$implonBoulogne = PHP;

Basiques

Sommaire

- Données
- Variables
- Opérateurs
- Fonctions
- Arrays

Introduction

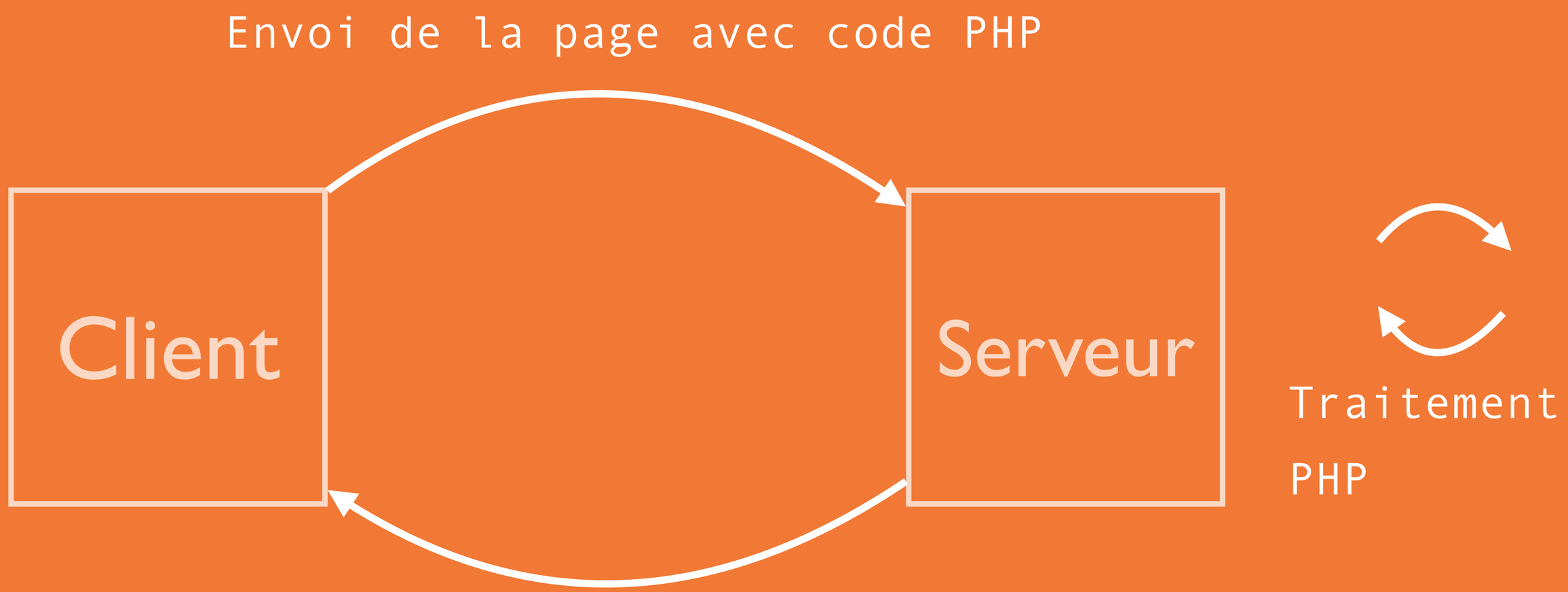
Inventé par Rasmus Lerdorf en 1994: voulait faire le suivi des visiteurs sur son site web.

PHP: Hypertext Preprocessor, c'est une solution Open Source.

À l'heure d'aujourd'hui, la dernière version stable est PHP 5.6.8, elle est recommandé pour tous les débutants.

La dernière Release de PHP a été dévoilé : PHP 7.0

Introduction : Côté serveur



Introduction : Initialisation

Il faut initialiser chaque morceau de code contenant du PHP avec des balises spécifiques.

```
<?php
    $mystring = "12";
    $myinteger = 20;
?>
```

Introduction : Initialisation

Vous pouvez inclure des fichiers php externes.

```
<?php
    include("monfichier.php");
?>
```

Données

Types de données

String	« abc »	« Vive le PHP »
Integer	30	-20
Floats	4.2	1.00023
Booleans	true	false
Arrays	variable spéciale	
Objects	variable + complexe	

Données  Forcer un type

PHP va automatiquement convertir les types des données autant que nécéssaire, mais si besoin vous pouvez spécifier un type précis pour passer au dessus du type définit par défaut avec PHP.

```php
<?php
    $myString = "Vive les formateurs";
    $myInteger = (integer)$myString;
    var_dump($myInteger)
?>
```

Variables  Variables set

isset() sert à vérifier si une variable est bien définie or not. Très utilisé pour la suite de vos projets. Permet également d'éviter les warning avec PHP !

```
$petiteVar = 1;
if (isset($petiteVar)) {
    echo "Définie  \n";
} else {
    echo "Pas définie… \n »;
}
```

Variables  Constantes

Une constante est une variable qui restera inchangée car elle ne peut être modifiée plus tard dans le code contrairement aux variables.

Une constante prend deux paramètres : un nom, et une valeur

```
<?php
  define(‘YEAR’,’2014’);
  define("CURRENT_TIME", time());
?>
```

Variables : SUPERGLOBALES

$_GLOBALS	Toutes les variables globales dont les variables d'environnements
$_GET	Envoyées avec une requêtes GET
$_POST	Envoyées avec une requêtes POST
$_FILES	Envoyées avec une requêtes pour l'upload de fichiers
$_COOKIE	Stocke les variables de cookies
$_SESSION	Stocke les variables de sessions
$_ENV	Stocke les variables d'environnements

Opérateurs  Comparaisons

$a == $b	Egaux
$a === $b	Identique
$a != $b	Pas égaux
$a <> $b	Pas égaux
$a !== $b	Pas identique
$a < $b	Plus petit
$a > $b	Plus grand
$a <= $b	Plus petit ou égal
$a >= $b	Plus grand ou égal

Fonctions  var_dump()

var_dump() est un bon moyen de voir ce qu'il y a à l'intérieur d'une variable.

C'est un bon réflexe à prendre en main pour debugger notamment !

```
<?php
    $capitalcities['England'] = array("Capital"=>"London",
"Population"=>40000000, "NationalSport"=>"Cricket");
    var_dump($capitalcities)
?>
```

Conditions & boucles

```php
<?php
if ($a > $b){
  echo "a est plus grand que b »;
}
?>
```

Conditions & boucles

```
if ($a > $b) {
    echo "a est plus grand que b";
} elseif ($a == $b) {
    echo "a est égal à b";
} else {
    echo "a est plus petit que b";
}
```

Conditions & boucles

```
for ($i = 1; $i <= 10; $i++) {
    echo $i;
}
```

Fonctions PHP

Fonction utilisateur

```
<?php
    function sayHello($name) {
        return "Hello $name";
    }

    print sayHello("Maxime");
?>
```

Fonctions PHP

Paramètres par défaut

```php
<?php
    function coucou($name = "Maxime") {
        return "Hey $name!\n";
    }

    coucou();
    coucou("Morgan");
    coucou("Enzo");
?>
```

Arrays

Tableaux associatifs

```
<?php
    $myarray = array("a"=>"Pommes", "b"=>"Oranges",
"c"=>"Poires");
    var_dump($myarray);
?>
```

On appelle tableau associatif la relation clé - valeur. Vous pouvez spécifier à chaque fois un nom de clé spécifique (sauf de type float)

Arrays

Ranger des tableaux

sort()	Tri les éléments dans l'ordre alphabétique - Réécrit les clés
rsort()	Tri les éléments dans l'ordre inverse - Réécrit les clés
asort()	Tri les éléments dans l'ordre alphabétique - Garde les clés
arsort()	Tri les éléments dans l'ordre inverse - Garde les clés
ksort()	Tri les clés dans l'ordre alphabétique
krsort()	Tri les clés dans l'ordre inverse

Arrays

Ranger des tableaux

```
array(3) {
    ["a"]=>
    string(6) "Pommes"
    ["b"]=>
    string(7) "Oranges"
    ["c"]=>
    string(5) "Poires"
}
```

Arrays

Tableaux associatifs

```
<?php
    $myarray = array("a"=>"Pommes", "b"=>"Oranges",
"c"=>"Poires");
    var_dump($myarray);
?>
```

On appelle tableau associatif la relation clé - valeur. Vous pouvez spécifier à chaque fois un nom de clé spécifique (sauf de type float)

Arrays

Tableaux multidimensionnels

```
<?php
    $capitalcities['England'] = array("Capital"=>"London",
"Population"=>40000000, "NationalSport"=>"Cricket");
    $capitalcities['France'] = array("Capital"=>"Paris",
"Population"=>2440000, "NationalSport"=>"Football");
?>
```

Arrays

Tableaux multidimensionnels

```
array(2) {
    ["England"]=>
    array(3) {
        ["Capital"]=>
        string(6) "London"
        ["Population"]=>
        int(40000000)
        ["NationalSport"]=>
        string(7) "Cricket"
    }
    ["France"]=>
    array(3) {
        ["Capital"]=>
        string(7) "Paris"
        ["Population"]=>
        int(2440000)
        ["NationalSport"]=>
        string(5) "Football"
    }
```

Arrays

Sauvegarder un array

```
$array["a"] = "Oh";
$array["b"] = "Ha";
$array["c"] = "Hey";
      // serialize un array, permet son enregistrement dans un fichier
par exemple
$str = serialize($array);
      // encode pour le convertir en une version safe pour le passer
sur le web
$strenc = urlencode($str);
```

Arrays

Sauvegarder un array
- suite

```php
// decode
$arr = unserialize(urldecode($strenc));
var_dump($arr);
```

Erreurs PHP

Afficher les erreurs en local

```
// Afficher les erreurs à l'écran
ini_set('display_errors', 1);
// Enregistrer les erreurs dans un fichier de log
ini_set('log_errors', 1);
// Nom du fichier qui enregistre les logs (attention
aux droits à l'écriture)
ini_set('error_log', dirname(__file__) . '/
log_error_php.txt');
```

Challenge 

Créer une variable qui contient des noms de jeux vidéos.

Écrire une fonction qui affiche une ligne de tableau contenant un seul nom de jeu vidéo passé préalablement en paramètre de la fonction.

Compléter ensuite votre code avec une boucle qui affichera à chaque tour votre fonction pour finalement obtenir tous les noms des jeux.

(1 nom de jeu = 1 ligne de tableau)

Utilisation de Bootstrap pour la forme et un plug-in jQuery (datatables.js) pour un tri dynamique dans le tableau.

Some ressources

- http://www.hackingwithphp.com/
- https://secure.php.net/manual/fr/index.php