

Tecnologies d'accés amb PHP

Introducció:

PHP és un llenguatge de programació interpretat del costat del servidor i d'ús general que s'adapta al desenvolupament web. Permet la connexió a diferents tipus de servidors de bases de dades tant SQL com NoSQL tal com: MySQL, SQLite, PostgreSQL i PDO. Em centraré en PDO, però abans comentaré breument les tecnologies comentades.

Tecnologies:

- MySQL: sistema de gestió de base de dades relacional.
- SQLite: sistema de gestió de base de dades relacional compatible amb ACID. El motor de SQLite no és un procés independent amb el qual el programa principal es comunica. La biblioteca SQLite s'enllaça amb el programa passant a ser part del mateix.
- PostgreSQL: és un sistema de gestió de base de dades relacional orientat a objectes i de codi obert, a més no conté un gestor d'errors (bugs).

PDO

1. Introducció a PDO:

PHP Data Object (PDO) és una interfície lleugera per a accedir a base de dades en qualsevol tipus de PHP, és a dir que les funcions per realitzar consultes i obtenir dades són les mateixes que en qualsevol servidor SQL utilitzant MySQL i PostgreSQL, per exemple. Així i tot, no es pot realitzar ninguna de les funcions de les bases de dades gastant l'extensió PDO per sí mateixa; s'ha d'utilitzar un controlador de PDO específic de la base de dades per a tindre accés a un servidor de base de dades.

2. Classes PDO, PDOStatement i PDOException:

- Classe PDO: per a representar la connexió entre PHP i un servidor de base de dades.
- Classe PDOStatement: representa una sentència preparada i també permet accedir al conjunt de resultats associats.
- Classe PDOException: representa els errors generats PDO.

3. Accés a base de dades amb amb PDO des de PHP:

Connexió a la base de dades:

```
$pdo=new PDO("mysql:host=$host;dbname=$dbname",$usuario,$password);
```

mysql: tipo de base de datos. Podria ser: mssql, sybase, sqlite, etc..

Si tot va bé amb \$pdo es tindrà l'objecte que gestionarà la connexió amb la base de dades.

Tancar la connexió a la base de dades:

Es recomanable tancar sempre la connexió a la base de dades quan no es vaja a utilitzar aquesta. Al fer-ho, s'accelera la lliberació de recursos.

Excepcions i PDO:

PDO pot utilitzar les excepcions per a gestionar els errors, això vol dir que qualsevol cosa que es faça amb PDO es pot encapsular dins d'un bloc try/catch.

Es pot forçar PDO per a que treballi en qualsevol d'aquest modes:

- PDO::ERRMODE_SILENT: El mode per defecte. Per a revisar els errors es gasten ->errorCode() i ->errorInfo().
- PDO::ERRMODE_WARNING: genera errors warning PHP però permet l'execució normal de l'aplicació.

- PDO::ERRMODE_EXCEPTION: la més utilitzada en PDO. Dispara una excepció permetent gestionar l'error de manera amigable.

Insert, update i delete:

Inserir noves dades, actualitzar-los o borrar-los són algunes de les operacions més comuns en una base de dades.

Consultes preparades:

Una consulta preparada és una sentència SQL precompilada que es pot executar moltes vegades enviant dades al servidor.

- Assignació amb marcadors anònims: per a vincular els marcadors amb el seu valor es pot gastar bindParam o bindValue. Diferències: bindParam vincula la variable al paràmetre i en el moment de fer l'executable és quan s'assigna realment el valor de la variable a eixe paràmetre mentre que bindValue assigna el valor de la variable a eixe paràmetre en el moment d'executar bindValue.
- Assignació amb marcadors coneguts: és la manera més recomanable de treballar amb PDO, ja que en el moment d'utilitzar bindParam o bindValue es pot especificar el tipus de dades i la llargària màxima dels mateixos. També permeteixen treballar amb objectes des de la base de dades (en cas que les propietats d'aquest objecte coincideixen amb els noms dels camps de la tabla en la base de dades).

Select:

Les dades s'obtenen mitjançant el mètode ->fetch() o ->fetchAll().

- ->fetch(): obtén la següent fila d'un recordset(conjunt de resultats).
- ->fetchAll(): retorna un array que conté totes les files del conjunt de resultats (el tipus de dades a retornar es pot indicar com a paràmetre).
- fetch_assoc: per a executar la consulta SELECT si no hi ha paràmetres en la consulta es pot utilitzar ->query() de l'objecte PDO.
- fetch_obj: en este tipus de consulta es crearà un objecte standard per cada fila que es llegeix en el recordset.

4. Altres mètodes interessants:

- lastInsertId(): retorna l'id de l'últim registre insertat en la taula.
- quote(): és una manera de protecció contra la injecció SQL.
- rowCount(): retorna un enter indicant el número de files afectades per l'última operació.

5. Webgrafia:

Acceso a base de datos utilizando PDO [en línia] <<http://www.ramip.net/programacion-web/acceso-a-base-de-datos-utilizando-pdo.html>> [Consulta: 20 novembre 2023]

Acceso a bases de datos con PDO [en línia] <https://manuais.iessanclemente.net/index.php/Acceso_a_bases_de_datos_con_PDO#SELECT> [Consulta: 20 novembre 2023]

PHP [en línia] <<https://es.wikipedia.org/wiki/PHP>> [Consulta: 30 novembre 2023]

MySQL [en línia] <<https://ca.wikipedia.org/wiki/MySQL>> [Consulta: 30 novembre 2023]

PostgreSQL [en línia] <<https://es.wikipedia.org/wiki/PostgreSQL#>> [Consulta: 30 novembre 2023]