SQL

WEBFORCE3 ★★★<HELLO WORLD>

- On peut faire des conditions complexes avec les opérateurs `AND` et `OR`.

Exemple:
Si le champs 'c' est au format DATETIME

- `DELETE FROM bdd.fichier where (nom='DURAND' AND prenom='Paul') OR idEleve=8`

Syntaxe SQL

WEBFORCE3 ★★★<HELLO WORLD>

- La fonction `NOW()` retourne la date et l'heure, pour un champs date ou datetime

Exemple:
Si le champs 'c' est au format DATETIME

- `INSERT INTO bdd.fichier(a,b,c) values('valA', 'valB', NOW())`

Syntaxe SQL

WEBFORCE3 ★★★<HELLO WORLD>

- La fonction `LIMIT debut, fin` permet de modifier le nombre d'enregistrements renvoyé, et la position de départ

Exemple:

```
SELECT a,b from bdd.fichier LIMIT 10,10
```

Syntaxe SQL

WEBFORCE3 ★★★<HELLO WORLD>

- La fonction `ORDER BY` permet de modifier l'ordre de tri du jeu d'enregistrements renvoyé, selon le champs de son choix. Deux options possibles:
  - `ASC`
  - `DESC`

```
Exemple:
SELECT a,b from bdd.fichier where a=1 order by a ASC
```

Exercice pratique:

Lire des enregistrements dans la table INSEE en utilisant les différentes possibilités de la clause WHERE

Les erreurs

WEBFORCE3
★★★<HELLO WORLD>

- En cas d'erreur MySQL retourne un message d'erreur plus ou moins compréhensible.
- En cas d'erreur de requête il nous retourne l'endroit ou il lui semble avoir l'erreur, avec parfois quelques subtilités pour la trouver.

PDO

- L'extension PDO (PHP Data Object) fournit une interface d'abstraction à l'accès aux données, ce qui signifie que l'on va utilise les mêmes fonctions pour exécuter des requêtes quelle que soit la base de données utilisée.
- PDO utilise un driver spécifique à chaque base de données pour travailler.

PDO

PDO utilise deux classes:

- PDO représente une connexion avec une base de données.
- PDOStatement représente une requête préparée et, une fois exécutée, le jeu de résultats associé.

PDO: Connexion

WEBFORCE
★★★<HELLO WORLD>

```
<?php
// connexion avec PDO
$dbh = new PDO('mysql:host=localhost;dbname=test',
                $user, $pass);
print_r($dbh);
…
…
// fermeture de connexion
$dbh = null;

?>
```

PDO: Exécuter une requête

PDO utilise utilise deux fonctions pour exécuter des requêtes:

- PDO::exec() exécute une requête SQL et retourne le nombre de lignes affectées par la requête. Exec() ne retourne pas de résultat pour une requête SELECT!
- PDO::query() exécute une requête SQL et retourne un jeu de résultats en tant qu'objet PDOStatement.

PDO: Exécuter une requête

WEBFORCE3 ★★★<HELLO WORLD>

```
<?php
//connexion MySQL avec PDO
$dbh = new PDO('mysql:host=localhost;dbname=test',
                $user, $pass);

/* Effacement de toutes les lignes de la table FRUIT */
$count = $dbh->exec("DELETE FROM fruit WHERE couleur = 'rouge'");

/* Retourne le nombre de lignes effacées */
echo "Retourne le nombre de lignes effacées :";
echo "Effacement de $count lignes.";


?>
```

PDO: Exécuter une requête

WEBFORCE3
★★★<HELLO WORLD>

```
<?php

function getFruit($conn)
{
  $sql =  'SELECT name, color, calories FROM fruit ORDER BY name';
  foreach  ($conn->query($sql) as $row)
  {
    print $row['name'] . "\t";
    print  $row['color'] . "\t";
    print $row['calories'] . "\n";
  }
}

?>
```

Exercices pratiques