I) Principes et concepts

PHP - Initiation

Principes et concepts

I.1) Les balises PHP

Les scripts PHP sont très souvent semblables à des pages HTML dont certaines zones seront substituées par le résultat de l'exécution d'un programme.

Les zones de scripts sont placées entre les balises **<?php** et **?>**.

Selon la configuration du **php.ini** (directive **short_open_tag = ON**) la balise d'ouverture du script pourra également être **<?** (à éviter).

Le déclenchement de l'interprétation sera fait la plupart du temps si le nom du fichier se termine par **.php**.

PHP - Initiation

Principes et concepts

I.2) Fonctions d'affichage

Le but initial de PHP est de permettre la création de page web dynamiques et donc de pouvoir **produire des flux de texte HTML**.
2 fonctions principales permettent de produire du texte en sortie :

- le mot clef **echo**
- la fonction **print()**

Ces fonctions prennent en argument des chaînes alphanumériques éventuellement placées dans des variables, et envoient leur contenu à l'écran. Le mode **echo** permet d'envoyer des valeurs séparées pas des **,** sans avoir à les entourer de **()**.

Syntaxe

echo *mon_texte*
print(*mon_texte*);

PHP - Initiation

Principes et concepts

Exemple

```
<html>...
  <body>
    <?php
      echo '<p>Bonjour à tous</p>';
    ?>
  </body>
</html>
```

Afficher dans le navigateur le code source HTML d'une page générée par PHP ne permet de voir que le code HTML produit. Ainsi le résultat de l'ouverture de la page précédente sera :

```
<html>...
  <body>
    <p>Bonjour à tous</p>
  </body>
</html>
```

PHP - Initiation

Principes et concepts

I.3) Syntaxe et construction

La syntaxe de PHP est très proche de celle du langage C.
L'usage est de placer **une instruction par ligne**. (Il est cependant possible de mettre plusieurs instructions sur une même ligne, tant que l'on respecte la "séparation" des instructions)

Le séparateur d'instruction est **;** (**point-virgule**).

Comme pour tout langage de programmation, il est recommandé d'avoir recours de façon systématique à **l'indentation**, ce qui augmente la lisibilité.

Il faut penser que dans un script PHP peuvent cohabiter du HTML et du script PHP, on a donc tout intérêt à se soucier de la **lisibilité du code**.

PHP - Initiation

Principes et concepts

Le code PHP peut donc être implanté au sein du code HTML.
Le fait d'écrire uniquement du code PHP là où il est nécessaire rend la programmation plus simple. (Il est plus simple d'écrire du code HTML qu'un enchaînement de fonctions **echo**).

Il est tout à fait possible d'écrire plusieurs "portions" de script PHP à divers endroits de notre code HTML (y comprit avant même la balise **<HTML>**). En effet, les variables ou fonctions déclarées seront accessibles dans l'intégralité de la page.

```
<html>
  <head>
    <title><?php echo 'ma première page'; ?></title>
  </head>
  <body>
    <?php echo 'Salut à tous !'; ?>
  </body>
</html>
```

PHP - Initiation

Principes et concepts

I.4) Les commentaires

A ne surtout pas négliger, comme dans tout langage !
Là encore la syntaxe est inspirée du C.

- Commentaire sur une ligne : **//** ou **#**
- Commentaire sur plusieurs lignes : **/*** ... ***/**

Exemple

```
<?php
// Voici un commentaire sur une ligne
# Un autre commentaire sur une ligne
/* Et ici un commentaire
sur plusieurs
lignes */
?>
```

PHP - Initiation

Principes et concepts

I.5) Les alertes

Certaines commandes peuvent émettre des alertes.
Le mode **error_reporting** du fichier **php.ini** permet de modifier le niveau de verbosité.
Le caractère **@** placé en préfixe d'un commande permet de supprimer localement les messages d'alertes.

Exemple

```
<?php
// la commande suivante n'affichera aucune alerte.
    @fopen('fichier_inexistant', 'r');

// la commande suivante affichera une alerte.
    fopen('fichier_inexistant', 'r');
?>
```