



Page web modèle

<http://eric.univ-lyon2.fr/~jdarmont/docs/web/td3modele.php>

Fonctions prédéfinies

Nous allons à titre d'exemple utiliser quelques fonctions de manipulation de chaînes de caractères.

1. Créer le squelette d'une page web classique (en-tête, corps, etc.). La sauvegarder sous le nom `ex3.php`.
2. À l'aide de PHP, affecter la chaîne de caractères suivante à une variable. Respecter exactement les sauts de ligne lors de l'affectation.

```
<p>Nous allons tester ↵  
quelques <em>exemples</em> de fonctions ↵  
de manipulation de chaînes de caractères.</p>
```
3. Afficher la chaîne de caractères, puis afficher le résultat de l'application des fonctions suivantes à la chaîne de caractères : `nl2br`, `strip_tags`, `strtoupper`, `strrev`, `ucwords`, `crypt`. Que font ces fonctions ?
4. Afficher les N/10 premiers caractères de la chaîne en colonne. N est la taille de la chaîne. Utiliser la fonction `strlen` pour l'obtenir.
5. Afficher la sous-chaîne commençant au 6^{ème} caractère de la chaîne et finissant 6 caractères plus loin. Utiliser la fonction `substr($chaîne, $debut, $taille)`. Remarque ?
6. Remplacer les occurrences de « `em` » par « `strong` » dans la chaîne. Pour cela, afficher le résultat de l'application de la fonction `str_replace($sous_chaine_a_remplacer, $sous_chaine_de_replacement, $chaîne_complète)` à la chaîne de caractères.

Référence de toutes les fonctions PHP

- En français : <http://php.net/manual/fr/funcref.php>
- En russe : <http://php.net/manual/ru/funcref.php>

Fonctions personnalisées

7. Définir une fonction de nom `rectangle` prenant deux paramètres `longueur` et `largeur` et permettant d'afficher un rectangle plein du type suivant à l'écran.

```
Rectangle 4 x 10  
*****  
*****  
*****  
*****
```

8. Appeler plusieurs fois la fonction `rectangle` avec diverses valeurs des paramètres (par exemple, 15x10, 15x15, 2x2, 1x1).
9. Dans la fonction `rectangle`, ajouter une valeur par défaut de 10 aux deux paramètres. Appeler à nouveau la fonction, en n'indiquant aucune valeur de paramètre, puis seulement la longueur. Peut-on indiquer seulement la largeur ?
10. Faire retourner à la fonction `rectangle` la surface du rectangle affiché. Dans le programme principal, récupérer cette valeur dans une variable et l'afficher (après affichage d'un rectangle 20x20, par exemple).
11. Écrire une fonction `parag` qui prend en paramètres deux chaînes de caractères et les encadre avec des balises de paragraphe (`<p>...</p>`). Affecter les chaînes « Paragraphe1 ? » et « Paragraphe2 ? » à deux variables, les passer en paramètres à la fonction `parag`, puis afficher les chaînes de caractères. Vérifier que les balises de paragraphe ont bien été ajoutées en consultant le code source de la page web depuis le navigateur.

Inclusions

12. Créer un nouveau fichier nommé `image.html`. Y placer le code HTML permettant d'afficher l'image <http://eric.univ-lyon2.fr/~jdarmon/docs/web/camembert.png> au centre de l'écran (et uniquement ce code, pas d'en-tête, de corps de page, etc.). Inclure le fichier `image.html` dans la page `ex3.php` à deux reprises.
13. Créer un nouveau fichier nommé `fonctions.inc.php`. Y placer le code des fonctions `rectangle` et `parag` entre `<?php ?>` et les supprimer du fichier `ex3.php`. Inclure le fichier `fonctions.inc.php` dans la page `ex3.php`. L'endroit où est spécifiée l'inclusion a-t-il de l'importance ?

Validation

14. Vérifier que le code HTML produit à l'aide de PHP est valide (<http://validator.w3.org>).
15. Mettre en commentaire le code correspondant à la première partie du TD (test des fonctions de manipulation de chaînes).
16. Déterminer précisément d'où proviennent les erreurs de validation.

Correction

```
<!-- ex3.php -->

<!DOCTYPE html>

<html>

<head>
  <meta charset="utf-8" />
  <title>Fonctions PHP - Modèle</title>
  <meta name="Author" content="Jérôme Darmont" />
  <meta name="Keywords" content="Web,PHP,Fonctions" />
  <meta name="Description" content="TD 3" />
</head>

<body>

<?php

// Fonctions sur les chaînes

echo "<h1>Chaînes</h1>\n";

$chaine="<p>Nous allons tester
quelques <em>exemples</em> de fonctions
de manipulation de chaînes de caractères.</p>";

echo "<p><strong>\$chaine =</strong> \$chaine</p>\n";
// Afficher un paragraphe (celui contenu dans $chaine) à l'intérieur d'un
// autre paragraphe (celui du echo) provoque une erreur de validation.
echo "<p><strong>nl2br(\$chaine) =</strong> ".nl2br($chaine)."</p>\n";
echo "<p><strong>strip_tags(\$chaine) =</strong> ".strip_tags($chaine)."</p>\n";
echo "<p><strong>strtoupper(\$chaine) =</strong> ".strtoupper($chaine)."</p>\n";
echo "<p><strong>strrev(\$chaine) =</strong> ".strrev($chaine)."</p>\n";
echo "<p><strong>ucwords(\$chaine) =</strong> ".ucwords($chaine)."</p>\n";
echo "<p><strong>crypt(\$chaine) =</strong> ".crypt($chaine)."</p>\n";
echo "<p><strong>N/10 premiers caractères :</strong><br />\n";

for ($i=0; $i<strlen($chaine)/10; $i++) echo "$chaine[$i]<br />";
echo "</p>\n";

echo "<p><strong>Sous-chaîne(6,6) :</strong> ".substr($chaine, 5, 6)."</p>\n";

echo "<p><strong>em => strong :</strong></p> ".str_replace("em", "strong",
$chaine)."</p>\n";

// Fonctions personnalisées

require("fonctions.inc.php");

echo "<h1>Rectangles</h1>\n";
rectangle(15,10);
rectangle(15,15);
rectangle(2,2);
rectangle(1,1);
rectangle();
rectangle(20);
$aire=rectangle(20,20);
echo "<p>Aire = $aire</p>\n";
```

```

echo "<h1>Passage de paramètre par référence</h1>\n";
$ch1="Paragraphe1 ?";
$ch2="Paragraphe2 ?";
parag($ch1, $ch2);
echo "$ch1 $ch2\n";

// Inclusion HTML

echo "<h1>Image, image</h1>\n";
include("image.html");
include("image.html");

?>

</body>
</html>

<?php // fonctions.inc.php

function rectangle($longueur=10, $largeur=10) {
    echo "<pre>$longueur x $largeur\n";
    for ($lon=0; $lon<$longueur; $lon++) {
        for ($lar=0; $lar<$largeur; $lar++) echo "*";
        echo "\n";
    }
    echo "</pre>\n";
    return $largeur*$longueur;
}

function parag(&$chaine1, &$chaine2) {
    $chaine1="<p>$chaine1</p>";
    $chaine2="<p>$chaine2</p>";
}

?>

<!-- image.html -->

<p style="text-align: center">
    
</p>

```