
Support de Travaux Pratiques Programmation web 2

Niveau 2^{ème} année Licence en Informatique de Gestion : Business
Intelligence

Etablissement ISGI Sfax

Année universitaire 2020-2021

Ministère de l'Enseignement Supérieure et de la Recherche Scientifique
Université de Sfax
Institut Supérieur de Gestion Industrielle de Sfax



Travaux Pratiques

Programmation Web 2

2020 – 2021

2^{ème} année Licence appliquée en Informatique de Gestion : Business Intelligence

Enseignante

Nessrine ELLOUMI, Assistante Vacataire à ISGI de sfax

Sommaire

TP 1 : Gestion des événements et contrôle de saisie sous JavaScript

TP 2 : Les bases de PHP

TP 3 : Structure conditionnelle, Boucles et Formulaires en PHP

TP 4 : Validation des formulaires en PHP

Ministère de l'Enseignement Supérieure et de la Recherche Scientifique
Université de Sfax
Institut Supérieur de Gestion Industrielle de Sfax



Programmation Web 2

TP 1 : Gestion des événements et contrôle de saisie en JavaScript

2020 – 2021

2^{ème} année Licence appliquée en Informatique de Gestion : Business Intelligence

Enseignante :

Nessrine ELLOUMI, Assistante Vacataire à ISGI de sfax

TP 1 : Gestion des événements et contrôle de saisie en JavaScript

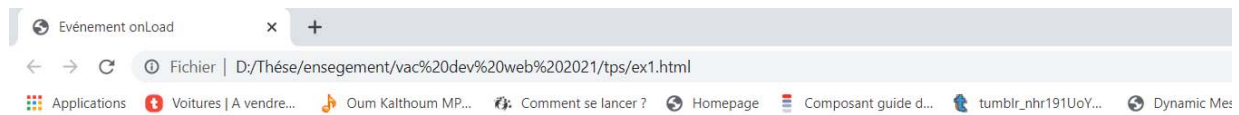
Objectifs :

- Prise en main du JavaScript (gestion des événements).
- Traitement des formulaires avec JavaScript.

1. Événements en JavaScript

Exercice 1 :

Ecrire un script JavaScript qui permet d'afficher message « bravo » et une alerte contient « Bonjour » dès le chargement de la page en utilisant la commande « onLoad() ».



Événement onLoad

Bravo c'est fait

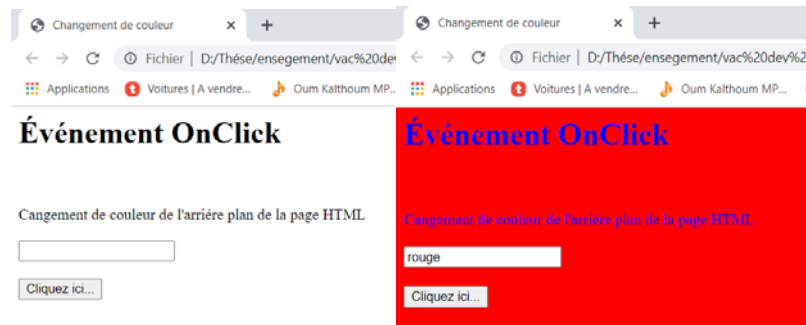
```
<HTML>
<HEAD>
  <TITLE>
    Événement onLoad
  </TITLE>
</HEAD>

<BODY onLoad="alert('Bonjour !')">
  <H1>Événement onLoad</H1>
  <BR><BR>
  <CENTER>
    Bravo c'est fait
  </CENTER>

  <BR><BR><BR>
</BODY>
</HTML>
```

Exercice 2 :

Ecrire un script JavaScript qui permet de changer le couleur de l'arrière-plan de la page en cour.



```

<!DOCTYPE html>
<HTML>
<HEAD>
  <TITLE>
    Changement de couleur
  </TITLE>
  <SCRIPT LANGUAGE="JavaScript">
    function Change() {
      document.fgColor = 'blue'
      document.bgColor = 'red'
      document.form1.coul.value = 'rouge' }
  </SCRIPT>
</HEAD>
<BODY>
  <H1>&Eacute;vénement OnClick</H1>  <BR><BR>
  Changement de couleur de l'arrière plan de la page HTML
  <BR><BR>
  <FORM NAME="form1">
    <INPUT TYPE="text" NAME="coul">
    <BR><BR>
    <INPUT TYPE="button" NAME="int" VALUE="Cliquez ici.." OnClick="Change()">
  </FORM>
</BODY>
</HTML>

```

Exercice 3 :

Ecrire un script JavaScript qui permet de changer l'état d'une lampe à l'aide de deux boutons
Allumer la lumière / Éteindre la lumière en utilisant la commande « onclick() ».



```

<!DOCTYPE html>
<html>
<body>
  <h2> Que peut faire JavaScript? </h2>

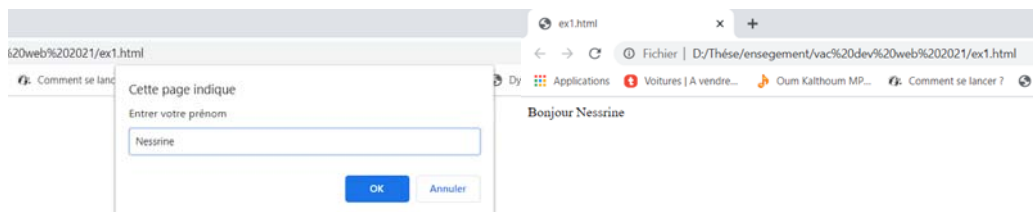
  <p> JavaScript peut modifier les valeurs des attributs HTML. </p>
  <button onclick="document.getElementById('myImage').src='pic_bulbon.gif'">
    Allumer la lumière
  </button>
  

  <button onclick="document.getElementById('myImage').src='pic_bulboff.gif'">Éteindre la
lumière</button>
</body>
</html>

```

Exercice 4 :

Ecrire un script JavaScript qui demande le Prénom de l'utilisateur et affiche « Bonjour 'votre prénom' ».



```

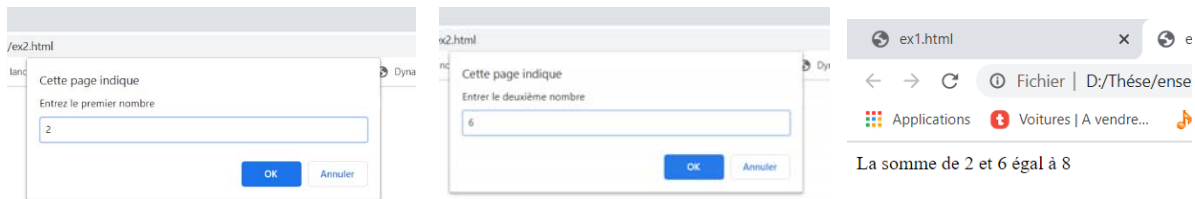
<!DOCTYPE html>

<html>
<head>
  <meta charset="utf-8" />
  <title>ex1</title>
</head>
<body>
  <p id="res"></p>
  <SCRIPT language="JavaScript">
    var phrase = "Bonjour";
    var prenom = prompt("Entrez votre prénom");
    document.getElementById("res").innerHTML = phrase+" "+prenom;
  </SCRIPT>
</body>
</html>

```

Exercice 5 :

Ecrire un script JavaScript qui permet de calculer la somme de deux entier en utilisant les méthodes du Javascript.



```
<!DOCTYPE html>
<HTML>
<head>
  <meta charset="utf-8" />
  <title>ex2</title>
</head>
<BODY>
  <script language="JavaScript">
    var a = prompt("Entrez le premier nombre");
    var b = prompt("Entrez le deuxième nombre");
    var somme = parseInt(a) + parseInt(b);
    document.write("La somme de " + a + " et " + b + " égal à " + somme);
  </script>
</BODY>
</HTML>
```

2. Vérifications des formulaires**Exercice 1 :**

Ecrire un script JavaScript qui permet de vérifier que le champ email obligatoire et bien rempli

Seul le champ email est obligatoire : il doit contenir le symbole @.

Identifiant	<input type="text"/>
Nom	<input type="text"/>
Promotion	<input type="text"/>
adresse	<input type="text"/>
email*	<input type="text"/>
téléphone	<input type="text"/>

```
<!DOCTYPE html>

<html>
<title>verif E-mail</title>
```



```

<script language="JavaScript" type="text/javascript">function check() {
  if (document.F.Email.value.indexOf("@") == -1) {
    alert("Email bien formé obligatoire.\n");
    document.F.Email.focus();
    return false;
  }
  return true;
}</script>
</head>

<body>

  <div>

    <h1>Un champ mail</h1>

    <div>
      Seul le champ <em>email</em> est obligatoire : il doit contenir
      le symbole <kbd>@</kbd>.
    </div>

    <center>

      <form name="F" onsubmit="return check()">
        <table>
          <tbody>
            <tr><td>Identifiant</td><td><input type="text" name="identite"></td></tr>
            <tr><td>Nom</td><td><input type="text" name="lenom"></td></tr>
            <tr><td>Promotion</td><td><input type="text" name="promotion"></td></tr>
            <tr><td>adresse</td><td><input type="text" name="adr"></td></tr>
            <tr><td>email<sup>*</sup></td><td><input type="text"
name="Email"></td></tr>
            <tr><td>téléphone</td><td><input type="text" name="tel"></td></tr>
          </tbody>
        </table>
        <input type="submit">
      </form>

    </center>

  </body>
</html>

```

Exercice 2:

Ecrire un script JavaScript qui permet de vérifier la sélection d'une question radio.

Une question radio

La question est obligatoire : il faut cliquer !

Quelle est votre couleur préférée ?

- ☐ Noir
- ☐ Rouge
- ☐ Bleu
- ☐ Vert

Envoyer

```
<!DOCTYPE html>
<html>
<title>Une question radio</title>
<script language="JavaScript" type="text/javascript">function check() {
  var glaces = document.F.col;
  var nb     = col.length;
  var i      = 0;
  while (i < nb) {
    if (col[i].checked) {
      return true;
    }
    i = i + 1;
  }
  alert("Choisissez un Couleur !\n");
  return false;
}</script>
</head>
<body>
  <div>
    <h1>Une question radio</h1>
    <div>
      La question est obligatoire : il faut cliquer !
    </div>
    <form name="F" onsubmit="return check()">
      Quelle est votre couleur préférée ?<br>
      <input type="radio" name="col" value="Noir"> Noir<br>
      <input type="radio" name="col" value="Rouge"> Rouge<br>
      <input type="radio" name="col" value="Bleu"> Bleu<br>
      <input type="radio" name="col" value="Vert"> Vert<br>
      <br>
      <input type="submit" value="Soumettre ma réponse">
    </form>
  </div>
</body></html>
```

Exercice 3:

Ecrire un script JavaScript qui permet de vérifier que l'utilisateur a choisi au moins une réponse d'une question (check box).

Une question checkbox

La question est obligatoire : il faut au moins une réponse mais pas plus de deux !

Où voulez vous voyager ?

- ☐ Sousse
- ☐ Tunis
- ☐ Hammamet
- ☐ Tabarka

soumettre ma réponse

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
  <title>Une question checkbox</title>
```

```
  <script language="JavaScript" type="text/javascript">function check() {
```

```
    var pays = document.F.elements["pays[]"];
```

```
    var nb    = pays.length;
```

```
    var choisi = 0;
```

```
    var min    = 1;
```

```
    var max    = 2;
```

```
    var i      = 0;
```

```
    while (i < nb) {
```

```
      if (pays[i].checked) {
```

```
        choisi = choisi + 1;
```

```
        if (choisi > max) {
```

```
          alert("Il faut en choisir max 2 pays.\n");
```

```
          return false;
```

```
        }
```

```
      }
```

```
      i = i + 1;
```

```
    }
```

```
    if (choisi < min) {
```

```
      alert("Il faut en choisir une autre !\n");
```

```
      return false;
```

```
    }
```

```
    return true;
```

```
  }</script>
```

```
</head>
```

```
<body>
```

```
  <div>
```

```
    <h1>Une question checkbox</h1>
```

```
    <div>
```

```
      La question est obligatoire : il faut au moins une réponse mais pas plus de deux !
```

```
    </div>
```

```
<form name="F" onsubmit="return check()">
  Oû voulez vous voyager ?<br>
  <input type="checkbox" name="pays[]" value="sousse"> Sousse<br>
  <input type="checkbox" name="pays[]" value="tunis"> Tunis<br>
  <input type="checkbox" name="pays[]" value="hammamet"> Hammamet<br>
  <input type="checkbox" name="pays[]" value="tabarka"> Tabarka<br>
  <br>
  <input type="submit" value="soumettre ma réponse">
</form>
</body>
</html>
```

Ministère de l'Enseignement Supérieure et de la Recherche Scientifique
Université de Sfax
Institut Supérieur de Gestion Industrielle de Sfax



Développement Web 2

TP 2 Les bases de PHP

2020 – 2021

2^{ème} année Licence appliquée en Informatique de Gestion : Business Intelligence

Enseignante:

Nessrine ELLOUMI, Assistante Vacataire à ISGI de sfax

TP 2 : Les bases de PHP

Objectif :

- Déploiement des fichiers PHP sous un serveur web local (XAMPP)
- Manipulation des variables et des tableaux sous PHP.

XAMPP est une solution de serveur Web multiplateforme gratuit et open source développé par Apache composé principalement d'un serveur HTTP Apache.

XAMPP permet d'interpréter les scripts écrits en PHP. Dans ce TP nous allons utiliser XAMPP parce que la plupart des serveurs Web actuels utilisent les mêmes composants que XAMPP, cela facilite la transition d'un serveur de test local à un serveur en direct.

- ✓ Télécharger le logiciel XAMPP et l'installer sous C:/

Question 1. Ecrire le code ci-dessous dans un script en utilisant Notepad++.

```
<!DOCTYPE html>
<html>
<head>
<title> Exercice 1</title>
</head>
<body>
<?php
    echo "Hello word";
?>
</body>
</html>
```

Enregistrer ce script dans l'emplacement « C:\xampp\htdocs » sous le nom « Exercice1.php ».

Question 1. Donner l'URL pour tester ce script dans le navigateur.

<http://localhost/Exercice1.php>

Question 2. Inspecter le code source de cette page web à partir du navigateur.

Pourquoi le code PHP n'existe pas ?

Parce que le code php est interprété par le serveur web.

Question 3. Est-ce que c'est possible d'écrire un code HTML dans un script PHP :

Oui, c'est possible d'écrire un code PHP dans un fichier HTML

```
<!DOCTYPE html>
<html>
<head>
<title> Exercice 1.3</title>
</head>
<body>
<?php
    $nom = "Nessrine";
    echo "<h1>TP PHP</h1>\n";
    echo "<p>Bonjour $nom</p>\n";
?>
</body>
</html>
```

Exercice 2 :

Le langage PHP fournit la fonction `phpinfo()` qui affiche des informations sur la configuration de PHP exemple « version, informations sur le serveur, environnement PHP, chemins, valeurs générales et locales de configuration ».

```
<!DOCTYPE html>
<html>
<head>
<title> Exercice 2</title>
</head>
<body>
<?php
    phpinfo();
-?>
</body>
</html>
```

Question 6. Ecrire le code ci-dessus dans un script PHP nommé « Exercice2.php ». Quel est la version du PHP utilisé?

PHP Version 8.0.8

Question 7. À partir des informations affichées par le script précédant, Quel est la version du serveur Apache.

Apache/2.4.48 (Win64) OpenSSL/1.1.1k

Exercice 3 :

Le PHP ne nécessite pas la déclaration de type ou l'initialisation des variables.

PHP fournit des fonctions pour afficher des informations sur les variables exemple la fonction « `var_dump()` » et « `var_export()` » qui permet d'afficher des informations d'une variable (son type et sa valeur).

« `var_dump()` »

```
<!DOCTYPE html>
<html>
<body>

<?php
$a = 32;
echo var_dump($a) . "<br>";

$b = "Hello world!";
echo var_dump($b) . "<br>";

$c = 32.5;
echo var_dump($c) . "<br>";

$d = array("red", "green", "blue");
echo var_dump($d) . "<br>";

$e = array(32, "Hello world!", 32.5, array("red", "green", "blue"));
echo var_dump($e) . "<br>";

// Dump two variables
echo var_dump($a, $b) . "<br>";
?>

</body>
</html>
```

« `var_export()` »

```
<!DOCTYPE html>
<html>
<body>

<?php
$a = 32;
echo var_export($a) . "<br>";

$b = "Hello world!";
echo var_export($b) . "<br>";

$c = 32.5;
echo var_export($c) . "<br>";

$d = array("red", "green", "blue");
echo var_export($d) . "<br>";

$e = array(32, "Hello world!", 32.5, array("red", "green", "blue"));
echo var_export($e) . "<br>";
?>

</body>
</html>
```

Question 9. Quel est le type de la variable \$a ?

\$a est de type entier.

Question 10. Enregistrer les scripts ci-dessus. Quel est la différence entre les deux fonctions?

`var_export()` affiche seulement la représentation d'une variable ; `var_dump()` affiche le contenu et le type des variables.

Exercice 4 :

En PHP, il existe trois types de tableaux :

- Tableaux indexés : Tableaux avec un index numérique

Il existe deux manières de créer des tableaux indexés :

L'index peut être attribué automatiquement (l'index commence toujours à 0), comme ceci :

```
$fruits = array("Banane", "Orange", "Fraise");
```

L'index peut être attribué manuellement :

```
$fruits [0] = " Banane ";
```

```
$fruits [1] = " Orange ";
```

```
$fruits [2] = " Fraise ";
```

Question 1 : Ecrire un script qui permet d'afficher tous les éléments du tableau \$fruits.

```
<!DOCTYPE html>
<html>
<body>

<?php
$fruits = array("Banane", "Orange", "Fraise");
$arrlength = count($fruits);

for($x = 0; $x < $arrlength; $x++) {
    echo $fruits[$x];
    echo "<br>";
}
?>

</body>
</html>
```

- Tableaux associatifs : Tableaux avec clés nommées

Les tableaux associatifs sont des tableaux qui utilisent des clés nommées. Il existe deux manières de créer un tableau associatif :

```
$fruits = array("Banane"=>"4", "Orange"=>"3", "Fraise"=>"2");
```

Ou bien :

```
$fruits ['Banane'] = "4";
```

```
$fruits ['Orange'] = "3";
```

```
$fruits ['Fraise'] = "2";
```

Question 2 : Ecrire un script qui permet d'afficher le prix de Banane.


```
<!DOCTYPE html>
<html>
<body>

<?php
$fruits = array("Banane"=>"4", "Orange"=>"3", "Fraise"=>"2");
echo "Prix de Banane : " . $fruits['Banane'] . " Dt .";
?>

</body>
</html>
```

- Tableaux multidimensionnels : Tableaux contenant un ou plusieurs tableaux

Un tableau multidimensionnel est un tableau contenant un ou plusieurs tableaux. PHP prend en charge les tableaux multidimensionnels de deux, trois, quatre, cinq niveaux ou plus. Les tableaux multidimensionnels qui contiennent plus de trois niveaux de profondeur sont difficiles à gérer pour la plupart des gens.

Question 3 : Ecrire un script qui permet d'afficher le contenu de ce tableau :

Fruit	En Stock
Banane	40
Orange	50
Fraise	20
Kiwi	30

```
<!DOCTYPE html>
<html>
<body>

<?php
$fruits = array (
    array("Banane",40),
    array("Orange",50),
    array("Fraise",20),
    array("Kiwi",30)
);

echo $fruits[0][0]." En stock: ".$fruits[0][1].", Kg ".<br>;
echo $fruits[1][0]." En stock: ".$fruits[1][1].", Kg ".<br>;
echo $fruits[2][0]." En stock: ".$fruits[2][1].", Kg ".<br>;
echo $fruits[3][0]." En stock: ".$fruits[3][1].", Kg ".<br>;
?>
// affichage avec boucle For:
<?php
for ($row = 0; $row < 4; $row++) {
    echo "<p><b>Ligne numero $row</b></p>";
    echo "<ul>";
    for ($col = 0; $col < 2; $col++) {
        echo "<li>".$fruits[$row][$col]."</li>";
    }
    echo "</ul>";
}
?>

</body>
</html>
```

Ministère de l'Enseignement Supérieure et de la Recherche Scientifique
Université de Sfax
Institut Supérieur de Gestion Industrielle de Sfax



Programmation Web 2

TP 3 : Structure conditionnelle, Boucles et Formulaires en PHP

2020 – 2021

2^{ème} année Licence appliquée en Informatique de Gestion : Business Intelligence

Enseignante :

Nessrine ELLOUMI, Assistante Vacataire à ISGI de sfax

TP 3 : Structure conditionnelle, Boucles et Formulaires en PHP

Objectifs :

- Utilisation des structures conditionnelles et les boucles.
- Développements des fonctions pour afficher des informations.
- Récupération des données à partir d'un formulaire HTML

1. Structures conditionnelle :

En PHP, nous avons les structures conditionnelles suivantes :

- ✓ **If** : exécute du code si une condition est vraie

```
If (condition) {  
    Code à exécuter si la condition est vraie ;  
}
```

Question 1: Ecrire un script qui permet d'afficher « Bonne journée ! » si l'heure actuelle (HEURE) est inférieure à 20.

```
<!DOCTYPE html>  
<html>  
<body>  
  
<?php  
$t = date("H");  
  
if ($t < "20") {  
    echo "Bonne journée !";  
}  
?>  
  
</body>  
</html>
```

- ✓ **if... else** : exécute du code si une condition est vraie et un autre code si cette condition est fausse.

```
If (condition) {  
    code à exécuter si la condition est vraie ;  
} else {  
    code à exécuter si la condition est fausse ;  
}
```

Question 2: Modifier le script précédent pour afficher « Bonne journée ! » si l'heure actuelle (HEURE) est inférieure à 20 et « Bonne nuit ! » sinon.

```

<!DOCTYPE html>
<html>
<body>

<?php
$t = date("H");

if ($t < "20") {
    echo "Bonne journée!";
} else {
    echo "Bonne nuit!";
}
?>

</body>
</html>

```

✓ **if...elseif...else** : exécute des codes différents pour plus de deux conditions

```

If (condition) {
    code à exécuter si la condition est vraie ;
} elseif {
    code à exécuter si la première condition est fausse et cette condition est vraie ;
} else {
    code à exécuter si la condition est fausse ;
}

```

Question 3: Modifier le script précédent pour afficher « Bonne matinée ! » si l'heure actuelle (HEURE) est inférieure à 10, « Bonne journée ! » si l'heure est inférieure à 20 et « Bonne nuit ! » sinon.

```

<!DOCTYPE html>
<html>
<body>

<?php
$t = date("H");
echo "<p>L'heure est " . $t;
echo " :</p>";

if ($t < "10") {
    echo "bonne matinée!";
} elseif ($t < "20") {
    echo "Bonne journée!";
} else {
    echo "Bonne nuit!";
}
?>

</body>
</html>

```

✓ **switch:** est utilisée pour effectuer différentes actions en fonction de différentes conditions.

```

switch(n) {
    case cas1 :
        code à exécuter si n=cas1 ;break;
    case cas2 :
        code à exécuter si n=cas2 ; break;
    case cas3 :
        code à exécuter si n=cas3;break;
}

```

```

...
default:
    code à exécuter si n est différent de toutes les cas ;
}

```

Question 4: Ecrire un script qui permet d'afficher un couleur choisi parmi les trois couleurs {rouge, vert, bleu}.

```

<!DOCTYPE html>
<html>
<body>

<?php
$favcolor = "rouge";

switch ($favcolor) {
    case "rouge":
        echo "le couleur choisi est rouge!";
        break;
    case "blue":
        echo "le couleur choisi est bleu!";
        break;
    case "green":
        echo "le couleur choisi est vert!";
        break;
    default:
        echo "le couleur choisi ni rouge ni bleu ni vert!";
}
?>

</body>
</html>

```

2. Les boucles :

En PHP, nous avons les types de boucles suivants :

✓ **while** : parcourt un bloc de code tant que la condition spécifiée est vraie.

```

while (condition vrai) {
    Traitement;
}

```

Question 1 : Ecrire un script qui permet d'afficher toutes les nombre inférieur ou égale à 5.

```

<!DOCTYPE html>
<html>
<body>

<?php
$x = 1;

while($x <= 5) {
    echo "Le nombre est: $x <br>";
    $x++;
}
?>

</body>
</html>

```

- ✓ **do...while** : parcourt un bloc de code une fois, puis répète la boucle tant que la condition spécifiée est vraie

```
do{  
    Traitement;  
} while (condition vraie)
```

Question 2 : Ecrire un script qui permet d'afficher tous les nombres inférieurs ou égaux à 100.

```
<!DOCTYPE html>  
<html>  
<body>  
  
<?php  
$x = 1;  
  
do {  
    echo "Le nombre est: $x <br>";  
    $x++;  
} while ($x <= 100);  
?>  
  
</body>  
</html>
```

- ✓ **for** : parcourt un bloc de code un nombre spécifié de fois

```
for (initialiser le compteur ; tester le compteur ; incrémenter le compteur) {  
    code à exécuter pour chaque itération ; }
```

Question 3 : Ecrire un script qui permet d'afficher les nombres entre 0 et 10 avec un pas de 10.

```
<!DOCTYPE html>  
<html>  
<body>  
  
<?php  
for ($x = 0; $x <= 100; $x+=10) {  
    echo "Le nombre est : $x <br>";  
}  
?>  
  
</body>  
</html>
```

- ✓ **foreach** : parcourt un bloc de code pour chaque élément d'un tableau

```
foreach ($tableau as $valeur) {  
    code à exécuter ; }
```

Question 4: Ecrire un script qui permet d'afficher le contenu d'un tableau.

```
<!DOCTYPE html>
<html>
<body>

<?php
$colors = array("rouge", "vert", "blue", "jaune");

foreach ($colors as $value) {
    echo "$value <br>";
}
?>

</body>
</html>
```

3. Les Fonctions :

En PHP il est possible de créer des fonctions.

- ✓ Une fonction est un bloc d'instructions qui peut être utilisé à plusieurs reprises dans un programme.
- ✓ Une fonction ne s'exécutera pas automatiquement lors du chargement d'une page.
- ✓ Une fonction sera exécutée par un appel à la fonction.

```
function nomfonction() {
    code à exécuter ; }
```

Question 1: Ecrire une fonction qui permet d'afficher un prénom donné par l'utilisateur.

```
<!DOCTYPE html>
<html>
<body>

<?php
function affiche_prenom($fpre) {
    echo "$fpre <br>";
}

affiche_prenom("Hend");
affiche_prenom("Adam");
affiche_prenom("Lina");
affiche_prenom("Ahmed");
affiche_prenom("Nadine");
?>

</body>
</html>
```

Question 2: Écrire une fonction qui affiche les années bissextiles et non bissextiles depuis l'année 1582 :

```
<!DOCTYPE html>
<html>
<head>
<title> A_bissextile</title>
</head>
<body>

<?php

function bissextile($annee) {
    if( (is_int($annee/4) && !is_int($annee/100)) || is_int($annee/400)) {
        // Année bissextile
        // vous remplacez le retour par ce que vous voulez
        return TRUE;
    } else {
        // Année NON bissextile
        // vous remplacez le retour par ce que vous voulez
        return FALSE;
    }
}

$i = 1582;

echo '<font color="#CC0000">Tableau des années bissextiles depuis ' . $i . '</font><br>';

for($i; $i<=date('Y'); $i++) {
    if(bissextile($i)) {
        echo '<strong>' . $i . ' : Bissextile</strong><br>';
    } else {
        echo $i . ' : Non bissextile<br>';
    }
}
```

4. Les Formulaires :

Lorsque l'utilisateur remplit un formulaire HTML et clique sur le bouton Soumettre, les données du formulaire sont envoyées pour traitement dans un fichier PHP. Il existe deux méthodes de récupération de données la méthode POST et la méthode GET.

Question 1 : Ecrire un script php nommé "welcome.php" qui permet d'afficher les données récupérer à partir d'un formulaire HTML en utilisant la méthode POST.

```
<html>
<body>

Welcome <?php echo $_POST["name"]; ?>

<br>
Your email address is: <?php echo
$_POST["email"]; ?>

</body>
</html>
```

```
<!DOCTYPE HTML>
<html>
<body>

<form action="welcome.php" method="post">
Name: <input type="text" name="name"><br>
E-mail: <input type="text" name="email"><br>
<input type="submit">
</form>

</body>
</html>
```

✓ Utilisez la méthode GET pour récupérer les données (welcome_get.php):

```
<html>
<body>

Welcome <?php echo $_GET["name"]; ?><br>

Your email address is: <?php echo
$_GET["email"]; ?>

</body>
</html>
```

```
<!DOCTYPE HTML>
<html>
<body>

<form action="welcome_get.php" method="get">
Name: <input type="text" name="name"><br>
E-mail: <input type="text" name="email"><br>
<input type="submit">
</form>

</body>
</html>
```


Question 2 : Ecrire un script php nommé "index.php" qui permet de récupérer un nom donné par l'utilisateur et d'afficher « Bonjour » si l'heure de connexion est inférieur à 18h et « Bonsoir » sinon en utilisant la méthode POST.

```
<!DOCTYPE html>

<html>
<head>
<title> Formulaire</title>
</head>
<body>

<?php
if(!Empty($_POST["nom"]))
{
    $nom = $_POST["nom"];
    $heure = date("H");
    if($heure >= 18)
    $message = "Bonsoir $nom,<br />";
    else $message = "Bonjour $nom,<br />";
    echo $message;
}
echo "<br />";
?>
<form action="" method="POST" name="form">
<input type="text" name="nom" value="">
<input type="submit" value="Envoyer">
</form>
</body>
</html>
```

Question 3 : Testez le script ci-dessus. Si on change la méthode d'envoi du formulaire (POST en GET), que faut-il modifier dans le script pour qu'il fonctionne ?

```
<!DOCTYPE html>

<html>
<head>
<title> Formulaire</title>
</head>
<body>

<?php
if(!Empty($_GET["nom"]))
{
    $nom = $_GET["nom"];
    $heure = date("H");
    if($heure >= 18)
    $message = "Bonsoir $nom,<br />";
    else $message = "Bonjour $nom,<br />";
    echo $message;
}
echo "<br />";
?>
<form action="" method="GET" name="form">
<input type="text" name="nom" value="">
<input type="submit" value="Envoyer">
</form>
</body>
</html>
```

Ministère de l'Enseignement Supérieure et de la Recherche Scientifique
Université de Sfax
Institut Supérieur de Gestion Industrielle de Sfax



Programmation Web 2

TP 4 : Validation des formulaires en PHP

2020 – 2021

2^{ème} année Licence appliquée en Informatique de Gestion : Business Intelligence

Enseignante:

Nessrine ELLOUMI, Assistante Vacataire à ISGI de sfax

TP 4 : Validation des formulaires en PHP

Objectifs :

- Validation des données entrées par l'utilisateur.
- Saisie des champs obligatoires.

1. Validation des champs :

Le formulaire HTML sur lequel nous allons travailler durant ce TP contient divers champs de saisie : des champs de texte obligatoires et facultatifs, des boutons radio et un bouton d'envoi :

Validation formulaire Simple

Nom et prenom:

E-mail:

Site Web:

Commentaire:

Genre: ☐ Femme ☐ Homme

Vos données

- Les règles de validation du formulaire ci-dessus sont les suivantes :

Champ	Règles de validation de champ
Nom	(Obligatoire. + Ne doit contenir que des lettres et des espaces
Email	(Doit contenir une adresse email valide (avec @ et .)
Site Internet	(Optionnel. S'il est présent, il doit contenir une URL valide
Commentaire	Facultatif. Champ de saisie multi-ligne (zone de texte)
Sexe	Minimum une des propositions doit être sélectionnée

Question : Ecrire le script nécessaire pour valider les champs de ce formulaire.

➤ Code

```
<!DOCTYPE HTML>

<html>

<head>

</head>

<body>

<?php

$name = $email = $gender = $comment = $website = "";
if ($_SERVER["REQUEST_METHOD"] == "POST") {

    $name = test_input($_POST["name"]);
    $email = test_input($_POST["email"]);
    $website = test_input($_POST["website"]);
    $comment = test_input($_POST["comment"]);
    $gender = test_input($_POST["gender"]);
}

function test_input($data) {
    $data = trim($data);
    $data = stripslashes($data);
    $data = htmlspecialchars($data);
    return $data;
}

?>

<h2>Validation formulaire Simple</h2>

<form method="post" action="<?php echo htmlspecialchars($_SERVER["PHP_SELF"]);?>">

    Nom et prenom: <input type="text" name="name">

    <br><br>

    E-mail: <input type="text" name="email">

    <br><br>

    Site Web: <input type="text" name="website">

    <br><br>

    Commentaire: <textarea name="comment" rows="5" cols="40"></textarea>
```

```
<br><br>
```

Genre:

```
<input type="radio" name="gender" value="female">Femme
```

```
<input type="radio" name="gender" value="male">Homme
```

```
<br><br>
```

```
<input type="submit" name="submit" value="Soumettre">
```

```
</form>
```

```
<h2>Vos données</h2><br><br><br><br>
```

```
</body>
```

```
</html>
```

2. Contrôle sur les champs obligatoires :

Sur le même formulaire, vérifier que les champs nom, email et sexe sont des champs obligatoires (non vide).

Question : Ecrire le script nécessaire pour valider les champs de ce formulaire.

Controle de saisie

* Champ obligatoire

Nom et prenom: *

E-mail: *

Site Web:

Commentaire:

Genre: ☐ Femme ☐ Homme *

Vos données:

➤ Code

```
<!DOCTYPE HTML>

<html>

<head>

<style>

.error {color: #FF0000;}

</style>

</head>

<body>

<?php

// define variables and set to empty values

$nameErr = $emailErr = $genderErr = $websiteErr = "";

$name = $email = $gender = $comment = $website = "";

if ($_SERVER["REQUEST_METHOD"] == "POST") {

    if (empty($_POST["name"])) {

        $nameErr = "Name is required";

    } else {

        $name = test_input($_POST["name"]);

    }

    if (empty($_POST["email"])) {

        $emailErr = "Email is required";

    } else {

        $email = test_input($_POST["email"]);

    }

    if (empty($_POST["website"])) {

        $website = "";

    } else {

        $website = test_input($_POST["website"]);

    }

}
```

```
}

if (empty($_POST["comment"])) {
    $comment = "";
} else {
    $comment = test_input($_POST["comment"]);
}

if (empty($_POST["gender"])) {
    $genderErr = "Gender is required";
} else {
    $gender = test_input($_POST["gender"]);
}
}

function test_input($data) {
    $data = trim($data);
    $data = stripslashes($data);
    $data = htmlspecialchars($data);
    return $data;
}

?>
<h2>Controle de saisie</h2>
<p><span class="error">* Champ obligatoire</span></p>
<form method="post" action="...">
    Nom et prenom: <input type="text" name="name">
    <span class="error">* </span>
    <br><br>
    E-mail: <input type="text" name="email">
    <span class="error">* </span>
```


Site Web: <input type="text" name="website">

Commentaire: <textarea name="comment" rows="5" cols="40"></textarea>

Genre:

<input type="radio" name="gender" value="Femme">Femme

<input type="radio" name="gender" value="Homme">Homme

*

<input type="submit" name="submit" value="Submit">

</form>

<h2>Vos données:</h2>

</body>

</html>

3. Vérifier le saisie des champs Nom, E-mail et site web :

Sur le même formulaire, vérifier que :

Le champ nom contient que des lettres

Le champ Email est bien formé utiliser la fonction PHP filter_var()

La syntaxe du champ site web doit être valide.

Question : Ecrire le script nécessaire pour vérifier la saisie des champs *Nom, E-mail et site web* de ce formulaire.

➤ **Code**

```
<!DOCTYPE HTML>
```

```
<html>
```

```
<head>
```

```
<style>
```

```
.error {color: #FF0000;}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<?php
```

```
// define variables and set to empty values
```

```
$nameErr = $emailErr = $genderErr = $websiteErr = "";
```

```
$name = $email = $gender = $comment = $website = "";
```

```
if ($_SERVER["REQUEST_METHOD"] == "POST") {
```

```
    if (empty($_POST["name"])) {
```

```
        $nameErr = "Name is required";
```

```
    } else {
```

```
        $name = test_input($_POST["name"]);
```

```
        if (!preg_match("/^[a-zA-Z-']*$/", $name)) {
```

```
            $nameErr = "Only letters and white space allowed";
```

```
        }
```

```
    }
```

```
if (empty($_POST["email"])) {
```

```
    $emailErr = "Email is required";
```

```
} else {
```

```
    $email = test_input($_POST["email"]);
```

```
    if (!filter_var($email, FILTER_VALIDATE_EMAIL)) {
```

```
        $emailErr = "Invalid email format";
    }
}

if (empty($_POST["website"])) {
    $website = "";
} else {
    $website = test_input($_POST["website"]);

    if (!preg_match("/^b(?:(:https?|ftp):\\V|www\\.)([-a-z0-9+&@#\\/%?=_~!:,;])*[-a-z0-9+&@#\\/%=_~!]/i",$website)) {
        $websiteErr = "Invalid URL";
    }
}

if (empty($_POST["comment"])) {
    $comment = "";
} else {
    $comment = test_input($_POST["comment"]);
}

if (empty($_POST["gender"])) {
    $genderErr = "Gender is required";
} else {
    $gender = test_input($_POST["gender"]);
}
}

function test_input($data) {
    $data = trim($data);
    $data = stripslashes($data);
    $data = htmlspecialchars($data);
}
```

```
return $data;  
}  
?>
```

<h2>Contole de saisie</h2>

<p>* Champ obligatoire</p>

<form method="post" action="[...]">

Nom et prenom: <input type="text" name="name">

*** **

**

**

E-mail: <input type="text" name="email" value="">

*** **

**

**

Site Web: <input type="text" name="website">

**

**

Commentaire: <textarea name="comment" rows="5" cols="40"></textarea>

**

**

Genre:

<input type="radio" name="gender" value="Femme">Femme

<input type="radio" name="gender" value="Homme">Homme

*** **

**

**

<input type="submit" name="submit" value="Submit">

</form>

**<h2>Vos données:</h2>

**

</body>

</html>