PHP – Requêtes préparées

PHP – Requêtes préparées

- Se connecter à une base de données
- Sécuriser l’accès aux données

PHP – Requêtes préparées

Introduction

- Presque toutes les bases de données gèrent les requêtes préparées
- Les requêtes préparées doivent être analysées (ou préparées) une fois
  - Ensuite elles peuvent être exécutées plusieurs fois avec des paramètres identiques ou différents
- La base de données analysera, compilera puis optimisera la requête

PHP – Requêtes préparées

Introduction

<u>Pourquoi les utiliser :</u>

- Elles permettent de répéter le cycle : analyse, compilation et optimisation
- Elle permettent de se protéger contre les injections SQL

Cela signifie que les instructions préparées utilisent moins de ressources, s'exécutent plus rapidement et sont plus sécurisées !

PHP – Requêtes préparées

Paramétrer les requêtes préparées

- La première chose à faire est de définir les paramètres dans les instructions préparées !
  - Paramètres commandés (ou indéxés)

```
$sql = "INSERT INTO users (login, password) VALUES (?, ?)";
```

  - Paramètres nommés

```
$sql = "INSERT INTO users (login, password) VALUES (:login, :password)";
```

PHP – Requêtes préparées

Paramétrer les requêtes préparées – Note importante

Attention !!

- Vous ne pouvez pas mélanger les paramètres commandés et nommés dans une requête préparée !
- Mais il est possible d'utiliser différentes types de paramètres dans des requêtes préparées distinctes

PHP – Requêtes préparées

Paramétrer les requêtes préparées – Méthodes fournies

- Une fois que la requête est prête, on doit la fournir au SGBD
  - Pour cela on utilise la méthode

    PDOStatement prepare(string $stmt)

```php
$sql = "INSERT INTO users (login, password) VALUES (?, ?)";

$statement = $pdo->prepare($sql);
```

PHP – Requêtes préparées

Paramétrer les requêtes préparées – Méthodes fournies

- Une fois que la requête est préparée, on peut l'utiliser
- Pour cela on définit les paramètres de la requête :
  - Soit directement dans la fonction execute ()

```
$sql = "INSERT INTO author (firstname, lastname) VALUES ( :firstname, :lastname )";

$statement = $pdo->prepare($sql);

$firstname = 'John';
$lastname = 'Doe';

$statement->execute(  array(':firstname' => $firstname,
                    ':lastname' => $lastname
          )
);
```

PHP – Requêtes préparées

Paramétrer les requêtes préparées – Méthodes fournies

- Une fois que la requête est préparée, on peut l'utiliser
- Pour cela on définit les paramètres de la requête :
  - Soit en appelant bindValue()

```php
$sql = "INSERT INTO author (firstname, lastname) VALUES ( :firstname, :lastname )";

$statement = $pdo->prepare($sql);

$firstname = 'John';
$lastname = 'Doe';

$statement->bindValue(':firstname',$firstname);
$statement->bindValue(':lastname', $lastname);

$statement->execute();
```

PHP – Requêtes préparées

Paramétrer les requêtes préparées – Méthodes fournies

- Il est possible de rajouter un paramètre supplémentaire à la fonction bindValue() afin de donner des indications sur le type de données à recevoir

```php
$sql = "INSERT INTO author (firstname, lastname) VALUES ( :firstname, :lastname )";

$statement = $pdo->prepare($sql);

$firstname = 'John';
$lastname = 'Doe';

$statement->bindValue(':firstname',$firstname, PDO::PARAM_STR);
$statement->bindValue(':lastname', $lastname, PDO::PARAM_STR);

$statement->execute();
```

PHP – Requêtes préparées

Paramétrer les requêtes préparées – Méthodes fournies

- Voici la liste des constantes pré-définies à connaitre

NOM DE LA CONSTANTE	EXPLICATION
PDO::PARAM_BOOL	Le paramètre doit être du type booléen
PDO::PARAM_NULL	Le paramètre doit être du type NULL
PDO::PARAM_INT	Le paramètre doit être du type ENTIER
PDO::PARAM_STR	Le paramètre doit être du type CHAR ou VARCHAR