

Lycée Technique de Taza

BREVET DE TECHNICIEN SUPÉRIEUR (BTS) EN

MULTIMÉDIA ET CONCEPTION WEB (MCW)

MODULE: DÉVELOPPEMENT MULTIMÉDIA

ANNÉE UNIVERSITAIRE : 2021/2022

PROFESSEUR: HICHAM ZAARAOUI

Chapitre 4

Les chaines de caractères

Chaine de caractères: les fonctions

int **strlen**(string \$chaine)

Retourne la longueur de la chaine.

Exemple:

```
<?php  
echo strlen("bonjour!");  
?>
```

Résultat:

8

Chaine de caractères: les fonctions

int **ord**(string \$chaine)

Retourne la valeur ASCII du caractère. Convertit le premier octet d'une chaîne en une valeur entre 0 et 255.

Exemple:

```
<?php  
echo ord("b"), "<br/>";  
echo ord("bonjour");  
?>
```

Résultat:

98

98

Chaine de caractères: les fonctions

string **chr**(int \$code)

Un \$code entre 0 et 255.

Retourne des caractères à partir de différentes valeurs ASCII .

Exemple:

```
<?php  
  
echo chr(52), "<br/>";//Valeur Decimale  
  
echo chr(052), "<br/>";//Valeur en octale  
  
echo chr(0x52); //Valeur en Hexadecimale  
  
?>
```

Résultat:

4
*
R

Chaine de caractères: les fonctions

int **strcmp**(string \$chaine1, string \$chaine2)

Comparaison sensible à la casse, cette fonction retourne < 0 si chaine1 est inférieure à chaine2; et > 0 si chaine1 est supérieure à chaine2, et 0 si les deux chaînes sont égales.

Exemple:

<?php

```
echo strcmp("bonjour","bonjour"), "<br/>";
```

```
echo strcmp("bonjour BTS","bonjour"), "<br/>";
```

```
echo strcmp("bonjour","bonjour BTS");
```

?>

Résultat:

0

4

-4

Chaine de caractères: les fonctions

string **strstr**(string \$chaine1, string \$chaine2)

Recherche la première occurrence d'une chaine2 dans une chaine1 et retourne une sous-chaîne de chaine1 allant de la première occurrence de chaine2 (incluse) jusqu'à la fin de la chaîne1. **strstr()** est sensible à la casse.

Exemple:

```
<?php  
echo strstr("bonjour tous le monde","t"),"<br/>";  
echo strstr("bonjour tous le monde","le");  
?>
```

Résultat:

```
tous le monde  
le monde
```

Chaine de caractères: les fonctions

array **explode**(string \$separateur, string \$chaine)

Découpe une chaine en morceau grâce au séparateur.

Exemple:

```
<?php  
print_r(explode(" ","Bonjour tous le monde. C'est une belle journee.");  
?>
```

Résultat:

Array ([0] => Bonjour [1] => tous [2] => le [3] => monde. [4] => C'est [5] => une [6] => belle [7] => journee)

Chaine de caractères: les fonctions

string **implode**(string \$ separateur, array \$array)

Rassemble les éléments d'un tableau en une chaîne.

Exemple:

```
<?php  
print_r (implode("-",array( "Bonjour", "tous", "le monde")));  
?>
```

Résultat:

Bonjour-tous-le monde

Chaine de caractères: les fonctions

int **strpos**(string \$chaine1, string \$chaine2)

Cherche la position numérique de la première occurrence de chaine2 dans la chaîne de caractères chaine1. elle est sensible à la casse.

Exemple:

<?php

```
echo strpos("hicham.zaaraoui@usmba.ac.ma", "@"), "<br/>";
```

```
echo strpos("hicham.zaaraoui@usmba.ac.ma", "z");
```

?>

Résultat:

15

7

Chaine de caractères: les fonctions

string **str_replace**(string \$chaine1, string \$chaine2, string \$chaine3)

Remplace toutes les occurrences de chaine1 dans chaine3 par chaine2. str_replace() est sensible à la casse.

Exemple:

```
<?php  
echo str_replace("le monde","BTS"," bonjour le monde");  
?>
```

Résultat:

bonjour BTS

Chaine de caractères: les fonctions

string **addslashes**(string \$chaine)

Ajoute des antislashes devant les caractères spéciaux (guillemets simples ('), guillemets doubles ("), antislash (\), NUL (l'octet NUL)).

Exemple:

```
<?php  
echo addslashes('Que produit ce "php" script');  
?>
```

Résultat:

Que produit ce \"php\" script

Chaine de caractères: les fonctions

string **stripslashes**(string \$chaine)

Supprime les antislashes d'une chaîne et retourne la chaîne nettoyée.

Exemple:

```
<?php  
echo stripslashes("Voici une ligne \ puis une autre \ Fin\  
?>
```

Résultat :

Voici une ligne puis une autre Fin

Chaine de caractères: les fonctions

string **htmlspecialchars**(string \$chaine)

Convertit les caractères spéciaux en entités HTML.

Exemple:

```
<?php
$ch = "c'est un texte en <b>bold</b> ";
echo htmlspecialchars ($ch) ;
?>
```

La sortie HTML du code ci-dessus sera (Afficher la source) :

c'est un texte en bold

Résultat au niveau du navigateur:

c'est un texte en bold

Chaine de caractères: les fonctions

string **htmlentities**(string \$chaine)

Convertit tous les caractères éligibles en entités HTML.

Exemple:

```
<?php
$ch = '<a href="https://www.w3schools.com">Go to w3schools.com</a>' ;
echo htmlentities ($ch) ;
?>
```

La sortie HTML du code ci-dessus sera (Afficher la source) :

Go to w3schools.com

Résultat au niveau de navigateur:

Go to w3schools.com

Chaine de caractères: les fonctions

string **ucfirst**(string \$chaine)

Force le premier caractère de la chaine en majuscule.

Exemple:

```
<?php  
echo ucfirst("bonjour tous le monde");  
?>
```

Résultat:

Bonjour tous le monde

Chaine de caractères: les fonctions

string **ucwords**(string \$chaine)

Force le premier caractère de chaque mot d'une chaine en majuscule.

Exemple:

```
<?php  
echo ucwords("bonjour tous le monde");  
?>
```

Résultat:

Bonjour Tous Le Monde

Chaine de caractères: les fonctions

string **strtoupper**(string \$chaine)

Met tous les caractères en majuscule

Exemple:

```
<?php  
echo strtoupper("BoNjour ToUs le Monde");  
?>
```

Résultat:

BONJOUR TOUS LE MONDE

Chaine de caractères: les fonctions

string **strtolower**(string \$chaine)

Met tous les caractères en minuscule.

Exemple:

```
<?php  
echo strtolower("BoNjour ToUs le Monde");  
?>
```

Résultat:

bonjour tous le monde

Chaine de caractères: les fonctions

Exercice d'application: Normalisation de la casse des chaînes

```
<?php
```

```
$nom="ENgels";
```

```
$prenom="jEan";
```

```
$adresse=" 21, rue compoinT";
```

```
$Ville="75018 pAris";
```

```
$mail="ENGELS@funPHP.Com";
```

```
?>
```

Question: Donner le script qui permet d'afficher le résultat suivant :

Résultat:

Mes coordonnées

Jean ENGELS

21, rue Compoint

75018 PARIS

Chaine de caractères: les fonctions

Correction:

```
<meta charset="utf-8" />
<?php $nom="ENgels";
$prenom="jEan";
$adresse="21,rue compoinT";
$Ville="75018 pAris"; $mail="ENGELS@funPHP.Com";
$prenom=ucfirst(strtolower("jEan"));
$nom=strtoupper($nom);
$adresse=ucwords(strtolower($adresse));
$Ville=strtoupper("75018 pAris");
$mail=strtolower("ENGELS@funPHP.Com");
echo "Mes coordonnées", "<br/>";
echo $prenom, $nom, "<br/>";
echo $adresse, "<br/>";
echo $Ville;
?>
```

Chaine de caractères: les fonctions

string **nl2br**(string \$chaine)

Retourne string après avoir inséré
 ou
 devant toutes les nouvelles lignes (\r\n, \n et \r). Elle permet d'améliorer la présentation des textes saisis dans les zones multilignes.

Exemple:

```
<?php  
echo nl2br("Voici une ligne \n puis une autre \n Fin\n");  
?>
```

Résultat:

Voici une ligne
puis une autre
Fin

Chaine de caractères: les fonctions

string **trim**(string \$chaine, charlist)

Supprime les espaces situés en début et en fin de chaîne

Exemple:

```
<?php
```

```
$var1="Voici Tazass";
```

```
echo $var1,"<br/>";
```

```
echo trim($var1,"Vos");
```

```
?>
```

Résultat:

Voici Tazass

ici Taza

Chaine de caractères: les fonctions

string **ltrim**(string \$chaine, charlist)

Enlève les espaces de début de chaine.

Exemple:

```
<?php
```

```
$var1="Voici Taza";
```

```
echo $var1,"<br/>";
```

```
echo ltrim($var1,"Voici");
```

```
?>
```

Résultat:

Voici Taza

Taza

Chaine de caractères: les fonctions

string **rtrim**(string \$chaine, charlist)

Enlève les espaces de fin de chaine.

Exemple:

<?php

```
$var1="Voici Taza";
```

```
echo $var1,"<br/>";
```

```
echo rtrim($var1,"Taza");
```

?>

Résultat:

Voici Taza

Voici

TP: Chaîne de caractères

Exercice 1:

Transformez une chaîne écrite dans des casses différentes afin que chaque mot ait une initiale en majuscule.

La chaîne est : TransFormeZ unE Chaîne éCRITe dans des cASses diFFérenTes afiN qUe chAQue MOT ait une inITiale en MAJUSCULE

Correction:

Code PHP

```
<?php

$ch="TransFormeZ unE Chaîne éCRITe dans des
cASses diFFérenTes afiN qUe chAQue MOT ait
une inITiale en MAJUSCULE";

echo utf8_decode(ucwords(strtolower($ch)));

?>
```

Résultat

Transformez Une Chaîne écrite Dans Des Casses Différentes Afin Que Chaque Mot Ait Une Initiale En Majuscule

TP: Chaîne de caractères

Exercice 2:

En utilisant la fonction `strlen()`, écrivez une boucle qui affiche chaque lettre de la chaîne PHP 5 sur une ligne différente.

Correction:

Code PHP

```
<?php
$ch="PHP5";
for ($i=0; $i<strlen($ch); $i++) {
    echo $ch[$i], "<br/>";
}
?>
```

Résultat

P
H
P
5

TP: Chaîne de caractères

Exercice 3:

À partir de deux chaînes quelconques contenues dans des variables, effectuez une comparaison entre elles pour pouvoir les afficher en ordre alphabétique naturel.

Nous utilisons la fonction `strtolower()` avant d'opérer la comparaison, sinon tous les caractères de A à Z sont avant les caractères a à z.

Code PHP

```
<?php
$ch1="bonjour";
$ch2="BTS";
$ch1=strtolower($ch1);
$ch2=strtolower($ch2);
if (strcmp($ch1,$ch2)<0) {
echo $ch1, "<br/>";
echo $ch2; }else {
echo $ch2, "<br/>";
echo $ch1;
}
?>
```

Résultat

bonjour
BTS

TP: Chaîne de caractères

Exercice 4:

Ecrivez un code php qui permet de :

- ☐ afficher le nombre de caractères d'une chaîne de caractères.
- ☐ mettre la première lettre de chaque mot de la chaîne en majuscule.
- ☐ afficher les caractères 3 à 5 de la chaîne.
- ☐ remplacer dans la chaîne les espaces par des soulignés.

Code PHP

```
<?php
$ch="Ecrivez un code php";
echo strlen($ch), "<br/>";//afficher le nombre de caractères.
echo ucwords($ch),"<br/>";//mettre la première lettre de chaque mot en majuscule.
for($i=2;$i<5;$i++){
echo $ch[$i]; //afficher les caractères 3 à 5 de la chaîne.
}
echo "<br/>";
echo str_replace(" ","_",$ch); // remplacer les espaces par des soulignés.
?>
```

Résultat

19
Ecrivez Un Code Php
riv
Ecrivez_un_code_php

TP: Chaîne de caractères

Exercice 5:

Ecrivez une boucle qui affiche le code ASCII de chaque lettre de la chaîne "PHP 7 news" sur une ligne différente.

Résultat prévu après l'exécution:

Le code ASCII du caractère "P" est : 80
Le code ASCII du caractère "H" est : 72
Le code ASCII du caractère "P" est : 80
Le code ASCII du caractère " " est : 32
Le code ASCII du caractère "7" est : 55
Le code ASCII du caractère " " est : 32
Le code ASCII du caractère "n" est : 110
Le code ASCII du caractère "e" est : 101
Le code ASCII du caractère "w" est : 119
Le code ASCII du caractère "s" est : 115

TP: Chaîne de caractères

Nous avons besoin d'une boucle qui va parcourir la chaîne, pour cela nous avons besoin de la fonction `strlen($ch)` qui renvoie la longueur d'une chaîne et la fonction `ord($ch)` qui renvoie le code ASCII d'un caractère.

Code PHP

```
<head>
<meta charset="UTF-8">
</head>
<?php
$ch="PHP 7 news"; for($i=0;$i<strlen($ch);$i++){
echo "Le code ASCII du caractère \"\$ch[$i]\" est : ",
ord($ch[$i]), "<br/>"; }
?>
```


TP: Chaîne de caractères

Exercice 6:

Rédigez le script qui permet d'afficher la chaîne : "item1item2" telle quelle.

Correction:

Pour interdire l'interprétation d'un code html on doit convertir les caractères éligibles en entités html soit par la fonction htmlspecialchars() ou htmlentities()

Code PHP

```
<?php
echo htmlspecialchars("<ul><li>item1</li><li>item2</li></ul>"), "<br/>";
echo htmlentities("<ul><li>item1</li><li>item2</li></ul>");
?>
```

TP: Chaîne de caractères

Exercice 7:

Ecrivez un script qui teste si une chaîne de caractères contient le caractère '@' (le cas d'une adresse e-mail).

Correction:

On peut utiliser pour ce test simple (qui est insuffisant pour une adresse email) la fonction `strpos` qui renvoie la position de la première occurrence d'une sous-chaîne dans une chaîne et renvoie `false` si elle trouve aucune occurrence.

Code PHP

```
<?php
$email="@adressesmail";
if(strpos($email,"@")==false){
    echo "$email n'est pas une adresse correcte";
}
else
{
    echo "$email est une adresse correcte";
}
?>
```

Remarque:

Il faut utiliser l'opérateur `===` au lieu de `==`, il ne fonctionnerait pas.

Par exemple:

`strpos("@exelibe","@")` renvoie `0 == false`.