

SQL, et introduction à PDO

Vous avez obtenu la note de 0 sur 13

Le 07/03/2018 à 09:02 vous aviez obtenu la note de 9 sur 13

L'acronyme CRUD correspond à Create / Read / Update / Delete, vrai ou faux ?

Explication:

L'acronyme informatique anglais CRUD (pour Create, Read, Update, Delete) désigne les quatre opérations de base pour le stockage et la manipulation d'informations dans une base de données.

- **Create** : Insérer des nouveaux enregistrements (écriture)
- **Read** : Récupérer des enregistrements (lecture)
- **Update** : Mettre à jour des enregistrements (écriture)
- **Delete** : Supprimer des enregistrements (écriture)

Vrai

Faux

Quelle clause SQL permet de trier/ordonner les résultats d'une requête ?

Explication:

Réservée aux requêtes SELECT, la clause ORDER BY permet de trier la liste des résultats de la requête, en ordre croissant (ASC) ou décroissant (DESC).

Exemple :

```
SELECT * FROM movies ORDER BY release_date DESC; // Remonte tous les films triés par date de sortie décroissante
```

```
SELECT * FROM movies ORDER BY RAND(); // Remonte tous les films dans un ordre pseudo aléatoire
```

```
SELECT * FROM movies ORDER BY lastname ASC, firstname ASC; // Remonte tous les films triés par ordre alphabétique sur le nom de famille, et à nom égal par ordre alphabétique croissant sur le prénom
```

SORT BY

SORT

ORDER BY

ORDER

Quelle fonction SQL permet d'obtenir la somme de toutes les valeurs d'une colonne ?

Explication:

La fonction SUM permet d'additionner l'ensemble des valeurs d'une colonne renvoyées par une requête.

Exemple :

```
SELECT SUM(scores) FROM games; -- Calcul et renvoie la somme de la colonne  
scores pour toutes les lignes  
SELECT SUM(price) FROM products; -- Même chose avec des prix de produits
```

COUNT

AVG

VSUM

MAX

Quel(s) paramètre(s) faut-il modifier si une erreur du type `[1045] Access denied for user 'root'@'localhost' (using password: YES)` survient ?

Plusieurs réponses sont possibles.

Explication:

L'erreur `[1045] Access denied for user 'root'@'localhost' (using password: YES)` survient lorsqu'un utilisateur non autorisé tente d'accéder à un serveur de base de données.

Il peut s'agir d'une erreur dans le nom d'utilisateur défini, d'une erreur de mot de passe, ou d'une erreur de configuration des droits de lecture/écriture.

Le host dans le DSN

VL'identifiant de connexion

VLe mot de passe de connexion

Le nom de la base de données

VLa liste des utilisateurs autorisés

Quels sont les erreurs les plus fréquentes rencontrées avec les base de données MySQL ?

Plusieurs réponses sont possibles.

Explication:

Les erreurs les plus fréquentes surviennent lors de la connexion à la base de données, ou durant l'exécution des requêtes SQL.

Un oubli ou une erreur dans la syntaxe de la requête en sont généralement la cause, parfois c'est plus sournois comme lors de l'utilisation involontaire d'un mot clé réservé par MySQL pour les noms de table ou de colonne.

VLes erreurs de connexion

VLes erreurs de syntaxe dans les requêtes

VL'utilisation de mots clés réservés par MySQL dans les noms de table ou de colonnes

En PHP que signifie l'acronyme PDO ?

Explication:

L'acronyme PDO correspond à PHP Data Objects qui désigne l'interface permettant d'interroger différents types de base de données.

Program Development Operator

VPHP Data Objects

Programs as Data Objects

Portable Distributed Object

Quelle méthode renvoie le dernier identifiant inséré suite à une requête INSERT, sur une table avec une clé primaire en AUTO_INCREMENT ?

Explication:

La méthode lastInsertId() de la classe PDO retourne le dernier identifiant auto incrémenté suite à une requête INSERT.

Exemple :

```
$db = new PDO('mysql:host=localhost;dbname=ma_bdd', 'login', 'password');

$query = $db->prepare('INSERT INTO ma_table SET col = :value');
$query->bindValue(':value', 'test', PDO::PARAM_STR);
$query->execute();
$insert_id = $db->lastInsertId();
```

Consulter la documentation PHP sur la classe PDO : <http://php.net/manual/fr/class.pdo.php>

La méthode lastInsertId() de la classe PDO

La méthode lastInsertId() de la classe PDOStatement

La méthode rowCount() de la classe PDOStatement

La méthode execute() de la classe PDOStatement

Quelle(s) méthode(s) de la classe PDOStatement peut-on utiliser récupérer les résultats d'une requête SQL ?

Plusieurs réponses sont possibles.

Explication:

Les méthodes `fetch()` et `fetchAll()` de la classe `PDOStatement` permettent de récupérer les résultats d'une requête `SELECT` préalablement exécutée.

Exemple :

```
// Sans prepare
$query = $db->query('SELECT * FROM ma_table');
$result = $query->fetch(); // Renvoie une seule ligne de résultat
$results = $query->fetchAll(); // Renvoie toutes les lignes de résultat

// Avec prepare
$query = $db->prepare('SELECT * FROM ma_table');
$query->execute();
$result = $query->fetch(); // Renvoie une seule ligne de résultat
$results = $query->fetchAll(); // Renvoie toutes les lignes de résultat
```

Consulter la documentation PHP sur la classe `PDOStatement` :

<http://php.net/manual/fr/class.pdostatement.php>

prepare

fetch

execute

VfetchAll

La classe PDO permet d'effectuer une connexion entre PHP et une base de données, vrai ou faux ?

Explication:

La classe PDO représente la connexion entre PHP et la base de données. Elle doit être instanciée avec un DSN valide pour être utilisée (c.f. "Connexion à la base de données").

Exemple :

```
$db = new PDO('mysql:host=localhost;dbname=ma_bdd, 'login', 'password');
```

Consulter la documentation PHP sur la classe PDO : <http://php.net/manual/fr/class.pdo.php>

VVrai

Faux

Quelles sont les informations obligatoires pour se connecter à une base de données avec PDO ?

Plusieurs réponses sont possibles.

Explication:

La connexion à la base de données avec PDO s'effectue en déclarant une nouvelle instance de la classe PDO, et en lui passant les paramètres suivants :

- Le DSN (composé du type et du chemin d'accès à la base host + db_name)
- Les identifiants de connexion
- Des options supplémentaires facultatives telles que l'encodage des caractères

Exemple :

```
$connexion = new PDO('mysql:host=localhost;dbname=ma_bdd', 'login',  
'password');
```

VHost (domaine)

VDSN (Data Source Name)

Encodage des caractères

Videntifiants de connexion

Quelle est le 3ème argument de la méthode bindValue() de la classe PDOStatement ?

Explication:

La méthode `bindValue()` permet de transmettre la valeur des paramètres d'une requête préparée.
(c.f. méthode `prepare()` de "La classe PDO").

Elle prend 3 arguments :

- Le nom du paramètre sous la forme `:nom`;
- La valeur du paramètre qui peut être une chaîne ou un nombre;
- Le type du paramètre (PDO::PARAM_STR pour les chaînes, PDO::PARAM_INT pour les nombres);

Exemple :

```
$query = $db->prepare('SELECT * FROM ma_table WHERE id = :id');  
$query->bindValue(':id', 42, PDO::PARAM_INT);  
$query->execute();  
$result = $query->fetch();
```

Le nom du paramètre

La valeur du paramètre

Le type du paramètre (chaîne ou nombre)

Quelle méthode permet de transmettre une valeur à un paramètre avec une requête préparée ?

Explication:

La méthode `bindValue()` permet de transmettre la valeur des paramètres d'une requête préparée.

Consulter la fiche "Les classes PDO et PDOStatement" pour plus d'information sur la méthode `bindValue`.

`VbindValue`

`bindParam`

`fetch`

`prepare`

Quelle méthode permet de connaître le nombre de lignes affectées par une requête `UPDATE` ou `DELETE` ?

Explication:

La méthode `rowCount()` de la classe `PDOStatement` renvoie le nombre de lignes affectées par une requête.

Dans le cas d'une requête `SELECT`, il s'agit du nombre de lignes renvoyées par la requête. Dans le cas d'une requête `UPDATE` ou `DELETE`, il s'agit du nombre de lignes modifiées ou supprimées.

affectedRows()

nbRows()

VrowCount()

fetchAll();