

Toulibre : présentation de ...



Mises en garde

- Cette présentation n'est qu'une présentation, et n'a aucun but d'exhaustivité : elle aspire seulement à aider votre insertion dans le monde de PHP.
- ~~Ce que vous allez voir est réalisé par des professionnels, n'essayez en aucun cas de le reproduire chez vous.~~ Vous êtes encouragé à tester, bidouiller, trafiquer et faire plein de bêtises, c'est comme ça qu'on apprend !
- De même, vous êtes encouragé à poser des questions.

La programmation

- Les ordinateurs servent à calculer des choses plus ou moins complexes.
- Ces calculs s'étendent de la simple addition à la génération de pages web ou de graphismes 3D.
- Afin d'effectuer ces calculs, il faut donner à l'ordinateur une suite d'instruction à suivre, c'est à dire le programmer.

Principes de base : les variables

- Les variables sont un moyen de mettre une valeur en mémoire afin de la réutiliser ultérieurement.
- Chaque variable a un type : nombre entier, réel, chaîne de caractère, tableau, etc...
- PHP converti les types implicitement, mais il faut y faire attention, car ceci est une source de failles de sécurité.
- En php, les variables se notent *\$nom_de_la_variable*

Principes de base : les fonctions

- Les fonctions sont une suite d'instructions prédéfinies.
- Elles peuvent prendre comme "argument" une ou plusieurs variables, qui servent à changer des détails dans l'exécution des instructions. Par exemple : *ajouter(2, 2)*.
- Une fonction peut retourner une valeur.
- En PHP, un grand nombre de fonctions existent dans le noyau de PHP, et plus encore dans des extensions.
- Il est aussi possible de définir ses propres fonctions

Principes de base : les opérateurs

- Ils opèrent sur les valeurs (principalement les variables et les valeurs retournées par les fonctions) pour les additionner, les comparer, etc.
- Il en existe plusieurs types
 - Attribution (`=`, `.=`, `+=`, `-=`)
 - Arithmétique (`+`, `-`, `*`, `/`)
 - Logique (`and`, `or`, `nor`)
 - Comparaison (`<`, `>`, `<=`, `>=`, `!=`, `==`)
 - Et d'autres types moins utilisés
- Par exemple : `$ajout = 2 + 2;`

Principes de base : les structures de contrôle

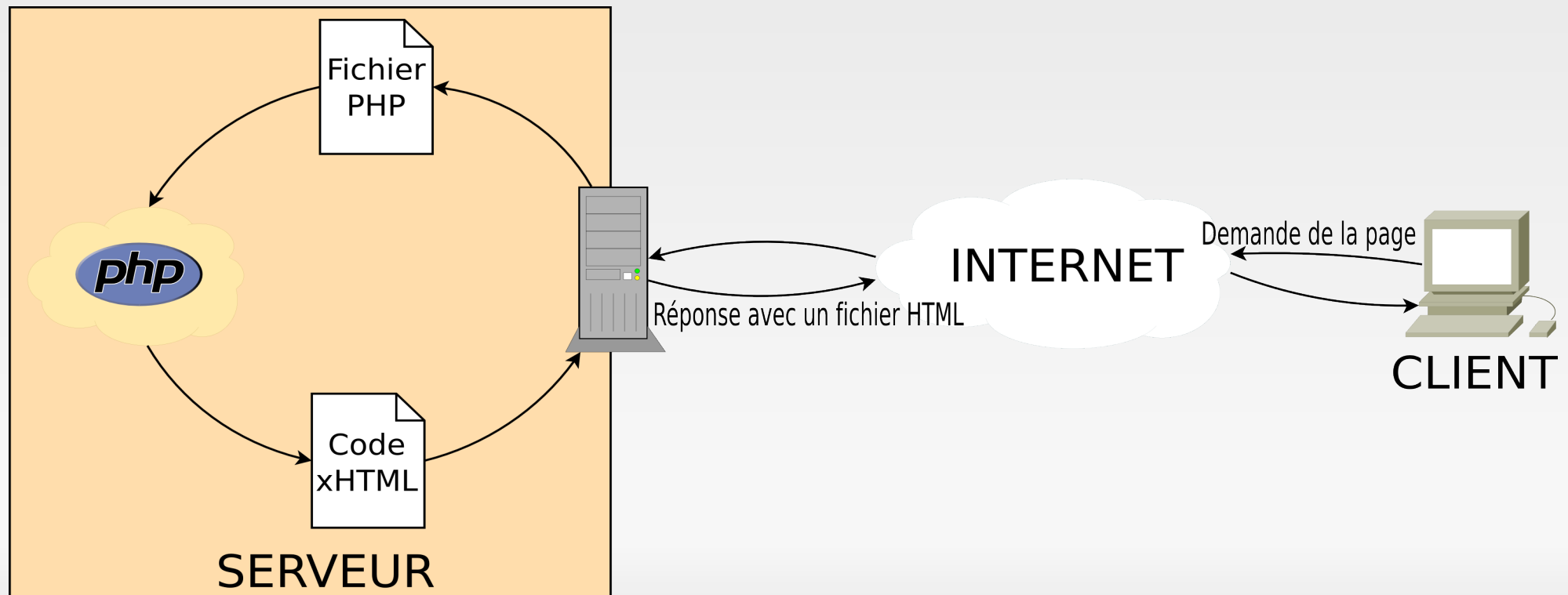
- Ce sont des instructions qui servent à faire prendre au programme un cheminement différent selon certaines conditions.
- Les principales sont :
 - La condition : *if()*
 - Les boucles *for()*, *while()* et *foreach()*
- Bien qu'il existe certaines exceptions permettant de passer outre ces structures, il est fortement conseillé de les utiliser le plus possible.

Et PHP dans tout ça ?

- Au milieu des années 90, un webmaster amateur décide de "dynamiser" son CV.
- À la base il voulait conserver une trace des visites. Il commence donc à coder des éléments qu'il peut insérer dans du HTML.
- Peu à peu, sa Personal Home Page s'étoffe.
- Depuis PHP a élargi ses horizons : il dispose de fonctions variées allant de la cryptographie à la lecture de fichiers audio.

Le mode de d'action de PHP

- PHP est prévu générer une page XHTML spécifique à la demande du client.
- Il s'intègre parfaitement avec le XHTML.



Petite démonstration...

- Quels sont les éléments de base du PHP ?
- Comment le PHP s'intègre-t-il dans le xhtml ?
- Comment peut-on inclure plusieurs fichiers dans un seul ?

Les bases de données

- Il faut souvent stocker des données sous forme de tableaux : on a créé pour cela les bases de données.
- Elles stockent les données dans un tableau à simple entrée : la liste des vente d'un commerçant par exemple.
- Dans le cas d'un site internet, elle va être utilisée pour stocker les pages, les auteurs, les commentaires, les visiteurs, etc...

Explications sur le fonctionnement des BDD

- On peut faire l'analogie entre des tableaux et une base de données.
- Chaque tableau est une table.
- Une colonne d'un tableau est un champ.
- La ligne devient un enregistrement (mais garde souvent le nom de ligne en PHP).
- À l'instar des variables, chaque champ possède un type : nombre, texte, date, mot de passe...

Explications sur le fonctionnement des BDD

- Le moteur de base données donne des outils qui permettent de traiter les données stockées.
- Il est possible de chercher une ou plusieurs lignes selon certains critères.
- Il est aussi possible de relier plusieurs tables entre elles : par exemple associer un auteur à plusieurs articles.
- Un langage a été mis en place pour que le programme puisse communiquer avec la base de données : SQL.

SQL

- Quand une application veut des données de la base, elle se connecte au serveur de base de données puis effectue une requête.
- Exemple : `SELECT * FROM article`
- Le serveur répond par un tableau contenant uniquement les données demandées.

MySQL

- MySQL est le moteur de base de données "préfér " de PHP.
- Il utilise sa variante de SQL pour les requettes.
- => D monstration

PHP/MySQL chez vous

- Pour utiliser PHP afin de faire des pages web, il faut un serveur web. Le plus répand est Apache.
- Sous windows, il existe des logiciels libres tels que EasyPHP ou WAMP qui installent tout seul PHP, MySQL et Apache.
- Sous linux, il suffit généralement d'installer les paquets de Apache, de PHP et de MySQL pour obtenir un serveur fonctionnel.
- Pour rendre un site visible sur le web, il faut utiliser un hébergeur.

La réalisation du jour

- Nous allons réaliser un site typique :
 - Différents articles
 - Un menu généré automatiquement
 - La possibilité pour les utilisateurs de rajouter leurs commentaires
- La phase qui consiste à créer un design au site a été faite à la précédente présentation du XHTML.

Méthode de réalisation

- Dans un premier temps il faut considérer les besoins pour mettre au point la structure de la base de données et du code PHP.
- La réalisation de la base de données dans une version minimale vient par la suite : elle permet au code PHP d'avoir quelque chose à utiliser pour le développement.
- On réalise ensuite la partie PHP.
- Une fois que tout opérationnel, il ne reste plus qu'à remplir le site (et ce n'est pas le plus facile !)
- Au travail !

Pour aller plus loin

- On peut trouver des tutoriaux à profusion sur developpez.com
- Le site de php (php.net) fournit une bonne documentation sur les différents éléments du langage.
- Le meilleur moyen d'apprendre est de lire du code source et de se lancer dans un projet concret !

The End

```
<?php echo "fin"; ?>
```

C'est fini

