



Rna

YOU CAN DO IT

dev

PHP : bases

RAMIANDRISOA Rindra Ny Aina

Plan



- **Variable**
- **Opérateurs**
- **Concaténation**
- **Condition**
- **Boucle**
- **TP**
- **Fonction**



Variable

Variable



- **Stockage de données / valeurs (texte / nombre ...)**
- **Une fois déclarée, elle peut être utiliser à volonté**
- **Commence par \$**
- **\$maVariable = valeur ;**
 - Camel case : 1^{re} lettre en minuscule et le début de chaque mot en majuscule

Variable



- **Type de variables :**

Type	Exemple
Texte	« Rna dev », « test »
Nombre	1 2,4
boolean	true, false
array	[1, 2, 3, 5]

Variable



```
<?php
$nom="Rakoto";
$age=17;
?>
<!DOCTYPE html>
<html>
  <head>
    <title>Premier programme PHP !</title>
  </head>
  <body>
    <p>Je m'appelle <?php echo $nom;?>,
    et j'ai <?php echo $age;?> ans</p>
  </body>
</html>
```



Opérateurs

Opérateurs



• Opérateurs arithmétiques :

Opérateur	Description	Exemple	Résultat
+	Addition	<pre><?php \$x=2 ; echo \$x+2 ; ?></pre>	4
-	Soustraction	<pre><?php \$x=2 ; echo 5-\$x ; ?></pre>	3
*	Multiplication	<pre><?php \$x=4 ; echo \$x*5 ; ?></pre>	20
/	Division	<pre><?php echo 15/5 ; echo "
" ; echo 5/2 ; ?></pre>	3 2.5
%	Modulo (reste de la division)	<pre><?php echo 5%2 ; echo "
" ; echo 10%8 ; echo "
" ; echo 10%2 ; ?></pre>	1 reste de la division de par 2 2 0
++	Incrémente la valeur par 1	<pre><?php \$x=5 ; \$x++; echo \$x ; ?></pre>	x=6
--	Décrémente la valeur par 1	<pre><?php \$x=5 ; \$x-- echo \$x ; ?></pre>	x=4

Opérateurs



- **Opérateurs de tâches:**

Opérateur	Exemple	Est le même que	Description
=	$x=y$	$x=y$	Assigne la valeur de y à x
+=	$x+=y$	$x=x+y$	Incrémente x de la valeur y
-=	$x-=y$	$x=x-y$	Décrémente x de la valeur y
=	$x=y$	$x=x*y$	Multiplie x par y
/=	$x/=y$	$x=x/y$	Divise x par y
. =	$x.=y$	$x=x.y$	Groupe x et y
%=	$x\%=y$	$x=x\%y$	Donne le reste de la division de x par y

Opérateurs



- **Opérateurs de comparaison:**

Opérateur	Description	Exemple
==	est égal à vérifier l'égalité	5==8 renvoie faux
===	égalité strict (valeur et type)	
!=	n'est pas égal	5 !=8 renvoie vrai
<>	n'est pas égal	5<>8 renvoie vrai
>	est plus grand que	5>8 renvoie faux
<	est plus petit que	5<8 renvoie vrai
>=	est supérieur ou égal à	5>=8 renvoie faux
<=	est inférieur ou égal à	5<=8 renvoie vrai



- **Opérateurs de comparaison:**

Opérateur	Description	Exemple
&& and	et	x=6 y=3 (x < 10 && y > 1) retourne vrai
 or	ou	x=6 y=3 (x==5 y==5) retourne faux
!	pas ou non	x=6 y=3 ! (x==y) renvoie vrai



Concaténation

Concaténation



- Rajouter plusieurs chaînes entre elles
- Se fait avec un .

```
<p><?php echo "Je m'appelle ".$nom.", et j'ai ".$age." ans" ;?></p>
```



Condition

Condition



Instruction	Signification
if	Si
else	Sinon
elseif	Sinon si
switch	Selon



Exemple de « if », « else », «elseif »

```
<?php
$nombre = 11;
if ($nombre >= 0 && $nombre < 10) {
    // on teste si la valeur de notre variable est comprise entre 0 et 9
    echo $nombre.' est compris entre 0 et 9';
}
elseif ($nombre >= 10 && $nombre < 20) {
    // on teste si la valeur de notre variable est comprise entre 10 et 19
    echo $nombre.' est compris entre 10 et 19';
}
else {
    // si les deux tests précédents n'ont pas aboutis, alors on tombe dans ce cas
    echo $nombre.' est plus grand que 19';
}
?>
```


Condition



Exemple de « switch » : switch, case, break, default

```
<?php
$nom = "Victor";

switch ($nom) {
    case 'Jean' :
        echo 'Votre nom est Jean.';
        break;
    case 'Benoît' :
        echo 'Votre nom est Benoît.';
        break;
    case 'Victor' :
        echo 'Votre nom est Victor.';
        break;
    default :
        echo 'Je ne sais pas qui vous êtes !!!';
}
?>
```



Boucle

Boucle



for : (pour chaque)

```
1.  for(initialization; condition; iteration)
2.  {
3.      // des instructions
4.  }
```

Boucle



Rna
YOU CAN DO IT
dev

```
<?php
$chiffre = 5;

// Début de la boucle
for ($i=0; $i < $chiffre; $i++) {
    echo 'Notre chiffre est différent de '.$i.'<br />';
}
// Fin de la boucle

echo 'Notre chiffre est égal à '.$i;
?>
```

Notre chiffre est différent de 0

Sortie :

```
Notre chiffre est différent de 0
Notre chiffre est différent de 1
Notre chiffre est différent de 2
Notre chiffre est différent de 3
Notre chiffre est différent de 4
Notre chiffre est égal à 5
```

Boucle



while: (tant que)

```
1.  initialization;  
2.  while (expression)  
3.  {  
4.      instructions;  
5.      compteur;  
6.  }
```

Boucle



```
<?php
$chiffre = 5;
$i = 0;

// Début de la boucle
while ($i < $chiffre) {
    echo 'Notre chiffre est différent de '.$i.'<br />';
    $i = $i + 1;
}
// Fin de la boucle

echo 'Notre chiffre est égal à '.$i;
?>
```

Sortie :

```
Notre chiffre est différent de 0
Notre chiffre est différent de 1
Notre chiffre est différent de 2
Notre chiffre est différent de 3
Notre chiffre est différent de 4
Notre chiffre est égal à 5
```



TP

Créer le tableau suivant avec une boucle

Numéro
1
2
3
4
5
6
7
8



Créer le tableau suivant : numéro impair

- Avec une boucle et une condition (le reste du nombre divisé par 2 est 1)
- Seulement avec une boucle

Numéro impair
1
3
5
7
9
11
13
15



Fonction

Fonction



Portion de code informatique nommée, qui accomplit une tâche spécifique

```
function nom_de_la_fonction(arguments){  
    // instructions  
    return ma_valeur_de_retour  
}
```

- . **function** : mot clé pour indiquer que c'est une fonction
- . **nom_de_la_fonction** : nom de la fonction et est obligatoire
- . **arguments** : les données que la fonction a besoin. Le nombre d'arguments peut varier de 0 à infini
- . **instructions** : les instructions de la fonction
- . **return** : mot clé pour indiquer la fonction retourne
« ma_valeur_de_retour »

Fonction



```
1  <?php
2      function somme($nb1, $nb2){
3          $valeur = $nb1 + $nb2;
4          return $valeur;
5      }
6  ?>
7  <!DOCTYPE html>
8  <head>
9      <title>Somme</title>
10 </head>
11 <body>
12     Somme = <?php echo somme(9, 13); ?>
13 </body>
14 </html>
```

Somme = 22