

LE LANGAGE PHP

- AUTEUR → BRUNO DIÉMERT
- SCÉNARISTE → LUCILE CLAUSEL
- PILOTAGE → MAMAGALUBRU
- LICENCE → CREATIVE COMMONS



PLAN DU COURS

- Introduction
- Le modèle client-serveur
- Exécuter une page PHP
- Convention d'écriture
- Créer une page PHP
- Les constantes et les variables
- Les types de données
- Les opérateurs PHP
- Logique conditionnelle
- Logique de boucle
- Les fonctions
- Quelques fonctions intégrées



INTRODUCTION



- Qu'est-ce que PHP ?
 - Hypertext PreProcessor est un langage internet de script inclus dans du code HTML s'exécutant du côté serveur spécialement conçu pour le développement WEB par Rasmus Lerdorf en 1994 pour des besoins personnels (rapide et facile)
- Langages de Scripts
 - Script client : code de programmation présent dans un fichier HTML, lui-même affiché dans un navigateur (par exemple JavaScript)
 - Script serveur : script exécuté sur le serveur qui transmet son résultat au client (par exemple PHP)

DIFFÉRENTES VERSIONS DE PHP

- De nombreuses versions de PHP depuis 1995
 - PHP 1.0 en 1994 (facile)
 - PHP 2.0 en 1996 (forte médiatisation)
 - PHP 3.0 en 1997 (rapide)
 - PHP 4.0 en 2002 (sécurisation)
 - PHP 5.0 en 2004 (adaptation à l'objet)
 - Branches supportées
 - PHP 5.4 juste maintenue en sécurité
 - PHP 5.5 et 5.6 maintenue en débogage et sécurité
 - PHP 5.7 sortie en décembre 2015
- Évolution et médiatisation très rapide du langage et succès immédiat grâce à sa simplicité
- Une maturité certaine (sécurité, objet, ...) atteinte depuis la version 5

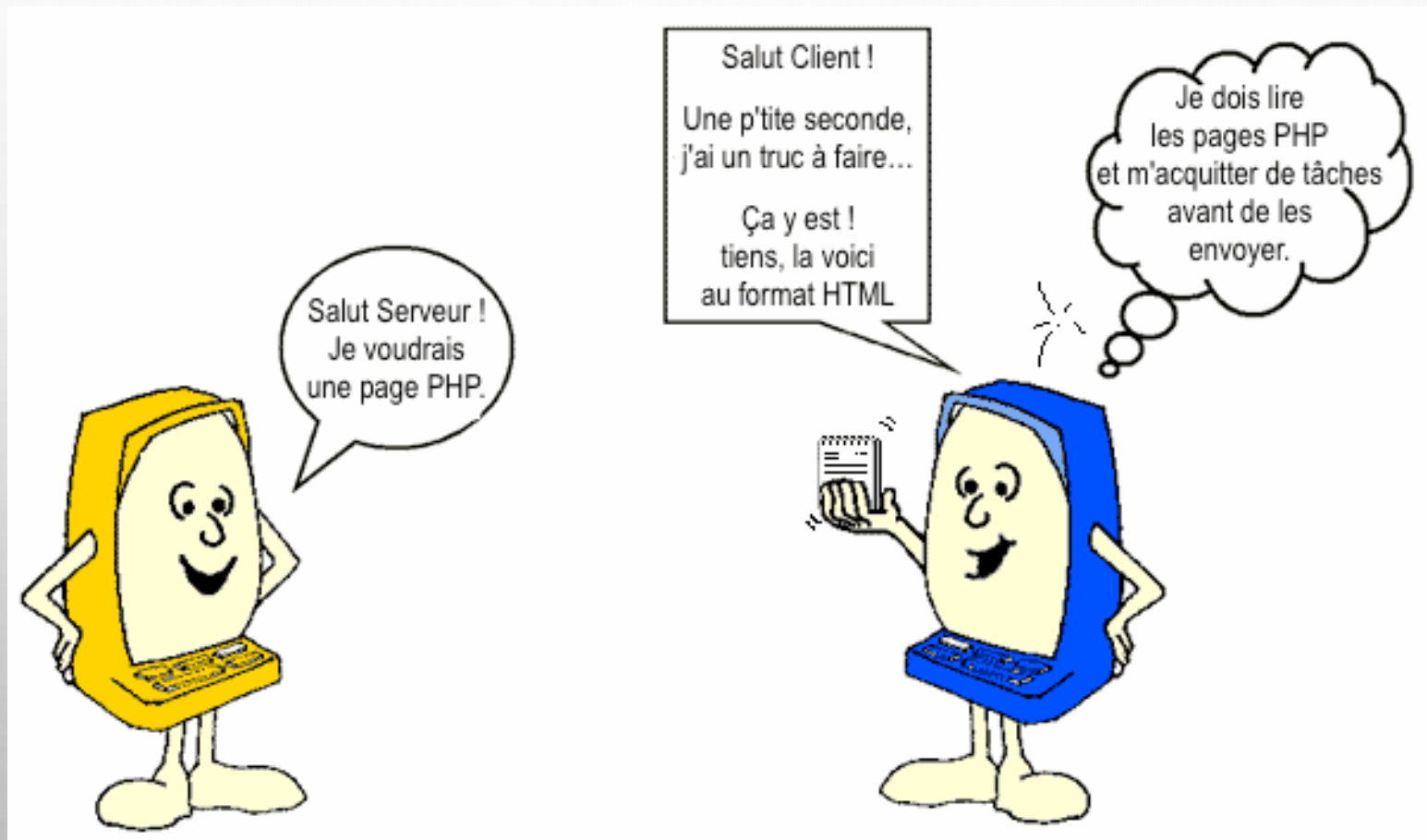


INTÉGRATION AU HTML

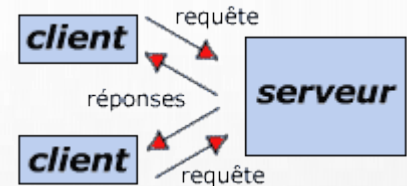


```
<html>
  <head>
    <title>Test PHP</title>
  </head>
  <body>
    <?php echo '<p>Bonjour le monde</p>'; ?>
  </body>
</html>
```

LE CLIENT ET LE SERVEUR



LE MODÈLE CLIENT-SERVEUR



- Demande et affichage d'une page statique
 - Le client demande une simple page existante contenant du code HTML et éventuellement JS
 - Le serveur renvoie directement la page demandée
- Affichage d'une page dynamique
 - Le client demande une page existante contenant non seulement du code HTML JS mais aussi du code PHP
 - Le serveur HTTP fait exécuter le code PHP de la page
 - Le serveur PHP renvoie le résultat de la page demandée

EXÉCUTER DES PAGES PHP

- un serveur Lamp ou Wamp ou Mamp
 - un système d'exploitation soit Linux, soit Windows, soit Mac
 - un serveur http Apache
 - un serveur de données Mysql
 - un serveur de code Php
- un environnement Wamp ou Lamp ou Mamp
 - code dans dossier racine apache → www
 - démarrage des différents serveurs
 - appel de l'url localhost dans un navigateur



CONFIGURER LE SERVEUR PHP

PHP Version 5.3.13



System	Windows NT LONGTARIN2 6.1 build 7601 (Windows 7 Home Premium Edition Service Pack 1) i586
Build Date	May 8 2012 18:48:39
Compiler	MSVC9 (Visual C++ 2008)
Architecture	x86
Configure Command	cscrip /nologo configure.js "--enable-snapshot-build" "--disable-isapi" "--enable-debug-pack" "--without-mssql" "--without-pdo-mssql" "--without-pi3web" "--with-pdo-oci=C:\php-sdk\oracle\instantclient10\sdk,shared" "--with-oci8=C:\php-sdk\oracle\instantclient10\sdk,shared" "--with-oci8-11g=C:\php-sdk\oracle\instantclient11\sdk,shared" "--enable-object-out-dir=../obj/" "--enable-com-dotnet" "--with-mcrypt=static" "--disable-static-analyze"
Server API	Apache 2.0 Handler
Virtual Directory Support	enabled
Configuration File (php.ini) Path	C:\Windows
Loaded Configuration File	I:\wamp\bin\apache\apache2.2.22\bin\php.ini

- php.ini
- phpinfo

LIGNE DE COMMANDE PHP

```
I:\wamp\bin\php\php5.3.13>php -v
PHP 5.3.13 (cli) (built: May 8 2012 18:50:09)
Copyright (c) 1997-2012 The PHP Group
Zend Engine v2.3.0, Copyright (c) 1998-2012 Zend Technologies
    with Xdebug v2.2.0, Copyright (c) 2002-2012, by Derick Rethans
```

```
I:\wamp\bin\php\php5.3.13>php bonjour.php Bruno
Bonjour Bruno !
```

```
<?php
// afficher un simple message en utilisant un paramètre passé sur la ligne de commande
echo "Bonjour $argv[1] !\n";
?>
```

QUIZ



- Le créateur du langage PHP est Rasmus... ?
- Marier les versions du langage (1, 3, 5, 7) avec leurs années de création (2004, 2015, 1994, 1997)
- PHP est un langage informatique compilé (Oui/Non) ?
- Le langage PHP est utilisable (en client serveur, en client mobile, en client embarqué, en client lourd, en client léger, en client voyageur, ...)
- Trier par ordre d'importance (1, 2, 3, 4, 5, ...), les qualités du langage PHP (facile, rapide, sécurisé, médiatisé, objet, ...)

CRÉER UNE PAGE PHP

- Les éditeurs de pages

- Avantages : rapide, sobre, disponible, gratuit
- Inconvénients : peu d'outils de projet ou de debug



- Les balises utilisées par PHP dans la page HTML

- `<? ... ?>`
- `<?PHP ...?>`
- `<SCRIPT language='php'> ... </Script>`



CONVENTIONS D'ÉCRITURE

- Le nommage → (\$) _ ou lettre puis lettres ou chiffres
- La sensibilité à la casse
 - \$maVar n'est pas la même variable que \$MAVar
- Mise en commentaire
 - // sur une ligne
 - /*...*/ sur plusieurs lignes
- Instruction PHP
 - Contient
 - des variables et des constantes
 - des opérateurs et des structures de contrôles
 - des appels de fonctions
 - Se termine par un point-virgule

```
<?php
include 'fichier-a-inclure.php';
echo "Ce message s'affiche normalement";
?>
```

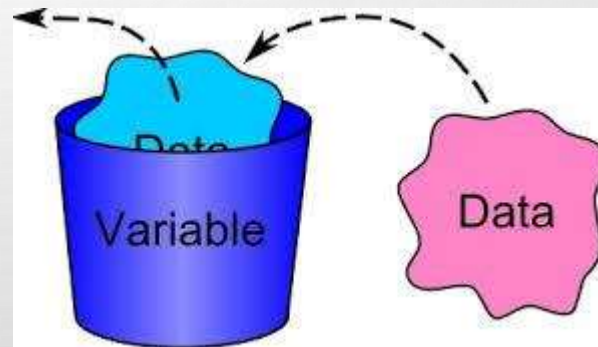
DÉFINITION DES CONSTANTES

- Une constante est une zone mémoire identifiée par un nom qui contient une valeur lisible mais non modifiable dans le programme
- Une constante PHP est composée de lettres, de chiffres et/ou de tirets du bas, par convention en majuscules
- La fonction `define` ou le mot-clé `const` permettent de définir une constante
 - `const VERSION = 'PHP 5.3';`
 - `define(VERSION, 'PHP 5.3');`
 - `define("TVA", 20);` ou `CONST TVA = 20`

DÉFINITION DES VARIABLES




- Une variable est une zone mémoire identifiée par un nom qui contient une valeur lisible et modifiable dans le programme
- Une variable PHP est un identifiant PHP composé de chiffres, de lettres et/ou de tirets du bas, le nom de la variable est toujours précédé du signe \$

```
$a='php';  
echo "I code $a";
```



LES TYPES DE DONNÉES SIMPLES

- Les nombres entiers
 - `$entier = 5 ;`
- Les nombres à virgule
 - `$reel = 5.15 ;`
- Les chaînes de caractères
 - `$chaine = 'bonjour' ;`
- Les booléens
 - `$booleen = TRUE ;`
- Les dates
 - `$date = "10/05/2021" ;`

Type de données	Exemple de valeur
string	"Du texte"
int	42
float	14.738
bool	true  false 
NULL	

LES TABLEAUX

```
<?php
// Déclaration d'un tableau vide
$fruits = array();

// Déclaration d'un tableau indexé numériquement
$legumes = array('carotte', 'poivron', 'aubergine', 'chou');

// Déclaration d'un tableau associatif
$identite = array(

    'nom' => 'Hamon',
    'prenom' => 'Hugo',
    'age' => 19,
    'estEtudiant' => true
);

?>
```

LES OBJETS

```
class SimpleClasse
```

```
{  
    private $var = 'une valeur par défaut';           // déclaration d'une propriété  
    public function displayVar() {                     // déclaration des méthodes  
        echo $this->var; }  
}
```

```
<?  
// la construction d'un objet  
class Personne {  
    var $nom, $prenom;  
    var $age;  
  
    function Personne ($n, $p, $a) {  
        $this->nom = $n;  
        $this->prenom = $p;  
        $this->age = $a;  
    }  
  
    // l'instanciation de l'objet  
    $p = new Personne("Toto", "Jean", 36, 1500);  
    // l'utilisation de l'objet  
    echo "L'âge de $p->prenom $p->nom est $p->age ans."  
?>
```

PORTÉE DES VARIABLES



Variable globale

- Les variables globales sont visibles sur tout le script PHP → **global**
`$VisiblePartout;`
- Les variables super globales sont visibles aussi sur tout le script PHP →
`$_POST['name'];`

Variable locale

- Les variables locales sont visibles uniquement à l'intérieur de la fonction qui les utilise → **non déclarées**

Variable statique

- Les variables statiques permettent de conserver les valeurs générées au sein d'une fonction d'un appel à l'autre → **static \$a = 0;**

TEST DES VARIABLES

Fonctions de typage

- `settype(nomDeLaVariable, "type")` → `settype($i, "integer");`
- `gettype(nomDeLaVariable)` → `gettype($i);`

Fonctions de test

- `isset($var)`
- `empty($var)`
- `unset($var)`
- `is_array($var)`
- `is_double($var), is_long($var)`
- `is_float($var), is_real($var),`
- `is_int($var), is_integer($var),`
- `is_string($var), is_object($var)`

Fonction isset()

```
<?php

if isset($_GET['param1']);
    echo $_GET['param1'];
else
{
    echo 'Paramètre non donné';
}

?>
```


LES OPÉRATEURS PHP

- Opérateur d'affectation → `$maVariable=1`
- Opérateur de concaténation → `$s = "Bonjour " . "à tous";`
- Opérateur d'assignation → `$monTableau=array("nom"=>"Bee" ,
"prenom" =>"Gee");`
- Opérateurs mathématiques
 - `+`, `-`, `*`, `/`, `\` (division des entiers, le résultat est un entier), `mod` (renvoie le reste d'une division) → `$maVariable=2+5;`
- Opérateurs de comparaison
 - `==` (égalité), `>`, `<`, `>=`, `<=`, `!=` et `<>` (différence) et `===` (identique en valeur et en type de données) → `if $maVariable==1`

AUTRES OPÉRATEURS PHP

- Opérateurs logiques → `if !feof($fp) && !$i<>0`
 - `$a AND $b` : Vrai si `$a` et `$b` sont tous deux vrais (idem `&&`)
 - `$a OR $b` : Vrai si l'un ou l'autre de `$a` et `$b` est vrai (idem `||`)
 - `NOT $a` : Vrai si `$a` est faux (idem `!`)
- Opérateur de référence → `$a=5; $b=&$a; //$b`
- Opérateur de suppression d'erreur → `$a=@(450/0);`
- Opérateur d'exécution externe → `$out=`c:\dir`;`

CONDITIONNELLE SIMPLE

Instruction If

```
If (condition) {  
    bloc d'instructions php;  
}
```

Instruction If ... Else

```
If (condition) {  
    bloc d'instructions php;  
}  
Else{  
    bloc d'instructions php;  
}
```

CONDITIONNELLE COMPLEXE

Instruction If ... Elsf ...

```
If (condition1) {  
    bloc d'instructions php;  
}  
Elsif(condition2){  
    bloc d'instructions php;  
}  
Else{  
    bloc d'instructions php;  
}
```

CONDITIONNELLE À SWITCH

- Switch valeur {
 Case v1 : bloc d'instructions php;
 Case v2 : bloc d'instructions php;
 Default : bloc d'instructions php;
}

```
Switch $choix{  
    case "A" :  
        {echo "Bon choix ".$choix;}  
    case "B" :  
        {echo "Mauvais choix ".$choix;}  
    default :  
        {echo "Absence de choix";}  
}
```

BOUCLE POUR

```
For ($i=valeurdepart; valeurfin; increment) {  
    bloc d'instructions php;  
}
```

```
For($i=10;$i<100;$i++){  
    // $i varie de 10 à 100 avec un pas de 1  
    $compteur=$i*$i;  
}
```


BOUCLE TANT QUE

```
While (condition) {  
    bloc d'instructions php;  
}
```

```
<?php  
$file = fopen("test.txt", "r");  
//lit et affiche chaque ligne du fichier  
while(! feof($file))  
{  
    echo fgets($file). "<br />";  
}  
fclose($file);  
?>
```

LES FONCTIONS

```
function nomDeLaFonction () {  
    bloc d'instructions php ;  
}
```

```
<?php  
function servir_cafe ($type = "cappuccino")  
{  
    return "Servir un $type.\n";  
}  
echo servir_cafe();  
echo servir_cafe(null);  
echo servir_cafe("espresso");  
?>
```

LES ARGUMENTS DE FONCTIONS

```
function nomDeLaFonction ($arg1,$arg2,...)    {  
    bloc d'instructions php                ;  
}
```

```
function message($msg)  
{  
    echo 'bonjour'.$msg;  
}
```

```
function somme($a,$b) {  
    $resultat = $a + $b ;  
    return $resultat ;  
}
```

QUELQUES FONCTIONS INTÉGRÉES

- Ce sont des fonctions prédéfinies qui permettent de simplifier la tâche du programmeur.
En voici 2 indispensables pour travailler en mode console.

```
<?php
/*
    afficher un simple message en utilisant un
    paramètre passé sur la ligne de commande
*/
echo "Bonjour $argv[1] !\n";
?>
```

```
<?php
setlocale(LC_CTYPE, 'fr_FR.UTF-8');
// comprendre la différence entre guillemets et apostrophe
$age = 10;
echo 'Paul a $age ans';
echo '<br/>';
echo "Paul a $age ans";
?>
```

```
□ <?php
// la saisie d'une valeur
echo "Saisir votre prenom ";
$prenom = fgets(STDIN);
// l'utilisation de la valeur saisie
echo "Bienvenue au PHP en mode console, $prenom !!!";
?>
```

DOCUMENTATION

- [DOCUMENTATION FRANÇAISE DE RÉFÉRENCE PAR CATÉGORIE](#)
- [DOCUMENTATION FRANÇAISE DE RÉFÉRENCE PAR NOM DE FONCTION](#)
[\(ICI MT RAND\)](#)



QUIZ PHP



- Marier les structures de contrôle PHP (else, for, elseif, while, case, switch, if, ...) avec leurs types de contrôle (conditionnelle, boucle, ...)
- La boucle while PHP arrive t'elle toujours à une fin (Oui/Non) ?
- Le caractère de fin d'instruction en PHP est le ... ?
- Parmi les types simples PHP figurent les entités suivantes (entier, réel, date, chaine, objet, booléen, tableau, ...)
- Trier par utilité (Fonctions de typage, Fonctions de test, ...), les fonctions PHP (settype, is_array, is_double, is_real , isset, gettype , is_integer...)

EXERCICES WEB



- Dire bonjour
- Définir une constante
- Définir un calcul
- Définir et parcourir un tableau trigonométrique
- Définir et parcourir un tableau de lettres
- Parcourir un tableau associatif
- Transformer une lettre initiale en majuscule
- Afficher le numéro de version de PHP
- Différencier apostrophes et guillemets

EXERCICES CONSOLE



- Calculer des périmètres et des surfaces
- Multiple de 5 et de 7
- Message de bienvenue à un(e) mineur(e)
- Récupération d'argument ligne de commande
- Rechercher le PGCD de 2 nombres
- Afficher les n mots d'une phrase sur n lignes
- Créer une classe Personne

BIBLIOGRAPHIE

- LEON ATKINSON, 2003, *PHP 5, PARIS*, CAMPUSPRESS
- MICHEL DUTREIX, 2005, *SERVEURS LAMP, NANTES*, ENI
- PHILIPPE RIGAUX, 2005, *MYSQL ET PHP*, GAP, O'REILLY
- JEAN ENGELS, 2005, *PHP 5 – COURS ET EXERCICES*, PARIS, EYROLLES
- GUILLAUME ROSSELLINI, 2008, *PHP5*, PARIS, DEVELOPPEZ.COM
- OLIVIER HEURTEL, 2013, *PHP5 – DÉVELOPPER UN SITE WEB DYNAMIQUE*, NANTES, ENI
- MATHIEU NEBRA, 2014, *CONCEVEZ VOTRE SITE WEB AVEC PHP*, CAEN, ZÉRO



SITOGRAPHIE

- [SITE DE RÉFÉRENCE : PHP.NET](#)
- [WIKIVERSITY : PHP](#)
- [SITE DE CONCEPTEURS : DEVELOPPEZ.COM](#)
- [COMMENT ÇA MARCHE](#)
- [OPENCLASSROOMS](#)



VIDÉOGRAPHIE

- [HTTPS://WWW.YOUTUBE.COM/WATCH?V=M9W_MMEIOL0](https://www.youtube.com/watch?v=M9W_MMEIOL0)
- [HTTPS://WWW.YOUTUBE.COM/WATCH?V=AEB_PUAIA0U](https://www.youtube.com/watch?v=AEB_PUAIA0U)
- [HTTPS://WWW.YOUTUBE.COM/WATCH?V=HIHENTZCSN4](https://www.youtube.com/watch?v=HIHENTZCSN4)
- [HTTPS://WWW.YOUTUBE.COM/WATCH?V=H2YPG4CF860](https://www.youtube.com/watch?v=H2YPG4CF860)

