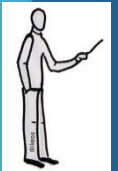


PHP et son interface PDO

- ▶ Auteur → Bruno Diémert
- ▶ Scénariste → Lucile Clausel
- ▶ Pilotage → Mamagalubru
- ▶ Licence → Creative Commons



Enregistrer les données en PHP

- Méthodes

- Cookies

- Variables de sessions

- Fichiers (texte, bureautique, csv, xml, ...)

- Qu'est-ce qu'une base de données relationnelle

- Base de données

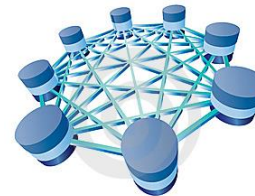
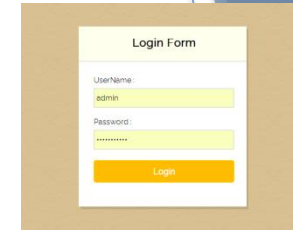
- Relationnelle

- Dans quels cas utiliser des bases de données

- Liaisons multiples

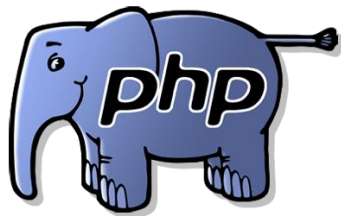
- Temps de conservation

- Grande quantité



Plan du cours PHP Interface PDO

- Enregistrer durablement des données
- Lier PHP et MYSQL via l'interface PDO
- Se connecter à une base de données MYSQL
- Accéder aux tables d'une base de données MYSQL
- Encapsuler les instructions SQL en PHP
- Rechercher, afficher des enregistrements
- Insérer, mettre à jour, supprimer des enregistrements



Lier PHP et MYSQL via l'extension PDO

● Liaison forte PHP et MYSQL

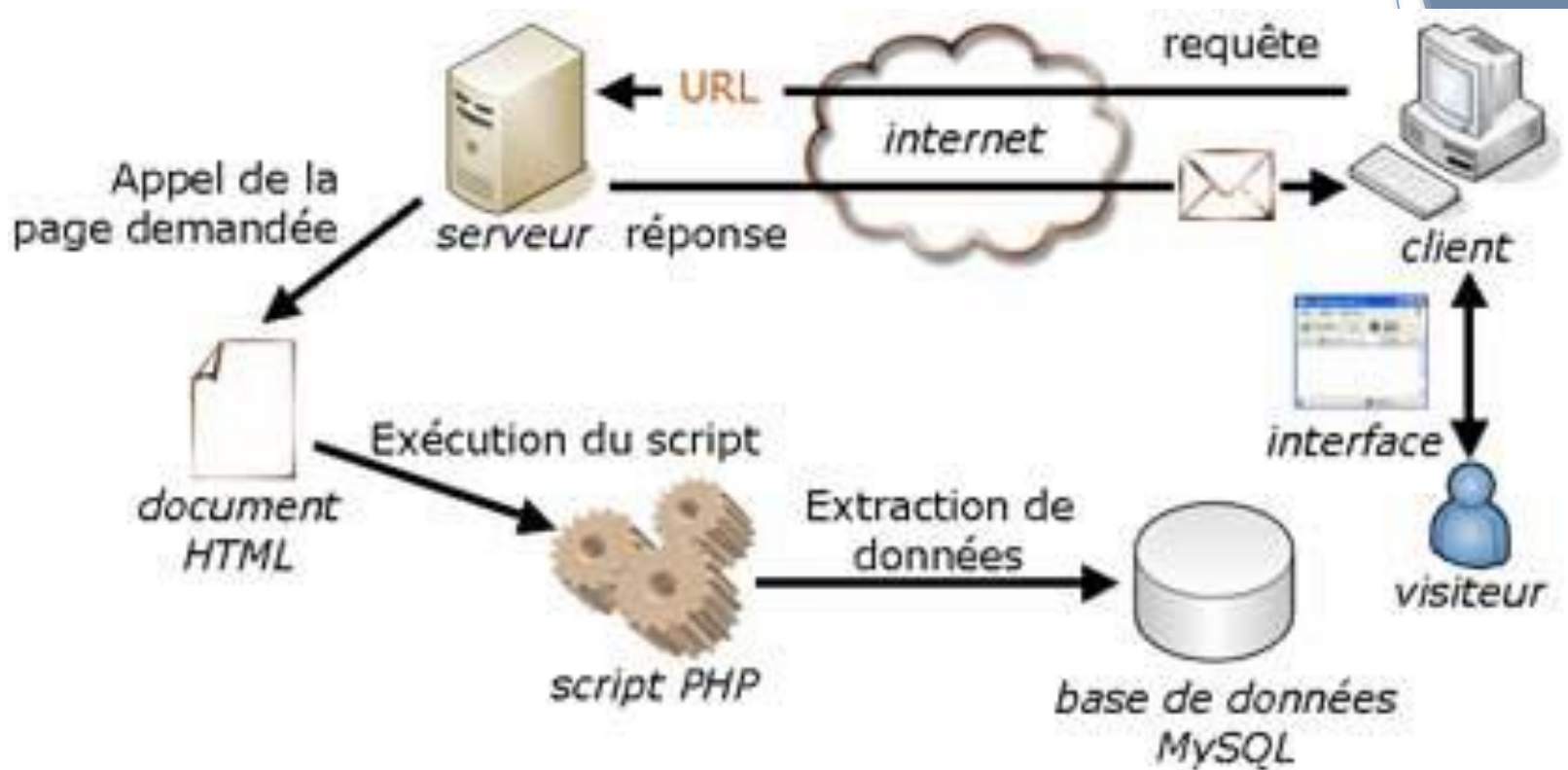
Il existe en standard PHP plus de 100 fonctions prédéfinies pour MySQL dans l'interface PDO, une bonne partie de ces fonctions permettent une encapsulation des ordres du langage SQL

● Liaison forte PDO et SQL

SQL est un langage normalisé permettant d'extraire des informations spécifiques, supprimer, insérer et mettre à jour les données des bases de données grâce à l'extension PDO (vérifier que `php_pdo_mysql` est cochée)



Le cheminement des informations



Se connecter à une base MySQL avec PHP

Connexion PHP à MySQL

```
try {  
    $bd = new PDO('mysql:host=localhost;dbname= 'bd' ;'bond' ;'007');  
}  
Catch (Exception $e) {  
    die('Erreur ouverture BD : '.$e.getMessage());  
}
```

Déconnexion PHP de MySQL

```
$bd = null;
```



L'instruction SQL de requête SELECT

Envoi de la requête de type SELECT PDO PHP à MySQL

```
try {  
    $reponse = $bd->query('select * from entreprises');  
}  
Catch (Exception $e) {  
    die('Erreur lecture BD : '.$e.getMessage());  
}
```

L'instruction SQL de récupération de requête SELECT

Extraction de réponse à requête PDO PHP à MySQL

```
try {  
    $reponse = $bd->query('select nom,prenom from entreprises');  
    while ($donnees = $requete->fetch()) {  
        echo $donnees['prenom'].' ' ;  
        echo $donnees['nom'].'<br>' ;  
    }  
    $reponse->closeCursor() ;  
}  
catch {...}
```


Insert SQL

Connexion, insertion

```
$bd = new PDO('mysql:host=localhost';dbname= 'base' ;'bond' ;'007');  
$bd->exec("INSERT INTO entreprises (sigle, adresse, logo) VALUES ('SNCF','Ici ou  
ailleurs','sncf.jpg') ");
```

Connexion, préparation, insertion

```
$bd = new PDO('mysql:host=localhost';dbname= 'base' ;'bond' ;'007');  
$bd->prepare("INSERT INTO matieres (code, matiere) VALUES (:code,:matiere) ");  
$bd->execute(array('code' => $code, 'matiere' => $matiere)) ;
```

Update SQL

Connexion, update

```
$bd = new PDO('mysql:host=localhost';dbname= 'base' ;'bond' ;'007');  
$bd->exec("UPDATE videos SET prix = '10' WHERE CODE = 'rabais'");
```

Connexion, préparation, update

```
$bd = new PDO('mysql:host=localhost';dbname= 'base' ;'bond' ;'007');  
$bd->prepare("UPDATE videos SET prix =:nvprix WHERE CODE =:code) ;  
$bd->execute(array('nvprix' => $nvprix, 'code' => $code)) ;
```

Delete SQL

Connexion, suppression

```
$bd = new PDO('mysql:host=localhost';dbname= 'base' ;'bond' ;'007');  
$bd->exec("DELETE FROM videos WHERE CODE = 'rabais'");
```

Connexion, préparation, update

```
$bd = new PDO('mysql:host=localhost';dbname= 'base' ;'bond' ;'007');  
$bd->prepare("DELETE FROM videos WHERE CODE = :code) ;  
$bd->execute(array('code' => $code)) ;
```

Les méthodes de la classe PDO

PDO — The PDO class

- PDO::beginTransaction — Initiates a transaction
- PDO::commit — Commits a transaction
- PDO::__construct — Creates a PDO instance representing a connection to a database
- PDO::errorCode — Fetch the SQLSTATE associated with the last operation on the database handle
- PDO::errorInfo — Fetch extended error information associated with the last operation on the database handle
- PDO::exec — Execute an SQL statement and return the number of affected rows
- PDO::getAttribute — Retrieve a database connection attribute
- PDO::getAvailableDrivers — Return an array of available PDO drivers
- PDO::inTransaction — Checks if inside a transaction
- PDO::lastInsertId — Returns the ID of the last inserted row or sequence value
- PDO::prepare — Prepares a statement for execution and returns a statement object
- PDO::query — Executes an SQL statement, returning a result set as a PDOStatement object
- PDO::quote — Quotes a string for use in a query
- PDO::rollBack — Rolls back a transaction
- PDO::setAttribute — Set an attribute

Comparatif PDO ext/MYSQL

	ext/mysqli	PDO_MySQL
Introduite en PHP version	5.0	5.1
Inclus avec PHP 5.x	Oui	Oui
Statut du développement	Active	Active
Cycle de vie	Active	Active
Recommandé pour de nouveaux projets	Oui	Oui
Interface orientée objet	Oui	Oui
Interface procédurale	Oui	Non
L'API supporte les requêtes non-bloquantes, asynchrones avec mysqlnd	Oui	Non
Connexions persistentes disponibles	Oui	Oui
L'API supporte les jeux de caractères	Oui	Oui
L'API supporte les requêtes préparées côté serveur	Oui	Oui
L'API supporte les requêtes préparées côté client	Non	Oui
L'API supporte les procédures stockées	Oui	Oui
L'API supporte les requêtes multiples	Oui	La plupart
L'API supporte les transactions	Oui	Oui
Les transactions peuvent être contrôlées avec SQL	Oui	Oui
Supporte toutes les fonctionnalités de MySQL 5.1+	Oui	La plupart

Quiz



- ▶ La fonction de connexion permettant d'ouvrir une base de donnée MYSQL est ... ?
- ▶ Le langage PHP utilise les paramètres suivants pour la connexion à une base de données : port, localhost, name, sql, open, user, password, base
- ▶ Pour charger un tableau avec une requête, il faut utiliser la fonction \$bd->query(), Oui/Non ?
- ▶ Marier les ordres SQL suivants (GRANT, SELECT, UPDATE, INSERT, ALTER, DELETE, DROP, REVOKE, CREATE) avec leur type de langage(DDL, DML, DSL)
- ▶ Trier par chronologie d'utilisation les fonctions PDO suivantes : new PDO(), prepare(), query(), exec(), closeCursor()...

Exercices (description générale)



- Créer un site avec gestion de base de données et de formulaires

incidents.employes	
ID	int(11)
nom	varchar(80)
prenom	varchar(80)
mail	varchar(160)
tauxHoraire	decimal(10,0)

incidents.logiciels	
ID	int(11)
titre	varchar(80)
description	varchar(240)
version	varchar(10)

incidents tickets	
ID	int(11)
titre	varchar(100)
description	varchar(500)
priorite	varchar(10)
status	varchar(10)
idlogiciel	int(11)
idemploye	int(11)
tempsResolution	decimal(10,0)

Exercices (description détaillée)



- ✓ Créer une connexion à une base de données
- ✓ Créer une table
- ✓ Peupler une table
- ✓ Peupler une table avec requête préparée
- ✓ Lire les enregistrements d'une ou plusieurs tables et les afficher dans une page HTML
- ✓ Modifier les enregistrements d'une table grâce à un formulaire de saisie
- ✓ Détruire les enregistrements d'une table grâce à un formulaire de saisie