Pages dynamiques avec PHP

Code: php-html

Originaux

url: **http://tecfa.unige.ch/guides/tie/html/php-html/php-html.html**

url: **http://tecfa.unige.ch/guides/tie/pdf/files/php-html.pdf**

Auteurs et version

- Daniel K. Schneider - Vivian Synteta - Olivier Clavel

- Version: 1.1 (modifié le 15/8/01 par DKS)

Prérequis:

- Savoir construire un formulaire HTML

  Module technique précédent: html-forms
- Avoir une idée du standard "CGI"

  Module technique précédent: cgi-intro
- Connaître les bases du langage PHP

  Module technique précédent: php-intro

© TECFA 15/8/01

Activités:

Module d'exercices: act-quiz

Objectifs:

- Ce module montre comment écrire des simples pages Web dynamiques
- Traiter des formulaires avec PHP
- Savoir programmer un forum simple
- Ecrire le résultat d'un formulaire dans un fichier

© TECFA 15/8/01

1. Table des matières détaillée

1. Table des matières détaillée 3
2. Traitement de simple formulaires avec PHP 4
2.1 Traitement de formulaires avec PHP I (Calcul) 4
A.La récupération des variables d'un formulaire 6
B.Calcul et affichage des résultats 6
2.2 Traitement de formulaires avec PHP II 8
2.3 Traitement de formulaires avec PHP III 11
A.Le formulaire HTML (seulement une partie): 11
B.A retenir (pour la suite) 12
C.Comment traiter la requête venant d'une page par la même page ? 13
D.Calcul et affichage des résultats 14
E.A retenir: 15
F.Si c'était trop compliqué: 15
2.4 Tester l'existence de variables et valeurs POST/GET 16
3. Annotation d'une page (écrire dans un fichier et inclure) 17
4. Questionnaires on-line et récupération dans un fichier 20
A.le formulaire 20
B.Afficher le contenu d'un fichier 23
5. Autres format que HTML 24
A.Exemple de génération VRML 24
B.Exemple de génération GIF 24
6. Conseils pour le débogage 25

 © TECFA 15/8/01

2. Traitement de simple formulaires avec PHP

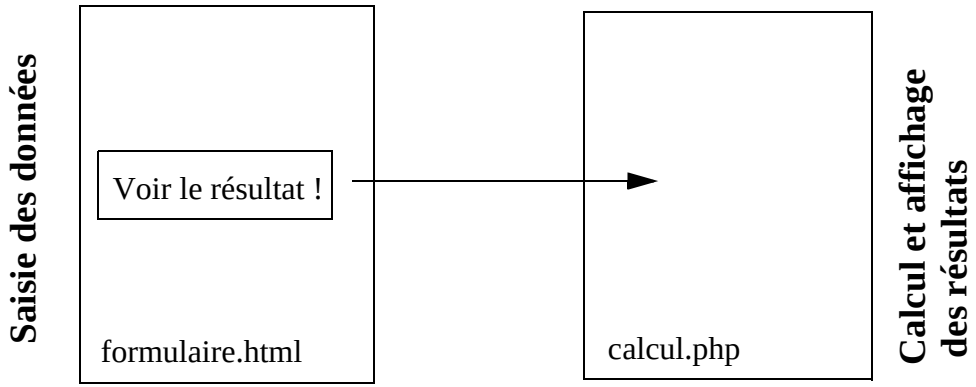
2.1 Traitement de formulaires avec PHP I (Calcul)

Exemple 2-1: Un simple quiz et POST vers un autre fichier PHP

url: Voir: **/guides/php/examples/simple-calculate/formulaire.html**

url: Source: **/guides/php/examples/simple-calculate/formulaire.text**

- Cet exemple montre:
  - comment traiter un formulaire HTML
  - comment calculer et afficher un résultat.

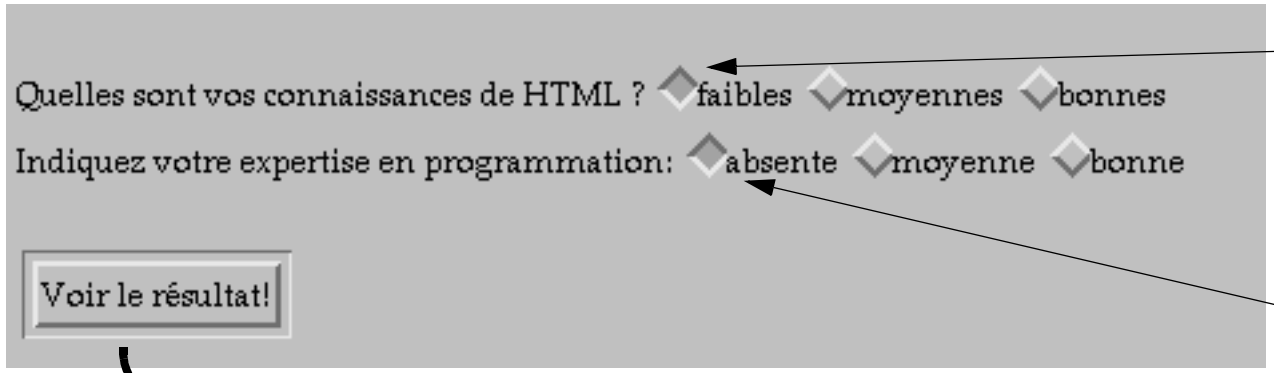


© TECFA 15/8/01

Le formulaire HTML (seulement une partie):

```
<form action="calcul.php" method=post>

Quelles sont vos connaissances de HTML ?
<input type="radio" name="choice" value="1" checked>faibles
<input type="radio" name="choice" value="2">moyennes
<input type="radio" name="choice" value="3">bonnes
<br>
Indiquez votre expertise en programmation:
<input type="radio" name="choice2" value="1" checked>absente
<input type="radio" name="choice2" value="2">moyenne
<input type="radio" name="choice2" value="3">bonne
<P>
<input type="submit" value="Voir le résultat!">
</form>
```



NAME=”choice”

NAME=”choice2”

En PHP les valeurs sont accessibles par
les variables $choice et $choice2

© TECFA 15/8/01

A. La récupération des variables d'un formulaire

- Un grand avantage de PHP par rapport à Perl / Python etc.
- Chaque `name=xxxx` définit automatiquement une variable `$xxxx`

Exemple:

- `$nom` représente en PHP le contenu du champs HTML identifié par "name=nom"
- Dans notre exemple, nous avons deux variables PHP: $choice et $choice2

B. Calcul et affichage des résultats

url: Voir: `/guides/php/examples/simple-calculate/calcul.phps`

- Le traitement des résultats consiste à additioner le score des deux réponses puis à donner un feed-back en fonction de celui-ci.

© TECFA 15/8/01

Voici le code:

```
<?
$score = $choice + $choice2;
echo("<h3>Votre score est de $score </h3>");
if ($score < 3) {
  echo ("<p>Vous &ecirc;tes un d&eacute;butant</p>");
} elseif ($score < 5) {
  echo ("<p>Vous avez un niveau moyen</p>");
} else {
  echo ("<p>Vous êtes un expert !</p>");
}
?>
```

A retenir:

- On utilise la fonction echo() pour afficher du HTML.
  Les $variables dans un string "echo" sont substituées par PHP.
- Notez comment générer du HTML conditionalisé
  (le message est différent selon le score de l'utilisateur).

Exercice 1: Prévoir l'accès direct à cette page PHP (sans données)

- (1) if (isset($choice)) then { ..... } else { echo "désolé ....."; }
- (2) Alternativement: if (!isset($choice)) {echo "désolé"; exit; }

2.2 Traitement de formulaires avec PHP II

Exemple 2-2: Checkboxes avec PHP - arrays

url: **/guides/php/examples/simple-calculate/formulaire4.phps**
url: **/guides/php/examples/simple-calculate/formulaire4.html**

Voici un extrait du formulaire html:

```
<form action="calcul4.php" method=post>
Quels sont vos couleurs préféres?
<br>
<input type="checkbox" name="choice[]" value="Red">Red
<table bgcolor="red" width="50"><tr><td> </td></tr></table>

<input type="checkbox" name="choice[]" value="Blue">Blue
<table bgcolor="blue" width="50"><tr><td> </td></tr></table>

<input type="checkbox" name="choice[]" value="Green">Green
<table bgcolor="green" width="50"><tr><td> </td></tr></table>
.....
<input type="checkbox" name="choice[]" value="Black">Black
<table bgcolor="black" width="50"><tr><td> </td></tr></table>

<input type="submit" value="Voir le résultat!">
</form>
```

- Notez bien la syntaxe pour stocker les choix dans un array: `"choice[]"`

© TECFA 15/8/01

Voici le code php qui traite le formulaire:

```
<?
echo("<h3>Vos couleurs préférées sont </h3>");

for ($i=0;$i<sizeof($choice);$i++) {
    if (isset($choice[$i])) {
       echo("$choice[$i] - ");
    }
}
?>
```

Exemple 2-3: Checkboxes avec PHP - multiples variables

- L'exemple 2-2 "Checkboxes avec PHP - arrays" [8] ci-dessus montre comment efficacement traiter un série de checkboxes
- L'exemple ici montre comment faire la même chose d'un façon inefficace. On crée une variable pour chaque checkbox
  url: **/guides/php/examples/simple-calculate/formulaire4.phps**
  url: **/guides/php/examples/simple-calculate/formulaire4.html**

 © TECFA 15/8/01

Voici un extrait du code HTML:

```
<form action="calcul3.php" method=post>
Quels sont vos couleurs préféres?
<input type="checkbox" name="choice1" value="Red">Red
<table bgcolor="red" width="50"><tr><td> </td></tr></table>

<input type="checkbox" name="choice2" value="Blue">Blue
<table bgcolor="blue" width="50"><tr><td> </td></tr></table>
.....
<input type="checkbox" name="choice6" value="Black">Black
<table bgcolor="black" width="50"><tr><td> </td></tr></table>
<input type="submit" value="Voir le résultat!">
</form>
```

Voici un extrait du code PHP:

```
<?
echo("<h3>Vos couleurs préférées sont </h3>");

if (isset($choice1)) {echo("$choice1 - ");}
if (isset($choice2)) {echo("$choice2 - ");}
if (isset($choice3)) {echo("$choice3 - ");}
if (isset($choice4)) {echo("$choice4 - ");}
if (isset($choice5)) {echo("$choice5 - ");}
if (isset($choice6)) {echo("$choice6");}
?>
```


© TECFA 15/8/01

2.3 Traitement de formulaires avec PHP III

Exemple 2-4: Prix calcul d'une voiture avec le même fichier PHP

url: **voir:** /guides/php/examples/calculate-demo/prix-bagnole.php

url: **source:** /guides/php/examples/calculate-demo/prix-bagnole.phps

- Cet exemple montre:
  - comment traiter un formulaire HTML avec du code dans la même page.

A. Le formulaire HTML (seulement une partie):

```
<FORM METHOD="POST" ACTION="<? echo $PHP_SELF ?>">
      ........
      <br>
      Entrez le cout de votre assurance/ Insurance<br>
      <input type="text" name=assurance size=8 value=0><br>

      Entrez le prix que vous ont coûté vos accidents / Accidents<br>
      <input type="text" size=8 name=accidents value=0><br>
........
      Entrez le coût de vos amendes ... / fines<br>
      <input type="text" size=8 name=amendes value=0><br>
      Votre Nom / Your name
      <input type="text" size=40 name=nom><br>
      <HR width=250><BR>
      <input type="submit" value="Alors / Let's see !?!" name="process">
  </FORM>
  </center>
```

B. A retenir (pour la suite)

- ACTION="<? echo $PHP_SELF ?>"
  dit qu'il faut poster le résultat vers le même fichier.
- C'est la seule chose qui change par rapport à l' exemple 2-1 "Un simple quiz et POST vers un autre fichier PHP" [4]
- Notez `name="process"` dans le "submit".
  Cette variable nous sera utile plus tard.

© TECFA 15/8/01

C. Comment traiter la requête venant d'une page par la même page ?

- d'abord ce n'est pas une nécéssité
  (on aurait pu appeler une autre page PHP)
- il suffit de tester si une variable définie implicitement par le formulaire est présente. Dans notre cas c'est $process:

```
<?
if (!isset($process)) {
?>
//... ici on affiche le formulaire (directement en HTML)
//    uniquement si $process = FALSE.

  <FORM METHOD="POST" ACTION="<? echo $PHP_SELF ?>">
  .....
  </FORM>

<?
} else {
//... ici on effectue les calculs
}
?>
```

A retenir:

- on peut même "mixer" du HTML à l'intérieur d'une instruction PHP
- Le "if" fonctionne un peu près comme tous les langages "C-like".
  A la place du "if - then - else" on aurait pu utiliser un simple "if" plus "exit", voir l'exemple 4-1 "Questionnaire et résultats dans fichier" [20]

D. Calcul et affichage des résultats

Voici le code:

```
    / / calculate the cost per month
$cost = round(((((((($accidents + $assurance) + ((((($consokilo / 100.0) * $kilomois) *
$prixcarbu) * 12.0)) + $vignette) + $tcs) + $autoroutes) + $entretien) + $amendes) / 12.0);

echo "<h2>Resultat/Result</h2>";

// print the name if we got one
if ($nom) { echo "$nom, votre "; } else { echo "Votre ";}

// print the results
echo "bagnole vous coutera environ $cost francs par mois / Your car costs about $cost Swiss
francs/month.<p>";

// give a short comment
  if ($cost < 10) {
   $evaluation = "Heh c’est pas sérieux / Mhh this doesn’t look serious !";
   }
  elseif ($cost < 400.0) {
   $evaluation = "Vous en sortez bien / This is fine :)";
   }
  else {
   $evaluation = "Vous ne vous en sortez pas bien, pensez aux transports communs / You pay quite
a lot!";
   }
echo "$evaluation <p>";
```

 © TECFA 15/8/01

E. A retenir:

- Ici on ne mixe pas HTML avec PHP (on aurait pu).
- On utilise la fonction echo() pour afficher du HTML.
  Les $variables dans un string sont substituées par PHP.
- Notez comment générer du HTML conditionalisé à 2 endroits
  (pour afficher le nom si le champs a été rempli et pour donner un commentaire sur le coût).

F. Si c'était trop compliqué:

Voir la version intégrée de l' exemple 2-1 "Un simple quiz et POST vers un autre fichier PHP" [4]:

Exemple 2-5: Un simple quiz et POST vers le même fichier PHP

url: Voir: **/guides/php/examples/simple-calculate/form-calcul.php**

url: Source: **/guides/php/examples/simple-calculate/form-calcul.text**

© TECFA 15/8/01

2.4 Tester l'existence de variables et valeurs POST/GET

Dans PHP vous pouvez tester si une variable "POST / GET" existe ou si elle a une valeur non-zéro ou non-vide (string):

2 méthodes

1. *"isset"* pour voir si une variable existe:
   - utiliser pour décider s'il faut afficher un formulaire ou traiter le formulaire

```
if (isset($traiter) ) { .... faire ....}
```

2. *"empty"* pour tester si une variable est zéro, vide ***ou*** inexistante
   - utiliser pour décider si l'utilisateur a rempli un champs ou clické sur un bouton

```
if (empty ($input) ) { ... ne rien faire} else { ... faire ...}
```


© TECFA 15/8/01

3. Annotation d'une page (écrire dans un fichier et inclure)

- Il s'agit d'un simple système d'annotation
- Il existe des solutions mysql / php plus utiles (mais plus compliquées)

Exemple 3-1: Un simple système d'annotation

Usage:

- Créer une page php avec un contenu (ici: forum.php)
- Inclure le module "annotate.php" (voir page suivante!)
- Créer un fichier *.php.comment (ici: forum.php.comment)
  - dans lequel le module va écrire les commentaires
  - ce fichier doit être "writable" par le serveur (uid=nobody, ou mode =666)

Le fichier exemple

url: **http://tecfa.unige.ch/guides/php/examples/forum/forum.php**

url: **http://tecfa.unige.ch/guides/php/examples/forum/forum.text**

```
Voici un texte ..... Il peut être aussi long ou compliqué que vous voulez !
<hr>
<?
require("annotate.php");
?>
```

Le module d'annotation

- Renommez ce fichier *.text en fichier *.php pour l'inclusion dans votre page.

  url: **http://tecfa.unige.ch/guides/php/examples/forum/annotate.text**

```
$secret = "zap";
$mode = "a";

// the module can't be called by itself
if ( basename($PHP_SELF) == "annotate.php") {
  exit;
}

// if the password is correct we overwrite
if ($password == $secret) {
  $mode = "w";
  $fp = fopen (basename($PHP_SELF) . ".comment", $mode);
  fwrite ($fp, $message);
  fclose ($fp);
}
// if there is a message we append it to a file called file.comment
// make sure that file.comment belongs to the server id (or world writable)
if ($message) {
  /* uncomment the next two lines to strip out html from input */
  /* $name = strip_tags($name); */
  /* $message = strip_tags($message); */
  $name = StripSlashes($name);
  $message = StripSlashes($message);
  $message = ereg_replace("\n\n", "\n<P>", $message);
```


© TECFA 15/8/01

```
    $date = date("l, F j Y, H:i");
    $message = "<B>$name </B> -- $date<P> $message <BR><HR>";
    $fp = fopen (basename($PHP_SELF) . ".comment", $mode);
    fwrite ($fp, $message);
    fclose ($fp);
    @readfile(basename(($PHP_SELF . ".comment")));
  }
```

A retenir:

- Pour écrire dans un fichier, il faut:
  - que le fichier appartienne à "nobody" (le serveur tourne sous ce nom) ou alternativement qu'il soit "world writeable" (chmod 666)
  - `fopen(<nom du fichier>, mode)`
    ouvre un fichier et retourne un "handle"
  - `fwrite (<handle>, "string")`
    permet d'écrire (il existe d'autres fonctions pour cela)
  - `fclose (<handle>)`
    ferme le fichier (IMPORTANT)
  - `@readfile (<file>)`
    insére le contenu d'un fichier

© TECFA 15/8/01

4. Questionnaires on-line et récupération dans un fichier

Exemple 4-1: Questionnaire et résultats dans fichier

url: **voir:** http://tecfa.unige.ch/guides/php/examples/form-file-demo/

- le fichier new-entry.php contient un formulaire et le code pour rajouter
- le fichier dump_results.php affiche le contenu du fichier

A. le formulaire

- posté et traité avec la même méthode rencontré dans la section 2.3, p. 11
- par contre une partie du HTML est générée par une fonction PHP:

```
$scales = array("food", "work", "love", "leisure", "sports");
function scale ($thing) {
   echo "<TR> <TD align=right>Importance of <STRONG>$thing</STRONG>:</TD>";
   echo "<TD><select name=$thing>";
   echo "<option value=1>1 - totally unimportant";
   echo "<option value=2>2 - not important";
   echo "<option value=3 selected>3 - rather not important";
   echo "<option value=4>4 - slightly important";
   echo "<option value=5>5 - rather important";
   echo "<option value=6>6 - very important";
   echo "</select>";
   echo "</TD></TR>";
   }
```

 © TECFA 15/8/01

```
/* the stuff below could have been done with a for loop, but I was exploring a bit :)  */
function dump_scales ()  {
 global $scales;
 reset($scales);
 do {
    $scale = scale(current($scales));
    echo "$scale\n";
    }
    while (next($scales));
 } ?>
<form> <table>
......
dump_scales();
......
</table> </form>
Ecrire dans un fichier
// check existance of file (or try to create it)
  // a better alternative to touch() would be is_file, is_writable and so on.
    $try = touch($file_name);
  if (!$try) {
    echo "<p>Sorry I can’t open a file, something is wrong";
    exit;
  }

  // this is the stuff we get from the form, we insert it into an array
    $input = array ($login, $password, $fullname, $url, $food, $work, $love, $leisure, $sports);

  // so we can make a big string with tabs between the elements
    // note that we add a \n (line break) to the end of the string.
    $output_line = implode ($input, "   ")."\n";
```


© TECFA 15/8/01

```
  // Now open the file (get a file pointer)
    // We will append to it and therefore use the "a" option
    $output_stream = fopen($file_name, "a");
  // and dump the string into the file
    $result = fputs ($output_stream, $output_line);

  // give feedback
    if ($result) {
      echo "<p>Your data have successfully been registered.";
    }
  else {
    echo "<p>Too bad, the db did not want your data.";
  }
  // close the file pointer
    fclose($output_stream);
  ?>

<?
  // EXIT here ... we don’t want to see the form again. If you do, kill the exit
  exit;
}
?>
```

A retenir:

- Voir aussi: exemple 3-1 “Un simple système d’annotation” [17]
- `fputs(<handle>, “string”)`
  permet d’écrire (il existe d’autres fonctions pour cela)

 © TECFA 15/8/01

B. Afficher le contenu d'un fichier

.... on ne se fatigue pas: le tout dans un <pre> avec un "include"

```
<BODY>
    <H1>PHP/MySQL Demo - Dump Database Contents</H1>

<?
  /* Daniel.Schneider@tecfa.unige.ch
  Will dump the contents of the results file
  */
?>

  <strong>Results registered so far:</strong>
<pre>
<? include("results/result.text"); ?>
</pre>
..........

</BODY>
```


© TECFA 15/8/01

5. Autres format que HTML

Principe: vous devez dans la PREMIERE ligne du script définir le content-type:

- Exemple: `Header("Content-type: image/gif");`

A. Exemple de génération VRML

- planter 100 arbres sans se fatiguer

  url: **voir:** /guides/php/examples/vrml-temple/mixing/

```
<?  Header("Content-type: model/vrml");
    echo "#VRML V2.0 utf8"; ?>

PROTO Tree ....

Transform {
   translation -5 0 -10
   children [
<?
for ($i=0; $i<10; $i++) {
     for ($j=0; $j<10; $j++) {
     echo "Tree { translation $i 0 $j }";
     }    }
?>
  ] }
```

B. Exemple de génération des images PNG

url: **voir:** /guides/php/examples/button

© TECFA 15/8/01

6. Conseils pour le débogage

Affichage de tous les avertissements et erreurs

- Si votre serveur est configuré pour ne pas afficher les simples warnings (par exemple utilisation de variables vide et non-initialisées), vous êtes conseillés de demander à PHP d'afficher un maximum des erreurs
- Insérer au *début de votre code* la ligne suivante !

```
error_reporting(63);
```

Informations

- Si vous voulez connaître toutes les variables auxquelles vous avez accès (y compris les variables / names transmis depuis un formulaire), insérez qq part dans le fichier:

```
phpinfo();
```


© TECFA 15/8/01

© TECFA 15/8/01