

# IUT de Montpellier - Programmation Web - TD1

## Introduction à PHP

Rémi COLETTA

Semaine du 19 Septembre 2011

## 1 HTML

Créer une page `index.html` avec le contenu suivant et enregistrer la dans le répertoire `public_html` de votre espace personnel.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>Insert title here</title>
</head>
<body>
Hello world
<!-- ceci est un commentaire-->
</body>
</html>
```

Ouvrez cette page dans le navigateur directement depuis votre gestionnaire de fichiers

`file:///chemin_de_mon_compte/public_html/index.html`

Ouvrez cette page dans le navigateur depuis le serveur web `file:///infolimon/~mon_login/index.html`

Quelle(s) différence(s) observez vous ?

Essayez d'ouvrir `file:///chemin_de_mon_compte/public_html/indAx.html` puis `file:///infolimon/~mon_login/indAx.html`. Quelle différence(s) observez vous dans le message d'erreur ?

## 2 PHP

### 2.1 hello world

Reprenez le code de la page `index.html`, enregistrez le sous `hello.php` et modifier le corps de la page web (entre balises `< body >`) en

```
<p>
<?php
$texte="hello"; //commentaire en PHP
$texte=$texte." "."world"; // concatenation de 2 chaines de caracteres
echo $texte;
?>
</p>
<?php
echo date(DATE_RFC822);
?>
```

Ouvrez cette page dans le navigateur directement depuis votre gestionnaire de fichier `file://chemin_de_mon_compte/public_html/hello.php`. Que se passe t'il ?

Ouvrez cette page dans le navigateur depuis le serveur web de l'IUT `file://infolimon/~mon_login/hello.php`. Puis rechargez la page après quelques secondes, que dire de son contenu ?

## 2.2 hello world

Créez une page `formulaire.html`, avec comme corps un formulaire composé de 3 champs texte, mis en page à l'aide d'un tableau :

```
<form method="get" action="echo.php">
<table BORDER=0>
<tr>
<td>Nom</td>
<td>
<input type="text" name="nom"/>
</td>
</tr>
<tr>
<td>Prenom</td>
<td>
<input type="text" name="prenom"/>
</td>
</tr>
<tr>
<td>Email</td>
<td>
<input type="text" name="email"/>
</td>
</tr>
<tr>
<td COLSPAN=2> <!-- on fusionne les 2 colonnes de cette ligne -->
<input type="submit" value="Envoyer"/>
</td>
</tr>
</table>
</form>
```

Vous trouverez de plus amples informations (et des exemples) sur les formulaires HTML à l'adresse <http://www.commentcamarche.net/contents/html/htmlform.php3>

Cliquez sur le bouton "Envoyer" vous voyez apparaître dans votre navigateur l'url : `file://infolimon/~mon_login/echo.php?nom=remi&prenom=coletta&email=coletta%40lirmm.fr`

Il faut donc désormais coder la page `echo.php`. Dans le corps de cette page, vous pouvez récupérer la valeur du champ "nom" du formulaire à l'aide de :

```
<?php
$variableNom = $_GET["nom"];
?>
```

Complétez cette page de sorte qu'elle affiche les nom, prenom et adresse email saisis dans le formulaire.

Afin d'éviter que les paramètres du formulaire n'apparaissent dans l'url, modifiez le formulaire pour qu'il appelle la méthode `post` :

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

et côté PHP, récupérez les paramètres avec

```
<?php
$variableNom = $_POST["nom"];
?>
```

### 3 Connection à la base de données

Connectez vous à votre base de données MySQL, à l'aide de l'interface PhpMyAdmin <http://infolimon/my> et changez votre mot de passe puis créez une table "contacts" possédant 3 champs texte : nom, prenom, email. (Attention, comme pour les variables : ne pas utiliser d'accent pour le nom des champs).

Dupliquez votre formulaire en `form-insert.html` et modifiez le pour qu'il appelle le script `insert.php`.

Ce script, dupliqué à partir de `echo.php` devra récupérer les valeurs du formulaire puis se connecter à la base, et insérer un tuple dans la table contacts.

```
<?php
$hostname='serveur_base_donne'; //infolimon
$login='root'; //login iut
$password='root'; // votre passwd
$base='test'; //login iut

$connect = mysql_connect($hostname,$login,$password) or die ("erreur de connexion");
mysql_select_db($base,$connect) or die ("erreur de connexion base");

$req = "INSERT INTO ma_table (nom,prenom,email) VALUES('$nom','$prenom','$mail')";
mysql_query($req) or die("erreur insertion");
?>
```

Insérez quelques tuples à la table contacts.

La partie connexion (jusqu'à l'instruction `mysql_select_db`) sera commune à toutes les pages qui accèdent à la base. Plutôt que de la copier/coller de multiples fois, il convient de l'écrire à part dans un fichier à part `connection.php`, puis de l'inclure en première ligne du script `insert.php`.

```
<?php
include("connection.php");
?>
```

Insérez quelques contacts dans la base de données à l'aide du formulaire contacts. Que se passe-t'il si vous insérez une chaîne de caractères sans "@" dans le champ email ?

#### 3.1 Vérification de l'adresse email à l'aide de Javascript

Pour ajouter une fonction javascript dans une page html, vous devez la définir dans l'entête de la page (entre les balises "`<head>`") :

```
<SCRIPT LANGUAGE="JavaScript">
function VerifForm(formulaire) {
var adresse = formulaire.email.value; //le champs "email" du form
var verif = /^[a-zA-Z0-9_-]+@[a-zA-Z0-9-]{2,}[.][a-zA-Z]{2,3}$/;
if (verif.exec(adresse) != null) {
formulaire.submit();
return(true);
}
```

```

else {
alert('Entrez une adresse e-mail valide!!!');
return(false);
}
}
</SCRIPT>

```

Puis appeler cette fonction lors de la validation du formulaire :

```
<form method="post" action="insert.php" onSubmit="return(VerifForm(this))">
```

Ajoutez cette fonctionnalité à votre formulaire et testez la. Est il possible de contourner cette sécurité?

### 3.2 Vérification de l'adresse email à l'aide de PHP

On peut écrire la même fonction de vérification d'une adresse email en PHP.

```

<?php
function VerifierAdresseMail($adresse)
{
    $Syntaxe='#^[\w.-]+@[\w.-]+\.[a-zA-Z]{2,6}$#';
    if(preg_match($Syntaxe,$adresse))
        return true;
    else
        return false;
}
?>

```

Où faut il ajouter cette fonction? Ajoutez cette fonctionnalité et testez la. Est il possible de contourner cette sécurité?

## 4 Il vous reste du temps?

### 4.1 Consultation des contacts présents dans la base

Les fonctions php *mysql\_query* et *mysql\_fetch\_array* permettent respectivement d'envoyer une requête et de traiter le résultat :

```

<?php
$req="SELECT * from contacts"; // requete
$res =mysql_query($req,$connection); // envoi de la requete

while ($tuple = mysql_fetch_array($res)) { //on récupere une ligne du resultat
    echo '$tuple['nom']'; //on récupère la valeur du champs nom de cette ligne
}
?>

```

Ecrire une page *list.php* qui affiche tous les contacts de la base, mis en page dans un tableau.

### 4.2 Clé primaire de la table contact

Avez vous pensé à définir une clé primaire dans la table contact? Si oui, que se passe t'il en cas de violation de cette contrainte d'intégrité?

Utilisez la fonction *mysql\_error* (<http://www.php.net/manual/en/function.mysql-error.php>) pour détecter cette violation. Plutôt qu'un message d'erreur qui nécessite de revenir en arrière à l'aide de la touche "back" de votre navigateur (peu ergonomique), revenez sur le formulaire de départ, avec les autres champs (nom et prenom) pré-remplis.

## 5 Chez vous

Vous pouvez installer Apache + PhP + MySql sur votre machine perso (WAMP sous windows, LAMP sous Linux, NAMP sous MacOS) Attention, pensez à modifier le php.ini pour mettre *display\_errors = On* pour avoir les messages d'erreurs. Car par défaut, le serveur est configuré en mode production ( *display\_errors = Off* ).