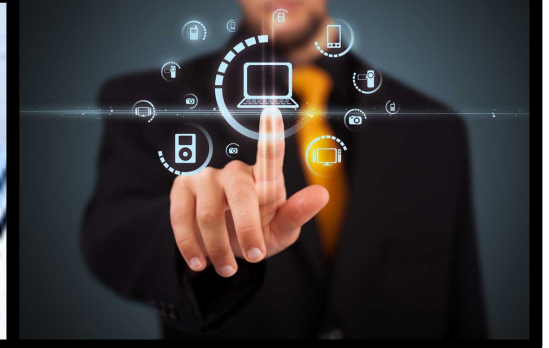


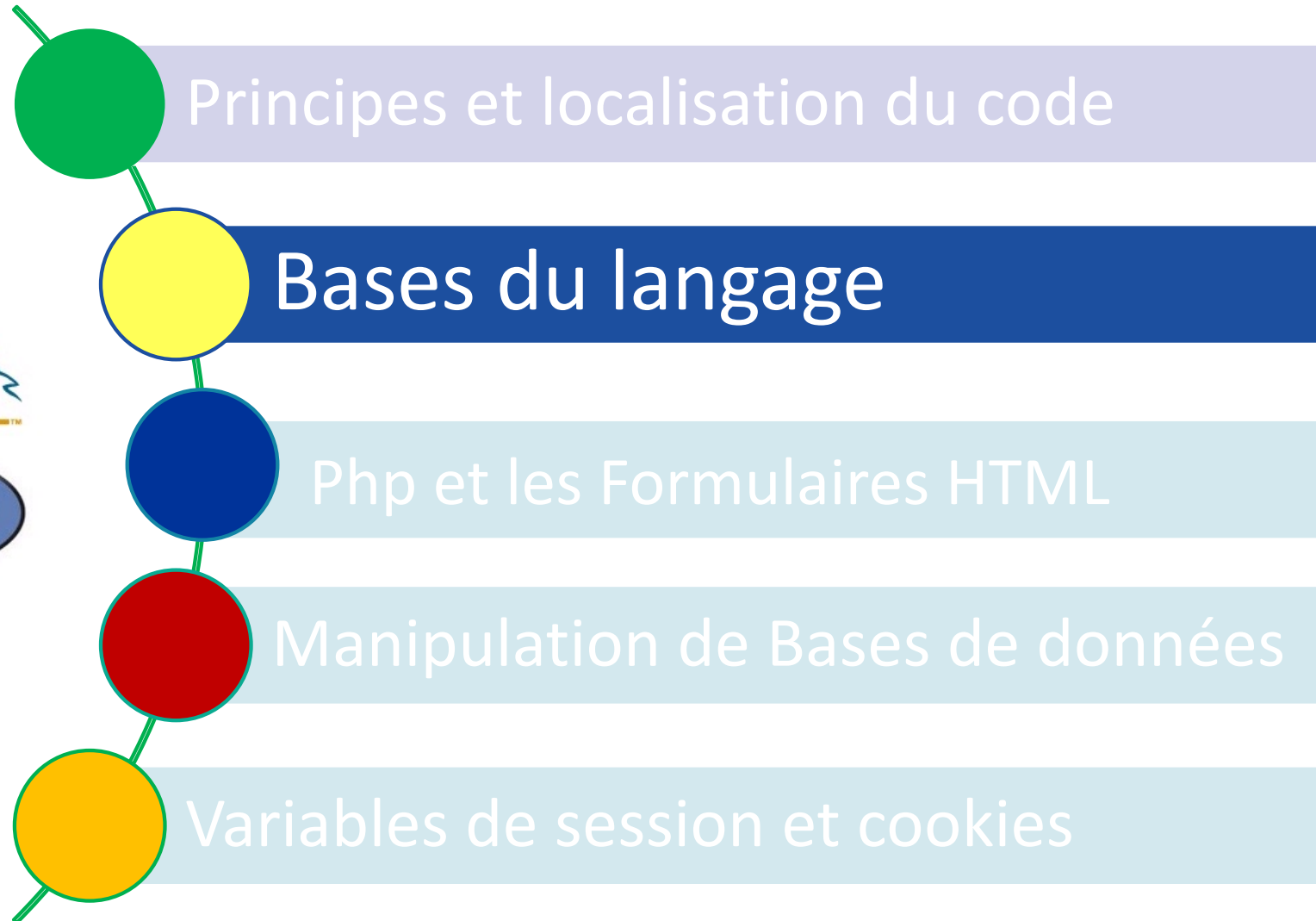
**UNIVERSITE DE CORSE**  
**Licence SPI 2ème année**  
**Option Informatique**

**UE Programmation WEB Back End**  
**CH1.2 – Bases du langage PHP**



Evelyne VITTORI  
vittori\_e@univ-corse.fr

# CH1- Fondamentaux Php



# CH1.2 – Bases du langage

## Objectifs

- Découvrir la syntaxe du langage php
  - Variables, types
  - Structures de contrôle
  - Notion de fonctions
  - Portée des variables
- Comprendre ses spécificités
- Faire des exercices d'application



# Variables et types

- Les noms de variables commencent par **\$**.
- Le type des variables n'est pas déclaré. Comme en JS et en Python
- Les conversions de type sont automatiques ou peuvent être forcées par un cast

```
<?php
    $x = 20;
    echo $x; //affiche 20
    $x .= " etudiants";
    echo $x; //affiche 20 etudiants
    $y= '2'; $y+=1; //$y=3
?>
```

# Constantes

- Les constantes ne commencent pas par un \$.
- Elles sont définies
  - avec la fonction **define**

```
define ("COLOR", "rouge");
```

- ou avec le mot clef **const**

```
const COLOR="rouge";
```

# Chaînes de caractères

- Le point (.) est l'opérateur de **concaténation**.

```
$x = 'Hello' . " World";  
$y = "$x!";  
$z = (5 * 10) . " " . $y;  
//affiche "50 Hello World!"
```

- Pour afficher des caractères spéciaux dans la page HTML, il faut les faire précéder d'un \

```
echo 'J\'aime PHP';  
//affiche J'aime PHP
```

# Chaînes de caractères

La **substitution des variables** fonctionne avec les guillemets doubles uniquement:

- Guillemets simples

- `$nom=' toto ' ;` `toto`

- `$bonjour='bonjour $nom' ;` `bonjour $nom`

- Guillemets doubles

- `$nom= "toto" ;` `toto`

- `$bonjour= "bonjour $nom" ;` `bonjour toto`

# Tableaux

## ■ Initialisation

```
$tab[0] = "A";  
$tab[1] = "B";  
$tab[2] = "C";
```

ou

```
$tab = array("A", "B", "C");
```

Nombre  
d'éléments

## ■ Parcours

```
for ($i = 0 ; $i < count($tab) ; $i++) {  
    echo "La valeur $i est $tab[$i] <br />";  
}
```

Affichage

La valeur 0 est A  
La valeur 1 est B  
La valeur 2 est C



# Exemple d'affichage d'un tableau

```
<!DOCTYPE html>
<head>
  <meta charset= "utf-8">
  <title>Tableaux php</title>
</head>
<body>
<?php
  $tabProf = array("Paoli","Nivet","Simonnet");
  echo "<table border='1'> <thead> <tr> <th> Nom </th>
</tr></thead><tbody>" ;
  for ($i = 0 ; $i < count($tabProf) ; $i++)
  {
    echo "<tr><td>".$tabProf[$i]."</td></tr>";
  }
  echo "</tbody></table>";
?>
</body>
</html>
```

Nom
Paoli
Nivet
Simonnet

# Tableaux associatifs

## ■ Initialisation

```
$tableau = array('rouge' => 4,  
                 'bleu'   => 10,  
                 "blanc"  => 25,  
                 "vert"   => 18) ;
```

## ■ Parcours

2 variables de parcours

**\$cle** prend successivement les valeurs rouge, bleu, blanc, vert

**\$val** prend successivement les valeurs 4, 10, 25, 18

```
$texte="<h2>Tableau des couleurs</h2>";  
foreach ($tableau as $cle => $val)  
{  
    $texte .= "tab[$cle] : $val <br>" ;  
}  
echo $texte;
```

**Affichage**

**Tableau des couleurs**

```
tab[rouge] : 4  
tab[bleu] : 10  
tab[blanc] : 25  
tab[vert] : 18
```

# Fonctions élémentaires

- **isset**(*\$variable*) : renvoie true si la variable est définie et false si elle est égale à null
- **empty**(*\$variable*) : renvoie true si la variable est vide soit:
  - égale à la chaîne vide, à false, à 0 ou null
- **count** (*\$tab*) : renvoie le nombre d'éléments d'un tableau.

# Opérateurs de comparaison

<b>&gt;</b>	<b>Plus grand que</b>
<b>&lt;</b>	<b>plus petit que</b>
<b>==</b>	<b>égal à</b>
<b>!=</b>	<b>non égal à, différent</b>
<b>  </b>	<b>ou</b>
<b>&amp;&amp;</b>	<b>et</b>
<b>+=</b>	<b>x+=y identique à x=x+y</b>
<b>x++</b>	<b>incrémente x</b>
<b>x--</b>	<b>décrémente x</b>

Comme en  
Java et en JS

# Structures conditionnelles

## IF

```
if (condition) {  
    . . .  
}  
else { . . . }  
    expression  
elseif { ...}
```



## SWITCH

```
switch (variable)  
{  
    case valeur1 : . . .  
        break;  
    case valeur2 : . . .  
        break;  
    default : . . .  
}
```

Comme en C,  
en Java, en JS

# Structures itératives



## FOR

```
for ($x = A; $x < B; $x++) {
```

```
break; pour interrompre
```

Comme en C,  
en Java, en JS

## WHILE

```
while (condition)  
{ ... }
```

## DO WHILE

```
do { . . .  
} while (condition)
```

# Structures itératives



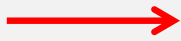
## FOR each (collection)

```
for each($coll as $x) { ... }
```

**\$x** prend successivement la valeur de chacun des éléments de la collection (ou tableau) coll

```
$tab = array("A", "B", "C");  
echo "<ul>";  
foreach ($tab as $valeur) {  
    echo "<li>$valeur</li>";  
}  
echo "</ul>";
```

**Affichage**

- 
- A
  - B
  - C

# Fonctions php

type de retour  
non déclaré

possibilité de  
valeurs par défaut

```
function moyenne($a=10,$b)
{
    return ($a+$b)/2 ;
}
```

```
function afficheTab($tab) {
    for ($i=0; $i<5; $i++) {
        echo "La valeur $i = $tab[$i] <br />";
    }
}
```

types des  
paramètres non  
déclarés



# Portée des variables

- Les variables définies dans un script sont globales dans le script mais ne sont pas accessibles dans les fonctions
- Les variables définies dans une fonction sont **locales** à la fonction

```
<?php
```

Portée globale

```
$x = 1;
```

```
function test()
```

```
{
```

Portée locale

```
    echo $x;
```

```
}?>
```

Rien ne sera  
affiché

# Variables globales

- Pour référencer une variable globale à l'intérieur d'une fonction, il faut la déclarer **global** de manière explicite dans la fonction.

```
<?php  
$x = 1;    Portée globale  
function test()  
{  
    global $x;  
    echo $x;  
}?>
```



1 sera affiché

# Variables globales

- Autre solution: utiliser le tableau associatif superglobal **\$GLOBALS** qui contient les valeurs de toutes les variables globales.

```
<?php
$x = 1;  Portée globale
function test()
{
    echo $GLOBALS['x'];
}?
```



1 sera affiché

# Variables statiques

- Une variable statique est locale à une fonction mais elle **conserve sa valeur** lorsque la fonction est appelée plusieurs fois.

```
<?php  
function compteur()  
{  
    static $cpt = 0;  
    echo $cpt;  
    $cpt++;  
}  
?>
```



- ❖ Lors du premier appel de la fonction compteur, \$cpt sera initialisé à 0
- ❖ Au 2<sup>ème</sup> appel, elle prendra la valeur 1,  
.....

# Exercice CH1.2.1 – Test date/heure



- En reprenant la structure de votre premier programme php, définissez un programme qui affiche la date et l'heure du jour, puis affiche un message spécifique selon l'heure:

- *Bonne journée les L2SPI* (avant midi)
- *Bon après-midi les L2SPI* (entre midi et 20h)
- *Bonne soirée les L2SPI* (après 20h)

**Nous sommes le 18/01/2024 et il est 10heures**  
**Bonne journée les L2SPI**

**Nous sommes le 18/01/2024 et il est 14 heures**  
**Bon après-midi les L2SPI**

**Nous sommes le 18/01/2024 et il est 21 heures**  
**Bonne soirée les L2SPI**

- Vous pourrez utiliser la fonction `Date(format)` qui retourne la date du jour selon le format passé en paramètre:
  - `Date("j/m/Y")` retourne la date au format jour/mois/année
  - `Date("H")` retourne l'heure

Pour plus de détails consultez <https://www.php.net/manual/fr/function.date.php>

## Exercice CH1.2.2 – Fonction



- Ecrire une fonction php ***estBissextile*** qui admet pour paramètre une année \$a et retourne true si l'année est bissextile, et false dans le cas contraire.
  - *NB: une année est bissextile si elle est divisible par 4 et non divisible par 100 ou si elle est divisible par 400*
  - L'opérateur modulo (reste de la division entière) en php est % :    10%2=0    23%4=3
- Définissez un programme de test qui affiche la liste des années bissextiles entre 2020 et 2050:

**Liste des années bissextiles entre 2020 et 2050**

2020 2024 2028 2032 2036 2040 2044 2048

## Exercice CH1.2.3 – Tableaux associatifs



Définissez un programme php réalisant les actions suivantes:

- Définissez une variable \$tabEtu de type tableau associatif contenant les noms et notes des étudiants suivants:
  - Paoli :10, Nivet :19, Simonnet :18
- Affichez le contenu de cette variable dans la page à créer sous la forme d'un tableau HTML
  - *Utilisez des instructions echo pour créer balises (les entêtes (th) et une boucle pour créer les lignes)*

Nom	Note
Paoli	10
Nivet	19
Simonnet	18

# Rappel HTML: Tableaux

Épaisseur bordure extérieure

Taille espace entre bordure et contenu

```
<table border="1"
  cellpadding="1"
  cellspacing="1" >
```

Épaisseur bordure entre cellules

```
<thead>
  <tr> <!-- ligne entête -->
```

```
    <th>TITRE 1</th>
```

```
    <th>TITRE 2</th>
```

```
</tr>
```

```
</thead>
```

...

TITRE 1	TITRE 2
valeur1	valeur2
Cellule fusionnée	

```
<tbody>
```

```
<tr> <!-- ligne 1 -->
```

```
  <td> valeur1 </td>
```

```
  <td> valeur2 </td>
```

```
</tr>
```

```
<tr> <!-- ligne 2 -->
```

```
  <td colspan="2">
```

```
    Cellule fusionnée
```

```
</td>
```

```
</tr>
```

```
</tbody>
```

```
</table>
```

...