

Internet et Outils

L1/IO2 2007 - 2008

S2-IO2

PHP



François Armand

`armand@informatique.univ-paris-diderot.fr`

Plan Général Prévisionnel (1/2)

(non contractuel)

• Cours Internet et Outils:

- [1/12] Intro, Internet, Web, XHTML (2H)
- [2/12] XHTML(2H)
- [3/12] CSS (2H)
- [4/12] **PHP (2H)**
- [5/12] Intégration XHTML, CSS, PHP (2H)
- [6/12] Introduction MySQL, Table simple (2H)

- [7/12] ► **Partiel (2H) 15 Mars 08 14H-16H** ◀
Amphis 2A et 8C

Plan Général Prévisionnel (2/2)

(non contractuel)

- Bases de données
 - [8/12] Tables multiples, Schémas (2H)
 - [9/12] Concepts un peu plus avancés (2H)
 - [10/12] Introduction aux réseaux(2H)
 - [11/12] Réseaux: compléments (2H)
 - [12/12] Révisions (2H)

PHP: Plan

- **Ce que vous allez découvrir**
- PHP et XHTML: premiers éléments
- Exécution conditionnelle
- Variables, Types de Données
- Chaînes
- Formulaire, vérification des champs
- Contrôle du flux d'exécution
- Fonctions
- Tableaux
- Ce qu'il faut retenir

Ce que vous allez découvrir

- Enrichir les documents XHTML avec du PHP
 - Comment imbriquer du PHP et du XHTML
 - Comment déclarer et manipuler des variables PHP
 - Créer et gérer des tableaux en PHP
 - Quelques fonctions: date, manipulation de chaînes
 - Contrôler l'exécution d'un script PHP
 - Réutiliser du code PHP

PHP: Plan

- Ce que vous allez découvrir
- **PHP et XHTML: premiers éléments**
- Exécution conditionnelle
- Variables, Types de Données
- Chaînes
- Formulaire, vérification des champs
- Contrôle du flux d'exécution
- Fonctions
- Tableaux
- Ce qu'il faut retenir

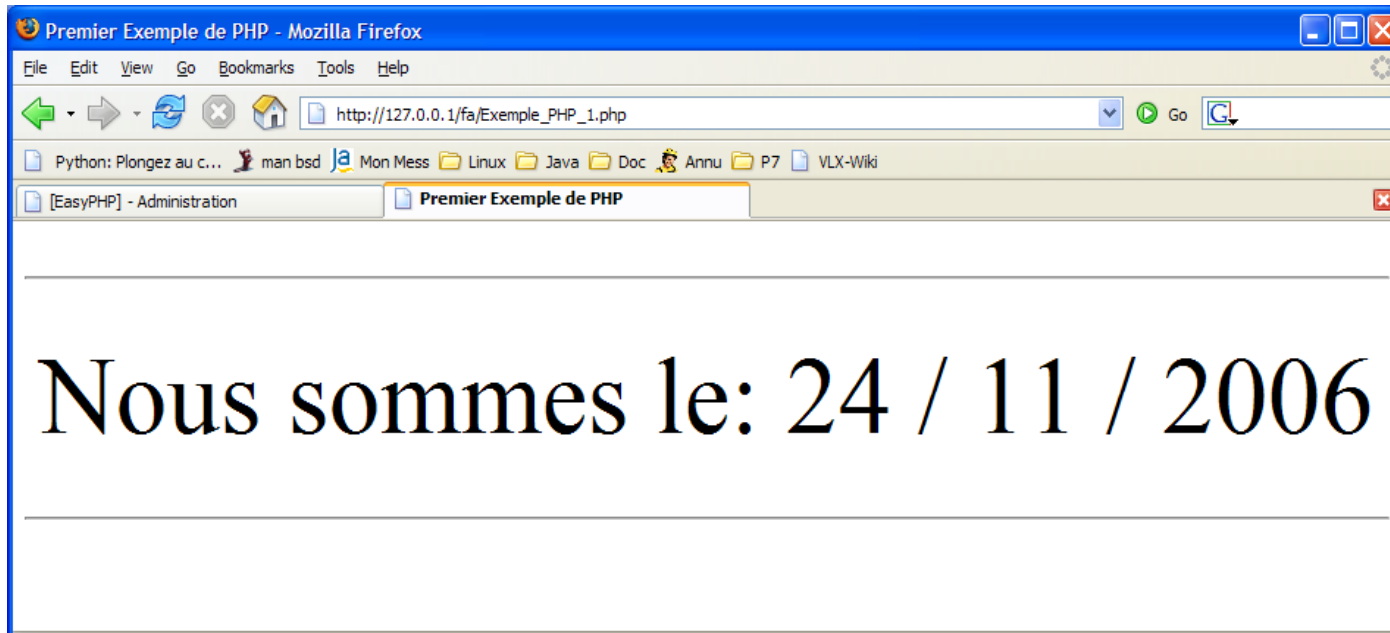
EasyPHP

- <http://www.easyphp.org/>
 - Environnement PHP, MySQL, PHPMYADMIN
 - Pour Windows
 - Permet un développement local et facile de scripts PHP, et de bases de données MySQL

PHP HyperText Preprocessor

- Langage de "programmation" proche du C
- Langage de script
 - interprète des commandes "built-in" ou pas
 - de nombreux modules d'interface avec d'autres outils dont évidemment MySQL
- S'utilise et s'exécute côté serveur pour produire "aisément" un document [x]html à renvoyer au client
- Imbriqué avec le "document html"
- Renommer le fichier ".html" en ".php"

Résultat Premier Exemple PHP



Exemple: fichier_(src PHP) côté "serveur"

```
<html>
```

```
<head><title>Essai de PHP</title></head>
```

```
<body style="font-size: x-large; text-align: center">
```

```
<hr />
```

Nous sommes le `<?php` `echo Date ("j / m / Y");` `?>`

```
<hr />
```

```
</body>
```

```
</html>
```

La balise `<?php` est conforme à XHTML

Code reçu par le Navigateur

```
<html>
<head>
<title> Premier Exemple de PHP </title>
</head>
<body style="font-size: x-large; text-align:center">
<hr />
Nous sommes le: 24 / 11 / 2006<br />
<hr />
</body>
</html>
```

Deuxième Exemple (Firefox)



Deuxième Exemple_(src php)

```
<h1>Un essai de PHP</h1>
```

```
<?php
```

```
echo "Client: ", $_SERVER["HTTP_USER_AGENT"];
```

```
echo "<br /><br />Serveur:";
```

```
echo $_SERVER["SERVER_NAME"], "<br />"
```

```
?>
```

- Balise PHP `<?php..... ?>`
- 3 commandes "echo" séparées par des `;`
- Arguments de "echo" séparés par des `,`
- `$_SERVER["chaine"]`: tableau... cf plus loin

Deuxième Exemple, suite

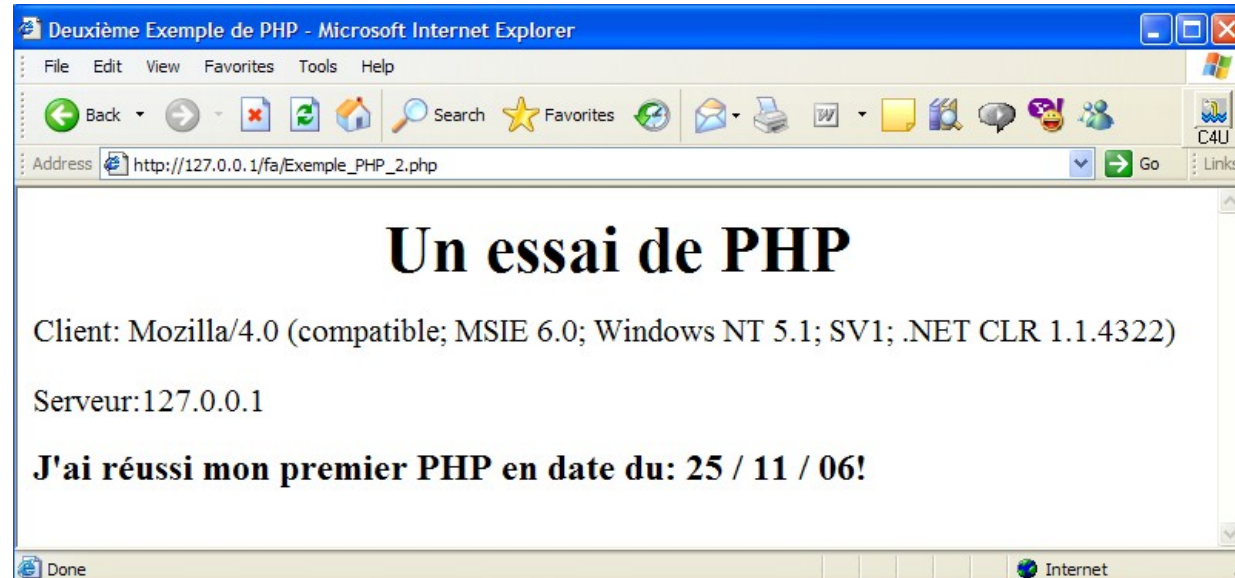
<h3> J'ai réussi mon premier PHP

```
<?php echo "en date du: " , Date ("d / m / y"), "!" ?>
```

</h3>

- On peut insérer plusieurs séquences PHP dans un fichier.
- Chaînes entre "..."
- Date est une fonction PHP (cf plus loin)

Résultat Affiché (I.E.)



- La chaîne affichée après "Client" diffère, et permet d'identifier la version de navigateur (**user agent**)

Code reçu par le navigateur

```
<html><head>
<title> Deuxième Exemple de PHP </title>
</head> <body style="font-size: x-large">
<h1 style="text-align:center">Un essai de PHP</h1>
```

Client: Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US;
rv:1.8.0.8) Gecko/20061025 Firefox/1.5.0.8

Serveur:127.0.0.1

<h3> J'ai réussi mon premier PHP
en date du: 24 / 11 / 06!</h3>

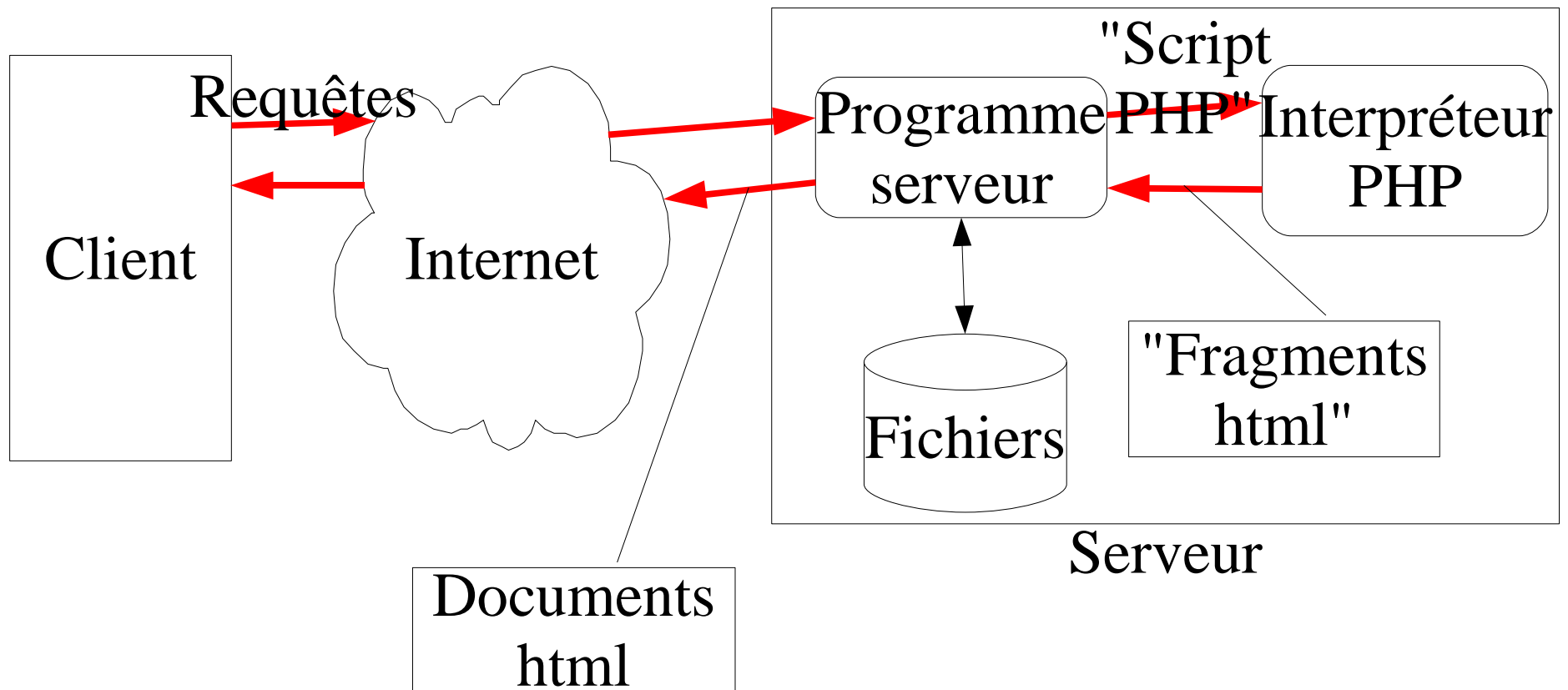
</body> </html>

- **En bleu:** les parties générées par le code PHP

Rôle du serveur

- Avec du [x]html, le serveur se "contente" de trouver le fichier contenant le document correspondant à l'URL et de le renvoyer
- Avec PHP, le serveur doit d'abord analyser les documents pour:
 - trouver toutes les occurrences de `<?php... ?>`
 - faire interpréter les commandes php par un interpréteur php
 - Substituer les chaînes `<?php... ?>` par le résultat de l'interprétation

Architecture CGI



Aide, Manuel PHP

- Le monde PHP est (très) vaste et (très) riche
- Ce cours n'a pour but que de vous donner quelques rudiments,
- Si vous cherchez une réponse à une question,
- Si vous cherchez une fonction, une syntaxe,
- Consultez:

<http://www.php.net/> Site officiel

<http://fr2.php.net/manual/fr/> Pages manuels en français

http://fr.wikipedia.org/wiki/PHP_hypertext_preprocessor

PHP: Plan

- Ce que vous allez découvrir
- PHP et XHTML: premiers éléments
- **Exécution conditionnelle**
- Variables, Types de Données
- Chaînes
- Formulaire, vérification des champs
- Contrôle du flux d'exécution
- Fonctions
- Tableaux
- Ce qu'il faut retenir

Commentaires PHP

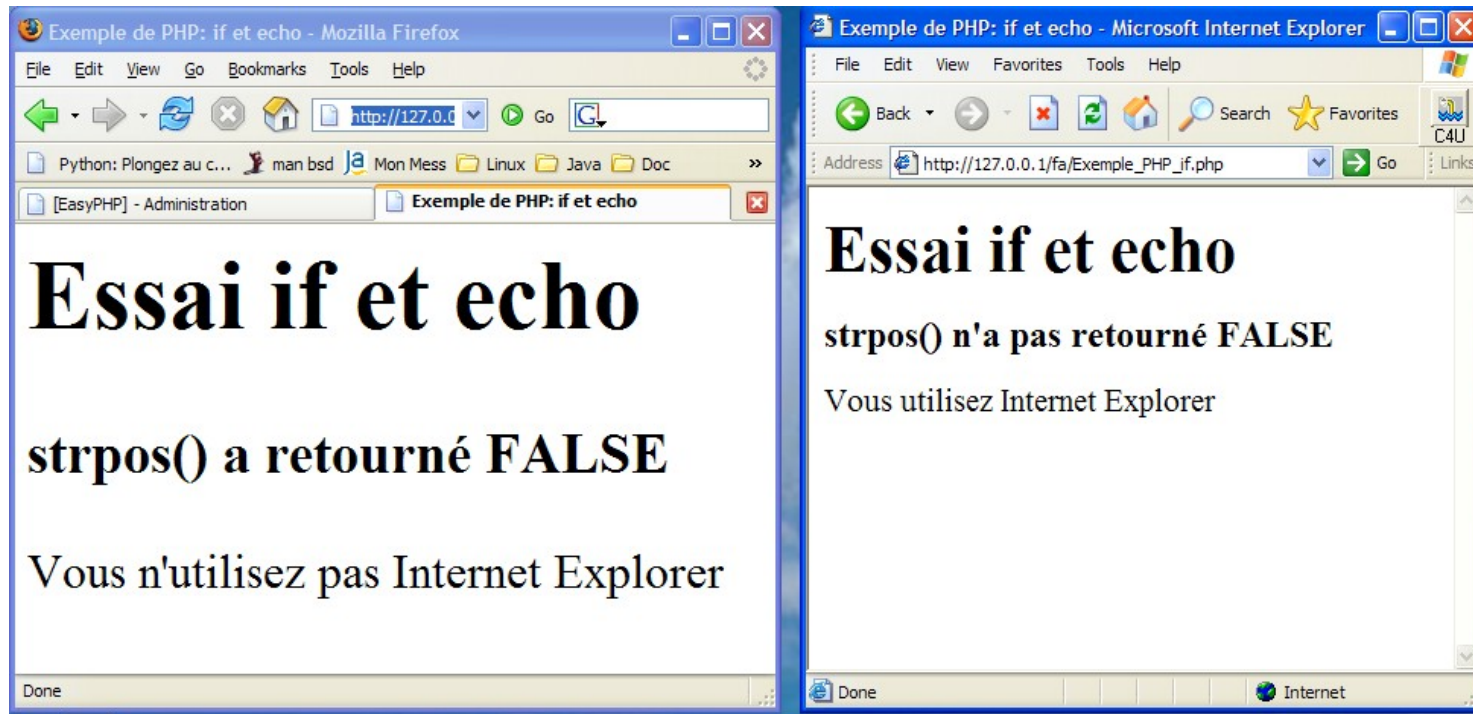
- Insérer des commentaires:
 - /* ceci est un
commentaire */
 - // commentaire jusqu'à la fin de ligne
 - # Ceci également

Exécution conditionnelle (php)

```
<?php
```

```
if (strpos($_SERVER['HTTP_USER_AGENT'], 'MSIE') !== FALSE) {  
    echo "<h3>strpos() n'a pas retourné FALSE</h3>";  
    echo "<p>Vous utilisez Internet Explorer</p>";  
} else {  
    echo "<h3>strpos() a retourné FALSE</h3>";  
    echo "<p>Vous n'utilisez pas Internet Explorer</p>";  
}  
?>
```

Résultats



Firefox

Internet Explorer

Exécution conditionnelle

```
if (condition) {  
    // code exécuté si la condition est vraie  
} else {  
    // code exécuté si la condition est fausse  
}
```

condition: vraie ou fausse (voir plus loin)

égalité a == b

inégalité a != b

strpos

- Dans l'exemple précédent, nous avons utilisé:
- **strpos** (chaîne1, chaîne2)
 - fonction qui renvoie:
 - indice du premier caractère de chaîne 2 dans chaîne1, si chaîne 2 est contenue dans chaîne1
 - Ex: strpos ("aaabbbccddeebbb", "bb") => 3
 - **FALSE** sinon
- PHP propose de nombreuses fonctions

Code PHP identique sans "echo" (voir)

```
<?php
```

```
if (strpos($_SERVER['HTTP_USER_AGENT'], 'MSIE') !==  
    FALSE) { ?>
```

```
<h3>strpos() n'a pas retourné FALSE</h3>
```

```
<p>Vous utilisez Internet Explorer</p>
```

```
<?php } else { ?>
```

```
<h3>strpos() a retourné FALSE</h3>
```

```
<p>Vous n'utilisez pas Internet Explorer</p>
```

```
<?php } ?>
```

PHP: Plan

- Ce que vous allez découvrir
- PHP et XHTML: premiers éléments
- Exécution conditionnelle
- **Variables**, Types de Données
- Chaînes
- Formulaire, vérification des champs
- Contrôle du flux d'exécution
- Fonctions
- Tableaux
- Ce qu'il faut retenir

Variables PHP

- Noms:
 - Commencent par \$
 - Longueur quelconque
 - Composés de lettres, chiffres, _
 - Lettre ou _ après le \$
 - Majuscules et Minuscules ne sont pas identiques
- Ex:
 - \$ma_var , \$Ma_Var, \$mon_1er_nom, \$_jyvai

Variables PHP

- Créer des variables:
 - \$note = 19;
 - \$coeff = 1.53;
 - \$Moi = "Francois Armand"

```
print_r ($note);
```

```
echo " Note: ", $note, " Nom: ", $Moi, " c: ", $koeff;
```

```
19 Note: 19 Nom :Francois Armand c:
```

Notice: Undefined variable: koeff in ex.php on line 2

- echo @\$koeff // n'affiche pas l'erreur

Variables PHP: exemple

```

<?php
$note = 19; $coeff = 1.53; // entier, valeur réelle
$Mon_Nom="Francois Armand"; // chaine de car.
print_r($note);
echo "<br /> Note: ", $note, "      Nom : ", $Mon_Nom,
    "<br />Autre coef: ", $koeff; // koeff pas defini!
echo @$koeff,"Ca marche?"; $Mon_Nom="Gaston";
?>
<p>Ma foi, cela semble fonctionner comme attendu.</p>
<?php echo "Mon nom est-il toujours d&eacute;fini? ",
    $Mon_Nom; // reference Mon_Nom defini ci-dessus ?>

```

Variables PHP: exemple

Essai de variables PHP - Mozilla Firefox

File Edit View Go Bookmarks Tools Help

http://127.0.0.1/fa/Exemple_PHP_variables.php

Python: Plongez au c... man bsd Mon Mess Linux Java Doc Annu P7 VLX-Wiki

[EasyPHP] - Administration php PHP: La syntaxe de base - Manual php PHP: strpos - Manual Essai de variables PHP

Un essai d'utilisation de variables

19

Note: 19 Nom : Francois Armand

Autre coef:

Notice: Undefined variable: koeff in **c:\documents and settings\francois\desktop\francois\p7\paris7\l1_06_07\myeasyphp\exemple_php_variables.php** on line 11

Ca marche?

Ma foi, cela semble fonctionner comme attendu.

Mon nom est-il toujours défini?

Done

Pas de valeur affichée, erreur ci-dessous

Pas de valeur affichée, mais pas d'erreur

La valeur de \$Mon_Nom a changé

- La définition des variables porte sur l'ensemble des blocs PHP

Variables PHP

- Supprimer des variables:
 - `unset($note);`
 - `unset($note,$coeff,$nom);`
- En PHP une variable peut-être:
 - définie avec une valeur, `$var="toto";`
 - définie et vide, `$var=""`;
 - non définie

PHP: Plan

- Ce que vous allez découvrir
- PHP et XHTML: premiers éléments
- Exécution conditionnelle
- Variables, **Types de Données**
- Chaînes
- Formulaire, vérification des champs
- Contrôle du flux d'exécution
- Fonctions
- Tableaux
- Ce qu'il faut retenir

Types de Données

- Entier:
 - `$note=1728; // Bravo c'est excellent!`
- Nombre à virgule flottante:
 - `$temp=37.2; // Le matin`
- Chaîne de caractères:
 - `$Mon_Nom="Est Personne!"`
 - `$Mon_Id="178087718400561" //chaîne pas entier!`
- Booléen:
 - `$condition=TRUE; $condition=FALSE;`

Types de Données

• Conversion

- `$entier=(int)$var2;`
- `$chaîne=(string)$var3;`
- `$flottant=(float)$var4;`
- Attention, on ne peut pas convertir n'importe quelle chaîne en entier (0 si impossible)
 - `$entier=(int)"1597" // OK entier = 1597`
 - `$entier=(int)"Je suis beau" // KO entier = 0`
 - `$entier=(int)"1597 Je suis beau" // entier = 1597`

Conversion: exemple simple

```
<?php
```

```
$entier1=4; $entier2=44;
```

```
$chaine="2007 une bonne annee!";
```

```
$string="une bonne annee 2007";
```

```
echo "entier1: ", $entier1, " entier2: ", $entier2;
```

```
echo "<br />chaine: ", $chaine, " string: ", $string;
```

```
$entier1=$chaine; $entier2=$string;
```

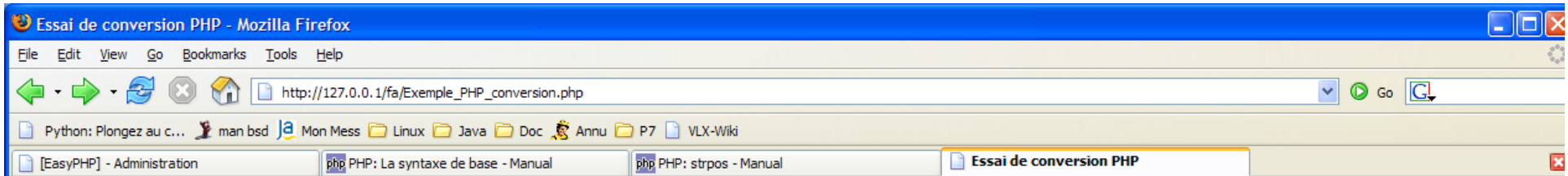
```
echo "<br />entier1: ", $entier1, "entier2: ", $entier2;
```

```
$entier1=(int)$chaine; $entier2=(int)$string;
```

```
echo "<br />entier1: ", $entier1, "entier2: ", $entier2;
```

```
?>
```

Conversion: résultat



Un essai de conversion

La variable 'entier1': 4 . La variable 'entier2': 44 .

La variable 'chaine' : "2007 une bonne annee!" et 'string': "une bonne annee 2007"

Sans conversion, 'entier1' vaut: "2007 une bonne annee!" et 'entier2' vaut: "une bonne annee 2007"

Avec conversion, 'entier1' vaut: 2007 et 'entier2' vaut: 0

- Sans conversion: on hérite valeur et type
- Avec conversion: on garde le type défini par la conversion

Calculs arithmétiques

- Opérateurs:

- $+ - * / \%$

- `$cpt++;` // équivalent à `$cpt=$cpt+1;`

- `$cpt--;` // équivalent à `$cpt=$cpt-1;`

- `$cpt/=2;` `$cpt+=77;` `$cpt-=2;` `$cpt*=5;`

- Quelques fonctions:

- `sqrt(64)` // => 8 racine carrée

- `ceil(27.68)` // => 28 entier immédiatement supérieur

- `floor(27.93)` // => 27 entier immédiatement inférieur

- `round(24.3)` // => 24; `round(24.6)` // => 25

PHP: Plan

- Ce que vous allez découvrir
- PHP et XHTML: premiers éléments
- Exécution conditionnelle
- Variables, Types de Données
- **Chaînes**
- Formulaire, vérification des champs
- Contrôle du flux d'exécution
- Fonctions
- Tableaux
- Ce qu'il faut retenir

Chaînes

```
$var="toto"
```

```
echo "Le nom de cette variable est $var<br />"
```

Le nom de cette variable est toto

```
echo 'Le nom de cette variable est $var<br />'
```

Le nom de cette variable est \$var

```
$str="Cette chaîne n'a qu'un \" guillemet";
```

```
$str='Cette chaîne " s\'écrit" avec une apostrophe';
```

Chaînes

- Concaténer des chaînes:

```
$str1="Hello";  
$str2="World!";  
$salut=$str1." ".$str2;  
echo $salut;  
Hello World!
```

- Supprimer des caractères "invisibles" (espaces)

- \$str_out= **trim**(\$chaine) // avant, après
- \$str_out= **ltrim**(\$chaine) // avant (gauche, left)
- \$str_out= **rtrim**(\$chaine) // après (droite, right)

Chaînes

- `str_repeat("xyz",5)`
 - `=>` `xyzxyzxyzxyzxyz`
- `str_replace("a","b",$str)`
 - Remplace toutes les occurrences de la chaîne a par la chaîne b dans la chaîne \$str
 - a et/ou b peuvent être des tableaux
 - `str_replace("e","a","ettention") =>` attention
- `strlen($str)`
 - Renvoie la longueur de la chaîne \$str
 - `strlen("ab de "); // 7`

Chaînes

- `strchr($tasdefoin, $aiguille)` ou `strstr(x, y)`;
 - Recherche la chaîne aiguille dans celle tasdefoin
 - Renvoie la chaîne commençant à la première occurrence, `FALSE` si `$aiguille` pas trouvée
 - `strchr("www.univ-paris-diderot.fr », ".");`
 - `=>.ufr-info-p7.jussieu.fr`
- `strrchr($tasdefoin, $aiguille)`
 - Idem, mais travaille sur la dernière occurrence
 - `strrchr("www.univ-paris-diderot.fr », ".");`
 - `=>.fr`

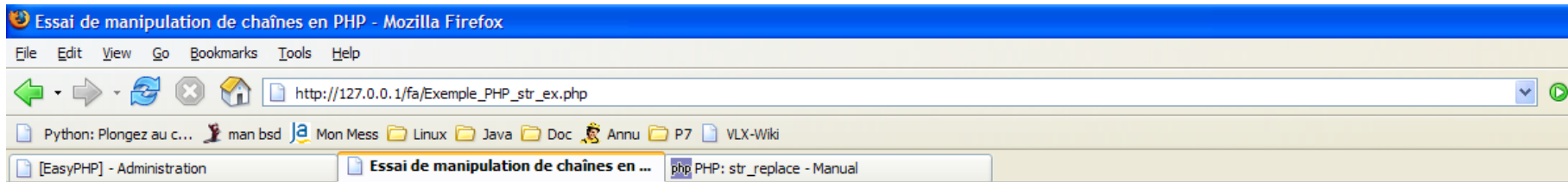
Chaînes

- `substr($str, debut [,fin])`
 - Renvoie le segment de chaîne \$str compris entre debut et fin.
 - `substr("abcdefgh", 2, 6);`
 - `=> cdef`
- `strcmp($str1, $str2)`
 - Renvoie 0 si les deux chaînes ont égales,
 - une valeur > 0 si `str1 > str2`
 - Une valeur < 0 si `str1 < str2`
 - `strcmp("abcde", "bcdef"); // -1`
 - `strcmp("abcde", "Abcde"); // 1`

Chaînes: Exemple

```
echo "Je suis en train d'exécuter le fichier ", __FILE__;  
$Chemin= str_replace("\\", "/", __FILE__);  
echo "<br />le fichier: ", substr(strrchr($Chemin, "/"), 1);  
$Fichier= trim(strrchr($Chemin, "/"), "/");  
echo "<br />le fichier: ", strtoupper($Fichier);  
$Comp= explode("/", $Chemin);  
echo "<br />le fichier:", $Comp[sizeof($Comp)-1];  
echo "<br />le fichier php: ", basename(__FILE__, ".php");  
echo " situé dans le répertoire: ", dirname($Chemin);
```

Chaînes: Exemple



Un essai de manipulation de chaînes

Je suis en train d'exécuter le fichier c:\documents and settings\francois\desktop\francois\p7\paris7\11_06_07\myeasyphp\exemple_php_str_ex.php
le fichier: exemple_php_str_ex.php
le fichier: EXEMPLE_PHP_STR_EX.PHP
le fichier: exemple_php_str_ex.php
le fichier php: exemple_php_str_ex situé dans le répertoire: c:/documents and settings/francois/desktop/francois/p7/paris7/11_06_07/myeasyphp
Et voilà!

PHP: Plan

- Ce que vous allez découvrir
- PHP et XHTML: premiers éléments
- Exécution conditionnelle
- Variables, Types de Données
- Chaînes
- **Formulaire, vérification des champs**
- Contrôle du flux d'exécution
- Fonctions
- Tableaux
- Ce qu'il faut retenir

Exemple de Formulaire

Exemple simple de formulaire - Mozilla Firefox

File Edit View Go Bookmarks Tools Help

http://127.0.0.1/fa/exemple_input.php

Python: Plongez au c... man bsd Mon Mess Linux Java Doc Annu P7 VLX-Wiki

[EasyPHP] - Administration PHP: Variables pré-définies - Manual Index of the HTML 4 Attributes Exemple simple de formulaire

Un formulaire

Les champs écrits en **rouge** doivent être obligatoirement renseignés.

Nom Armand **Prénom** François

Adresse 6 Place Jussieu
75006 Paris

E-mail armand@ufr-info-p7.jussieu.fr **Téléphone** 01 02 03 04 05

PAYS FR UK US BE

Pseudo: FA

Mot de passe ***** **Confirmez le** *****

Votre age

- 0-3 ans
- 3-12 ans
- 12-18 ans
- 18-29 ans
- 29-55 ans**
- 55-75 ans

Cinema Theatre

Done

Formulaire: Résultat

- Le traitement effectué consiste ici, à afficher, ce que le serveur a reçu.

```

Traitement du formulaire

Essai avec print_r
Array ( [nom] => Armand [prenom] => François [adresse] => 6 Place Jussieu 75006 Paris
[email] => FA [pays] => fr [passwd] => PASSEZ! [repasswd] => Passez [age] => 5 [go] =>
Envoi! )

Essai avec var_dump
array(9) { ["nom"]=> string(6) "Armand" ["prenom"]=> string(8) "François" ["adresse"]=>
string(28) "6 Place Jussieu 75006 Paris" ["email"]=> string(2) "FA" ["pays"]=> string(2) "fr"
["passwd"]=> string(7) "PASSEZ!" ["repasswd"]=> string(6) "Passez" ["age"]=> string(1) "5"
["go"]=> string(6) "Envoi!" }

Taille du tableau $_REQUEST: 9
$_REQUEST[nom]= Armand
$_REQUEST[prenom]= François
$_REQUEST[adresse]= 6 Place Jussieu 75006 Paris
$_REQUEST[email]= FA
$_REQUEST[pays]= fr
$_REQUEST[passwd]= PASSEZ!
$_REQUEST[repasswd]= Passez
$_REQUEST[age]= 5
$_REQUEST[go]= Envoi!

$_REQUEST[0]= Armand
$_REQUEST[1]= François
$_REQUEST[2]= 6 Place Jussieu 75006 Paris
$_REQUEST[3]= FA

```

Formulaire: Source XHTML

- Utilisation d'une table pour la mise en page
- Utilisation de la balise `<form>` et des balises `inputs`
- Utilisation de style pour:
 - Alignement
 - Champs obligatoires

Exemple de Formulaire

- Le même formulaire avec les délimitations de cellules visibles:

Exemple simple de formulaire - Mozilla Firefox

File Edit View Go Bookmarks Tools Help

http://127.0.0.1/fa/exemple_input.php

Python: Plongez au c... man bsd |a Mon Mess | Linux | Java | Doc | Annu | P7 | VLX-Wiki

[EasyPHP] - Administration | php PHP: Variables pré-définies - Manual | HTML TABLE TAG | Exemple simple de formulaire

Un formulaire

Les champs écrits en **rouge** doivent être obligatoirement renseignés.

Nom	Armand	Prénom	François
Adresse	6 Place Jussieu 75006 Paris		
E-mail	armand@ufr-info-p7.jussieu.fr	Téléphone	01 02 03 04 05
PAYS	FR <input checked="" type="radio"/> UK <input type="radio"/> US <input type="radio"/> BE <input type="radio"/>		
Pseudo:	FA		
Mot de passe	<input type="password"/>	Confirmez le	<input type="password"/>
Votre age	0-3 ans 3-12 ans 12-18 ans 18-29 ans 29-55 ans 55-75 ans		
<input type="button" value="Envoi!"/>		<input type="button" value="On recommence tout"/>	
Cinema <input type="checkbox"/> Theatre <input type="checkbox"/>			

Done

Formulaire: Source XHTML

```
<style type="text/css">
  td.droit {text-align:right}      /* Aligner a droite une case*/
  *.oblig {color:red}             /* Obligatoire */
  *.centre {text-align:center} /* Centrer le texte */
</style>
```

```
<form action="tst_action.php" method="get">
```

```
<table><tr>
```

```
<td class="droit oblig">Nom</td>
```

```
<td><input type="text" name="nom" size="30"
maxlength="60"/> </td>
```

```
<td class="droit oblig">Pr&eacute;nom</td>
```

```
<td><input type="text" name="prenom" size="30"
maxlength="60" /> </td>
```

```
</tr>
```

Le script PHP invoqué

Liste de classes

Formulaire: Source XHTML

- Le champs Adresse:

```
<td class="droit">Adresse</td>  
<td colspan="3">  
  <textarea cols="40" rows="4"  
    name="adresse">Votre adresse postale.  
  </textarea>  
</td>
```

Formulaire: Source XHTML

```
<tr>
  <td class="oblig">Pseudo:</td>
  <td colspan="3"><input type="text" name="email"
    size="80" /> </td>
</tr><tr>
  <td class="oblig">Mot de passe</td>
  <td><input type="password" name="passwd" size="16"
    /> </td>
  <td class="oblig">Confirmez le</td>
  <td><input type="password" name="repasswd"
    size="16" /> </td>
</tr>
```

Formulaire: Source XHTML

- Fin du formulaire:

```
<td colspan="2" style="text-align:center">  
  <input type="submit" value="Envoi!" name="go" />  
</td>  
<td colspan="2" style="text-align:center">  
  <input type="reset" value="On recommence tout" />  
</td>  
</table>  
</form>  
Cine <input type="checkbox" name="loisir" value="C" />
```

Script PHP: Exemple

```
<?php
echo "Essai avec print_r <br />";
print_r($_REQUEST);
echo "<hr />"; echo "Essai avec var_dump <br />";
var_dump($_REQUEST);
echo "<hr />";

echo 'Taille du tableau $_REQUEST: ',
count($_REQUEST) , '<br />';

foreach ($_REQUEST as $index => $name) {
    echo '$_REQUEST[',$index,']= ', $name,"<br />";
}
```

Script PHP: Exemple

```
echo "<hr />";
```

```
for (reset($_REQUEST), $i=0;  
     current($_REQUEST);  
     next($_REQUEST), $i++) {
```

```
    echo '$_REQUEST[',$i,']= ".current($_REQUEST);  
    echo "<br />";
```

```
}  
?>
```

Vérification des champs

Exemple simple de formulaire et de script php - Mozilla Firefox

Fichier Edition Affichage Aller à Marque-pages Outils ?

http://127.0.0.1/L1%20Essais%20PHP/Exemple_

man bsd Mon Mess Linux Java Doc Annu

[EasyPHP] - Administration Exemple simple de formulaire et...

Un formulaire

Les champs écrits en **rouge** doivent être obligatoirement renseignés.

Nom **Prénom**

Adresse

E-mail **Téléphone**

PAYS FR UK US BE

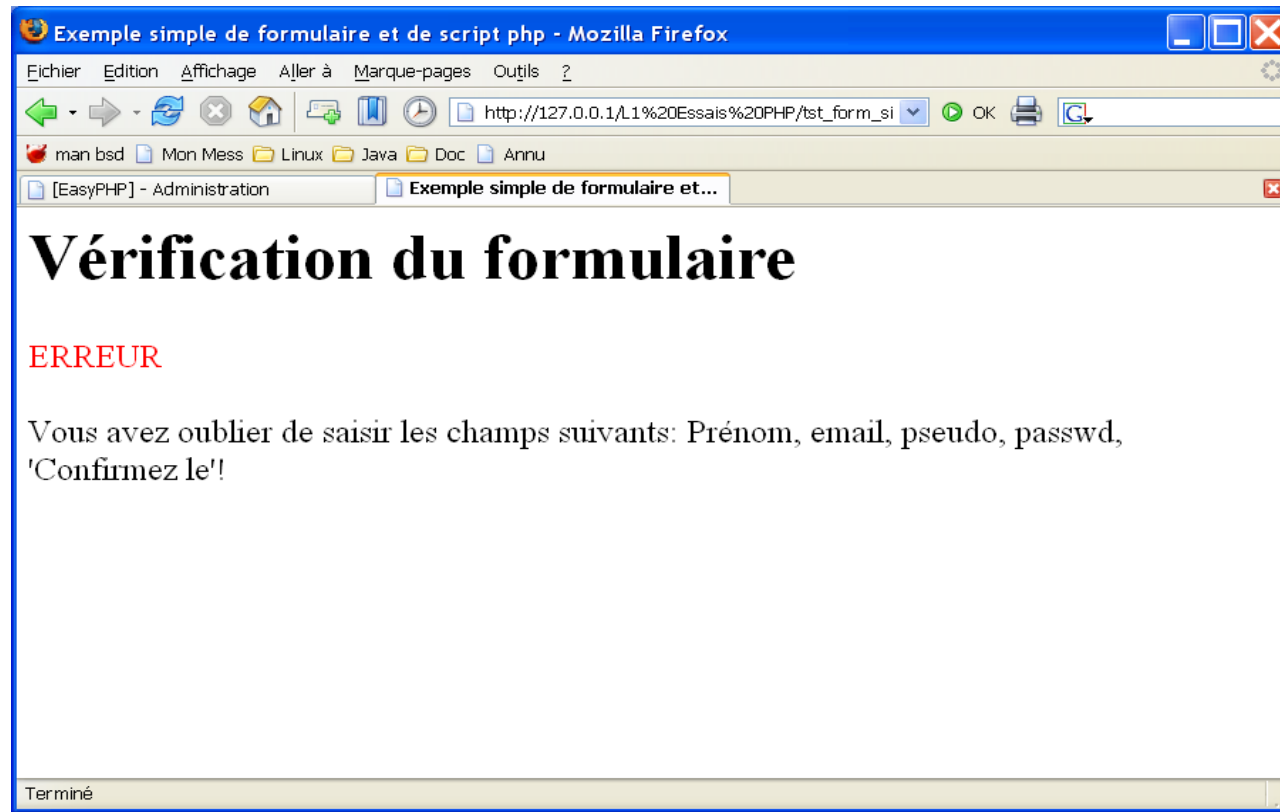
Pseudo:

Mot de passe **Confirmez le**

Votre age
0-3 ans
3-12 ans
12-18 ans
18-29 ans
29-55 ans

Terminé

Affichage d'une page d'erreur



Script PHP de vérification

- Définition des variables

```
define ("ENVOI", "Envoi!");
```

```
$ChampsManquants=""; $nom=""; $prenom="";  
$adresse=""; $email=""; $phone=""; $pseudo="";  
$passwd=""; $repasswd=""; $pays="";
```

- Pour chacune des valeurs du formulaire:

```
if (isSet($_GET['nom'])) {  
    $nom=$_GET['nom'];  
}
```

Script PHP de vérification

```
/* Verifier les champs obligatoires */
if (empty($nom)) { /* Le nom n'est pas fourni */
    /* Ajouter le nom du champ dans la liste des manquants */
    if (!empty($ChampsManquants)){
        /* Commencer par une , si nécessaire */
        $ChampsManquants=$ChampsManquants.", " ;
    }
    $ChampsManquants=$ChampsManquants."Nom";
}
if (strlen($prenom) == 0) {
    /* Code manquant pour lisibilité */
    $ChampsManquants= $ChampsManquants."Prénom";
}
```

Script PHP de vérification

- Afficher et renvoyer une erreur:

```
if (!empty($ChampsManquants)) {  
    echo '<div class="oblig">ERREUR</div><br />';  
    exit("Vous avez oublier de saisir les champs suivants:  
    ".$ChampsManquants.'!');  
}
```

PHP: Plan

- Ce que vous allez découvrir
- PHP et XHTML: premiers éléments
- Exécution conditionnelle
- Variables, Types de Données
- Chaînes
- Formulaire, vérification des champs
- **Contrôle du flux d'exécution**
- Fonctions
- Tableaux
- Ce qu'il faut retenir

Boucles

- `for (initialisation; condition; fin d'itération) {
 bloc
}`
- `while (condition) { bloc; }`
- `do { bloc; } while (condition);`
- Contrôle de boucle:
 - `continue; // passe à l'itération suivante`
 - `break; // sort de la boucle`

Contrôle du flux d'exécution

```
if (condition) {  
    bloc;  
} else {  
    bloc;  
}
```

```
switch ($var) {  
    case Valeur1: bloc; break;  
    case Valeur2: bloc; break;  
    default: bloc;  
}
```

Contrôle du flux d'exécution

- Conditions:
 - `$v1 == $v2` // les 2 valeurs sont-elles égales?
 - `$v1 === $v2` // même valeur et même type?
 - `$v1 != $v2` // valeur différentes?
 - `$v1 !== $v2` // valeur ou type différents?
 - `>` , `<` , `>=` , `<=`

PHP: Plan

- Ce que vous allez découvrir
- PHP et XHTML: premiers éléments
- Exécution conditionnelle
- Variables, Types de Données
- Chaînes
- Formulaire, vérification des champs
- Contrôle du flux d'exécution
- **Fonctions**
- Tableaux
- Ce qu'il faut retenir

Script PHP de vérification

```
/* Verifier les champs obligatoires */
```

```
if (empty($nom)) { /* Le nom n'est pas fourni */  
/* Ajouter le nom du champ dans la liste des manquants */  
if (!empty($ChampsManquants)){  
/* Commencer par une , si nécessaire */  
$ChampsManquants=$ChampsManquants.", " ;  
}  
$ChampsManquants=$ChampsManquants."Nom";  
}
```

```
if (strlen($prenom) == 0) {  
/* Code manquant pour lisibilité */  
$ChampsManquants= $ChampsManquants."Prénom";  
}
```

Fonctions

- Exemple:
 - La validation des paramètres reçus est faite à l'identique pour chaque paramètre
 - Les seules différences: paramètre, nom du champ
 - Cf: code encadré dans la page précédente
- Eviter de ré-écrire du code identique:
 - Gagner du temps
 - Eviter des erreurs
- => Utiliser une fonction

Définition / Utilisation de fonction

- Définition de la fonction

```
function verif_champ($valeur, $champ, $liste)
{
    if (empty($valeur)) {
        if (!empty($liste)) {
            $liste = $liste.", ";
        }
        $liste = $liste.$champ;
    }
    return $liste;
}
```

- Appel (utilisation) de la fonction:

```
verif_champ($email, "E-mail", $ChampsManquants);
verif_champ($pays, "Pays", $ChampsManquants);
```

Fonctions

```
function nom_de_ma_fonction($arg1, $arg2...)  
{  
    $ma_var_locale=17890;  
    Bloc; // Par exemple sortie html de haut de page  
    return $resultat;  
}
```

- Les variables internes à la fonction ne sont pas visibles de l'extérieur de la fonction
- Les variables externes ne sont pas visibles dans la fonction (sauf "global \$var")

Global

```
function verif_champ($valeur, $champ)
{
    global $ChampsManquants;
    if (empty($valeur)) {
        if (!empty($ChampsManquants)) {
            $ChampsManquants = $ChampsManquants.", ";
        }
        $ChampsManquants = $ChampsManquants.$champ;
    }
}
```

- Si pas « global »

- Notice: Undefined variable: ChampsManquants in c:\documents and settings\francois\desktop\francois\p7\paris7\l1_06_07\myeas yphp\tst_form_simple_verif.php on line 48

Inclusion

- **include** ("monfichier.inc");
 - Le fichier inclus doit contenir les balises `<?php..?>`
 - `<?php`
define ("MA_CSTE", 3);
define ("MA_COULEUR", "rouge")
?>
- **include_once**:
 - Évite les inclusions multiples inutiles ou dangereuses.

PHP: Plan

- Ce que vous allez découvrir
- PHP et XHTML: premiers éléments
- Exécution conditionnelle
- Variables, Types de Données
- Chaînes
- Formulaire, vérification des champs
- Contrôle du flux d'exécution
- Fonctions
- **Tableaux**
- Ce qu'il faut retenir

Tableaux

- Créer avec "index":
 - `$Films[0]="Shrek";`
 - `$Films[1]="Shrek 2";`
 - Pas d'obligation de commencer à 0
- Créer avec "clefs":
 - `$capitale['FR']="Paris";`
 - `$capitale['UK']="Londres";`
 - `$capitale['DE']="Berlin";`

Tableaux

- Créer sans index explicite:
 - `$Livre[]="Les Misérables";`
 - `$Livre[]="Harry Potter";`
 - `$Livre[]="Le seigneur des anneaux";`
- `echo "Livre[0]";`
Les Misérables
- `$Livre= array ("Les Misérables", "Harry Potter", "Le seigneur des anneaux");`

Tableaux

- `$rue= array(4 => "Rue de Lapp", "Rue de la Paix");`
- `echo $rue[5]`
 - Rue de la Paix
- `$annees= range(2001, 2038);`
- On peut aussi créer des tableaux multi-dimensionnels
 - `$Notes['Jules'] ['Exo 1']; $Notes['Jules'] ['Exo 2'];`
 - `$Notes['Jim'] ['Exo 1']; $Notes['Jim'] ['Exo 2'];`

Tableau: Parcourir

```
foreach($Livre as $index => $Titre)  
{  
    echo "Livre Numéro ",$index," : $Titre<br/>";  
};
```

Livre Numéro 0 : Les Misérables

Livre Numéro 1 : Harry Potter

Livre Numéro 2 : Le seigneur des anneaux

PHP: Plan

- Ce que vous allez découvrir
- PHP et XHTML: premiers éléments
- Exécution conditionnelle
- Variables, Types de Données
- Chaînes
- Formulaire, vérification des champs
- Contrôle du flux d'exécution
- Fonctions
- Tableaux
- **Ce qu'il faut au moins retenir**

Ce qu'il faut au moins retenir

- PHP est un langage interprété côté serveur
- Le code PHP est remplacé par le résultat de l'exécution du code PHP
- On ne peut pas "voir" le code PHP dans le source de la page affichée par le navigateur (côté client)
- Variables : \$xxx, entier, flottant, chaîne, booléen
- Variables prédéfinies

Ce qu'il faut au moins retenir

- L'accès aux champs saisis par l'utilisateur dans un formulaire se fait par accès au tableau `$_REQUEST`
- Contrôle de flux d'exécution classique:
 - `if {} else;`
 - `switch()`
 - `for (...)`
 - `while ()`
- Manipulation des tableaux

La Semaine Prochaine

- PHP avancé
 - Constantes
 - Erreurs,
 - Tableaux, classes,
 - Accès aux fichiers,
 - Upload de fichiers
 - Download de fichiers