

PHP

Passez à la vitesse supérieure avec PHP

Plan du cours

- Introduction à PHP
 - C'est quoi ?
 - Utilisation dans le développement web
- Architecture Client / Server
- Communication entre Client et Serveur
- Fonctionnement de site web statique et dynamique
- Environnement de développement
 - Téléchargement et installation d'un serveur (XAMPP, WAMP, MAMP,...)
- Premier pas avec PHP
 - Structure d'une page PHP
 - Affichage de texte
 - Commentaires

Plan du cours

- Variables et types de données
 - Déclaration de variable
 - Types de données de base
 - Opération sur les données
- Les structures conditionnelles
 - if, else, elseif, switch
- Les structure itératives
 - for, while, do while
- Inclusions de fichiers
 - include
 - require
- Les fonctions

Plan du cours

- Les tableaux
 - Tableaux indicés
 - Tableau associatifs
 - Parcours et affichage des éléments d'un tableau
- Formulaires HTML et Traitement PHP
 - Création des formulaire HTML
 - Soumission de formulaire et traitement des données
 - Validation des données côté serveur
- Interaction avec une Base de données CRUD
 - Requête d'insertion
 - Requête de lecture
 - Requête de mise à jour
 - Requête de de suppression

Plan du cours

- Sessions et Cookies
 - Utilisation des sessions pour maintenir l'état des utilisateurs
 - Gestion des cookies
- Gestion de fichiers
 - Lecture
 - Ecriture
- Hébergement d'un site

Interaction avec les bases de données

L'interaction avec une base de données est une compétence essentielle dans le développement web, car elle permet de stocker les données de manière persistante, les récupérer et les gérer.

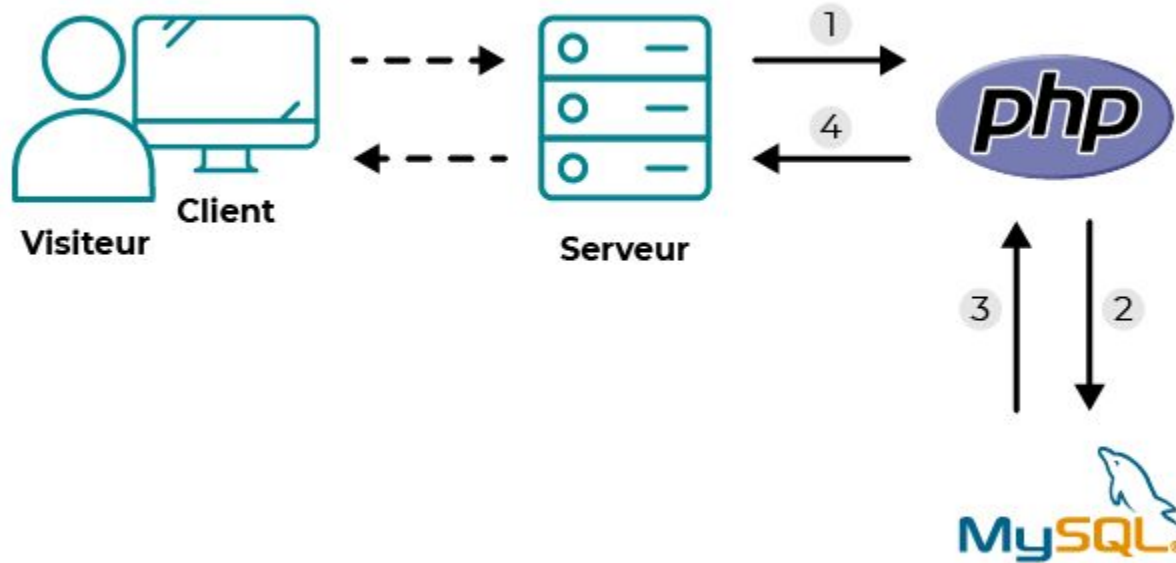
Interaction avec les bases de données

Une base de données est un ensemble structuré et organisé permettant le stockage de grandes quantités d'informations afin d'en faciliter l'exploitation (ajout, mise à jour, recherche de données).

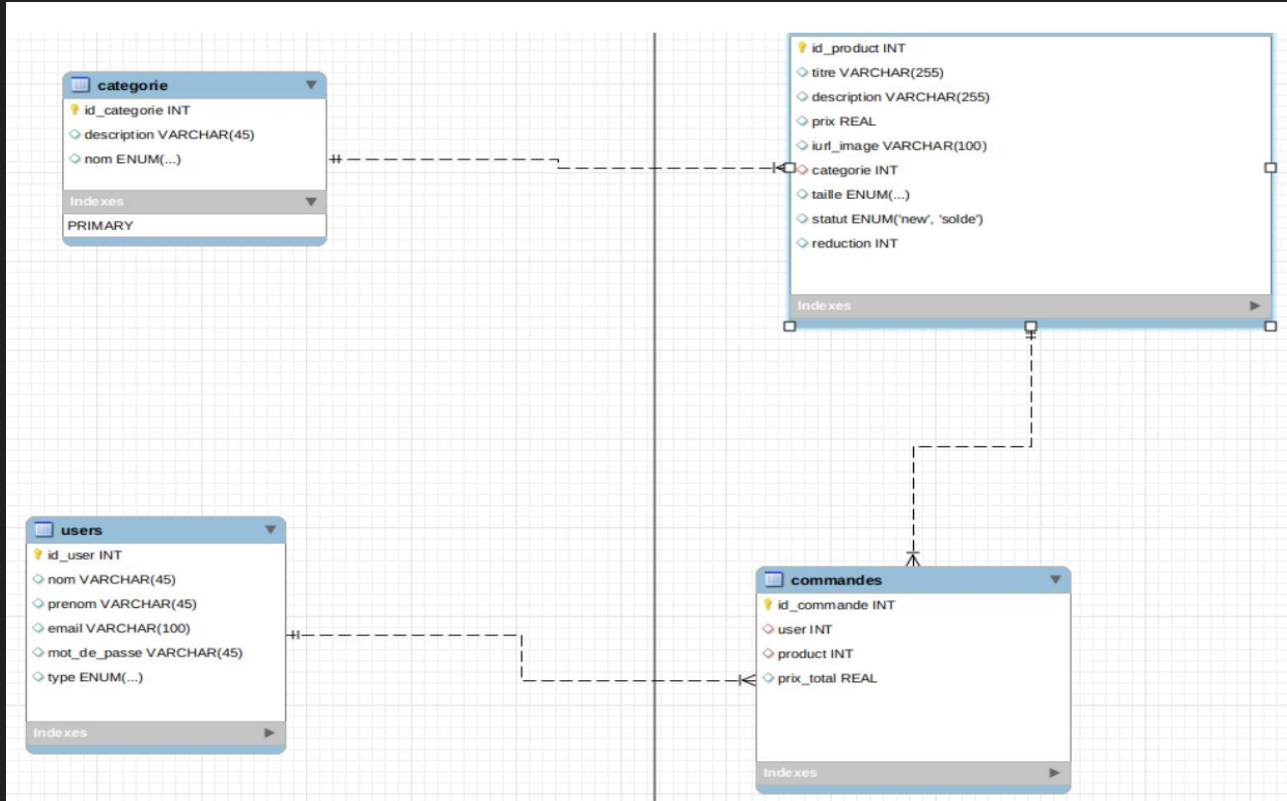
Pour permettre une utilisation optimale d'une base de données, il faut un **SGBD** Système de Gestion de Base de Données, (en anglais DBMS : Database management system).

le langage permettant d'interroger une base de données s'appel SQL (Structured Query Language).

Interaction avec les bases de données



structure d'une base de données



structure d'une table

Identifiant	Nom	Prenom
1	Dupont	Robert
2	Doe	John
3	Thiery	Thomas

connexion à une base de données avec PDO

PDO (PHP Data Objects) est une extension PHP qui fournit une interface unifiée pour accéder à une base de données.

Elle offre une couche d'abstraction qui facilite la gestion des connexions à la base de données et la manipulation des données, tout en favorisant la sécurité en encourageant l'utilisation de requêtes préparées.

voici comment établir une connexion avec une base de données:

```
$connexion = new PDO("mysql:host=adresse_du_serveur;dbname=nom_base_de_données" , "login", "mot_de_passe");
```

CRUD

Une fois la connexion établie avec la base de données on peut maintenant faire un CRUD.

CRUD est un acronyme qui désigne les opérations de base effectuées sur les données d'une base de données. Chaque lettre dans l'acronyme correspond à une action spécifique :

C - Create (Créer) :

L'opération de création consiste à ajouter de nouvelles données dans la base de données. Cela correspond à l'insertion de nouvelles lignes dans une table de la base de données.

CRUD

1. **R** - Read (Lire) :
2. L'opération de lecture consiste à récupérer des données existantes de la base de données. Cela correspond la sélection et l'affichage de données spécifiques à partir d'une table.
3. **U** - Update (Mettre à jour) :
4. L'opération de mise à jour consiste à modifier les données existantes dans la base de données. Cela correspond à la modification des valeurs de certaines colonnes d'une ligne.
5. **D** - Delete (Supprimer) :
6. L'opération de suppression consiste à retirer des données de la base de données. Cela correspond à la suppression de lignes spécifiques d'une table.

Requête d'insertion

```
// connexion a la base de données
$connexion = new PDO("mysql:host=adresse_du_serveur;dbname=nom_base_de_données", "login",
"mot_de_passe");

// preparation de la requete
$request = $conn->prepare("INSERT INTO la_table (champ1, champ2,...) VALUES (?, ?,...)");

// execution de la requete
$request->execute(array($valeur1, $valeur2, ...));
```

Requête de lecture

```
// connexion a la base de données
$connexion = new PDO("mysql:host=adresse_du_serveur;dbname=nom_base_de_données", "login",
"mot_de_passe");

// preparation de la requete
$request = $conn->prepare("SELECT * FROM la_table");

// execution de la requete
$request->execute();

// recuperation du resultat de la requete
$resultat = $request->fetchAll();
$resultat = $request->fetch();
```

Requête de modification

```
// connexion a la base de données
$connexion = new PDO("mysql:host=adresse_du_serveur;dbname=nom_base_de_données", "login",
"mot_de_passe");

// preparation de la requete
$request = $conn->prepare("UPDATE nom_table SET nom_colonne = ? WHERE nom_colonne = ?");

// execution de la requete
$request->execute(array($valeur1, $valeur2));
```


Requête de suppression

```
// connexion a la base de données
$connexion = new PDO("mysql:host=adresse_du_serveur;dbname=nom_base_de_données", "login",
"mot_de_passe");

// preparation de la requete
$request = $conn->prepare("DELETE FROM nom_table WHERE nom_colonne = ?");

// execution de la requete
$request->execute(array($valeur));
```

Les Sessions

En PHP, les sessions sont un mécanisme qui permet de stocker et de gérer des données utilisateur entre différentes pages web. Les sessions sont utiles pour maintenir l'état de l'utilisateur sur un site web, en conservant certaines informations même lorsque l'utilisateur navigue entre les pages.

Démarrage de session

Avant d'utiliser les sessions, on doit d'abord démarrer une session en utilisant la fonction `session_start()`. Cette fonction doit être appelée au début de chaque page où on souhaite utiliser les sessions.

```
<?php
session_start();
?>
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Document</title>
  </head>
  <body>
    ...
  </body>
</html>
```

Créer des variables de sessions

ont peut stocker des variables de sessions en utilisant le superglobal `$_SESSION`.

exemple:

```
$_SESSION['nom'] = 'John';
```

```
$_SESSION['age'] = 30;
```

Lecture des variables de sessions

On peut accéder aux variables de session sur n'importe quelle page où la session est active en utilisant à nouveau `$_SESSION`.

exemple:

```
echo "Bonjour, " . $_SESSION['nom'];
```

Supprimer les variables de sessions

Lorsqu'un utilisateur se déconnecte ou que la session doit être fermée, vous pouvez utiliser la fonction `session_destroy()` pour supprimer toutes les données de la session.

exemple:

```
session_destroy();
```