Université Toulouse I Capitole
IUT Rodez
2011

PAGES WEB DYNAMIQUES

EXERCICES

Les exercices présentés dans ce document illustrent des concepts souvent mis en pratique lors de la création de pages web dynamiques.

PWD01 HTML, ma première page

Exploiter le support HTML fourni en cours

```
<HTML>

 <HEAD>
  <TITLE>EXERCICE HTML SIMPLE </TITLE>
 </HEAD>

 <BODY>

  <CENTER><H1>Ma première page</H1></CENTER>

  J'écris du texte brut sans balise.
  <P><B>Voici un premier <BR>
  paragraphe en gras</B></P>
  <HR>
  Au-dessus, c'est une ligne horizontale

  <BR><BR>
  <I>J'écris maintenant en italique</I>

 </BODY>

</HTML>
```

PWD02 HTML, un peu plus avec des listes, liens et images

Plus d'infos HTML & compatibilités sur le site du w3c

exoHTMLliste.htm

```
<HTML>
<HEAD>
<TITLE>EXERCICE HTML : LISTES, IMAGES ET LIENS</TITLE>
</HEAD>
<BODY>

<BR>
<CENTER><H2>En HTML : Listes, images et liens</H2></CENTER>

<BR><BR>

<UL><H2>Mes beaux voyages</H2>
    <H4>
    <LI><A HREF="aubrac.htm">Expédition en Aubrac</A><BR><BR>
    <LI><A HREF="aveyron.htm">Aveyron sauvage</A>
    </H4>
</UL>

</BODY>

</HTML>
```

EXERCICE HTML : LISTES, IMAGES ET ...

En HTML : Listes, images et liens

Mes beaux voyages

- Expédition en Aubrac
- Aveyron sauvage

aveyron.htm

```
<HTML>
<HEAD>
<TITLE>EXERCICE HTML : LISTES, IMAGES ET LIENS</TITLE>
</HEAD>
<BODY>

<BR>
<CENTER><H1>Le long de l'Aveyron</H1></CENTER>

<BR><BR>
Une rivière traversant causses et gorges
<BR><BR><BR>
<IMG  HEIGHT="350"  ALT="Vertes prairies"  SRC="images\aveyron.jpg" >
<BR><BR>
La rivière Aveyron
<BR><BR>
  <A HREF="exoHTMLliste.htm"> Page précédente </A>

</BODY>

</HTML>
```

E HTML : LISTES, IMAGES ET ...

Le long de l'Aveyron

ivière traversant causses et gorges

La rivière Aveyron

Page précédente

PWD03 HTML & CSS, les guides de style

La charte graphique se définit dans le guide de style : le(s) fichier(s) css

style1.css

```
<STYLE type="text/css">

HTML {
}

BODY {
BACKGROUND-COLOR : LemonChiffon ;
FONT-FAMILY :  Verdana, Arial, Helvetica, Geneva, sans-serif;
}

P,UL,LI,TD {
FONT-SIZE : 100%;
COLOR : black;
}

H1 {
FONT-SIZE : 200%;
COLOR : red;
TEXT-ALIGN: center;
}

H2 {
FONT-SIZE : 150%;
FONT-STYLE : italic;
COLOR : red;
PADDING-LEFT:15px;
}

H3 {
COLOR : indigo;
PADDING-LEFT:45px;
FONT-WEIGHT : bold
}

H4 {
FONT-WEIGHT : normal;
PADDING-LEFT:90px;

}

a:link {color: green; text-decoration:underline;}
a:visited {color: gray; tex

</STYLE>
```

contenu_style1.htm

```
<HTML>

 <HEAD>
  <TITLE>EXERCICE CSS</TITLE>
  <LINK href="style1.css" rel="stylesheet" type="text/css">
 </HEAD>


<BODY>

<H1>Ma page avec le style 1.css</H1>

<BR>
Un petit texte brut et puis


<H2> Chapitre 1 : Le VTT en Aubrac</H2>
<H3> Descente de boraldes</H3>


<H4> Les boraldes sont des vallées profondes, entailles du plateau dévalant ju
</H4>

<BR><BR><BR><BR><A HREF=".\">Retour Page d'accueil</A>

</BODY>

</HTML>
```

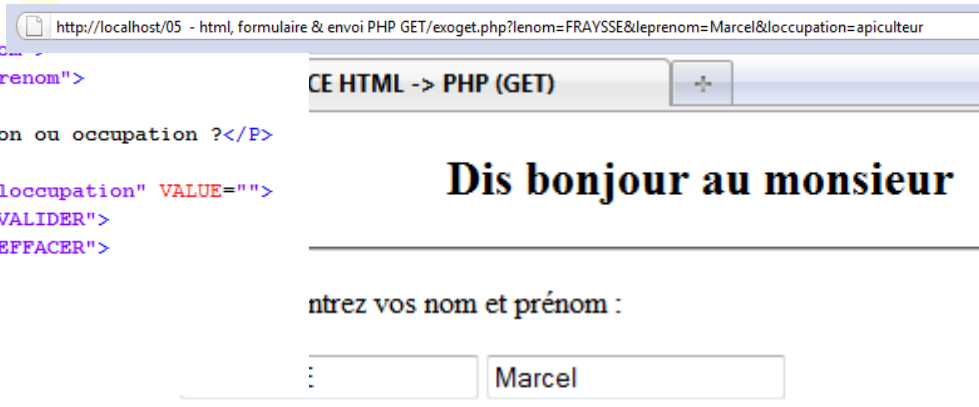
PWD04 HTML & PHP, formulaire et envoi d'informations à un script

Les formulaires sont codés en HTML, on envoie (GET ou POST) des variables à PHP

Toutes les docs sur PHP & ses API sont sur www.php.net

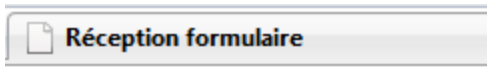
exoget.htm

```
<HTML>

 <HEAD>
  <TITLE>EXERCICE HTML -> PHP (GET)</TITLE>
 </HEAD>


<BODY>

 <CENTER><H2>Dis bonjour au monsieur</H2></CENTER><HR>
 <P>Bonjour, entrez vos nom et prénom :</P>

 <FORM ACTION=exoget.php METHOD=GET>

  <INPUT TYPE="TEXT" NAME="lenom">
  <INPUT TYPE="TEXT" NAME="leprenom">

  <P>Quelle est votre profession ou occupation ?</P>

  <INPUT TYPE="TEXT"   NAME ="loccupation" VALUE="">
  <INPUT TYPE="SUBMIT" VALUE="VALIDER">
  <INPUT TYPE ="RESET" VALUE="EFFACER">
 </FORM>

</BODY>

</HTML>
```



exoget.php

```
<HTML>

 <HEAD>
  <TITLE>Réception formulaire</TITLE>
 </HEAD>

 <BODY>

  <?php
  echo "<BR>Bonjour ",$_GET["leprenom"]," ",$_GET["lenom"],",<BR><BR>" ;
  echo "Alors comme ça, vous êtes ".$_GET["loccupation"]."." ;
  ?>

  <BR><BR><A HREF="exoget.htm">Retour Page d'accueil</A>

 </BODY>

</HTML>
```



Bonjour Marcel FRAYSSE,

Alors comme ça, vous êtes apiculteur.

Retour Page d'accueil

PWD05 PHP & MYSQL & l'accès aux données

L'api MySQL, un ensemble de fonctions pour communiquer avec une BD MySQL

Une bonne conception de la BD et notamment des contraintes d'intégrités est primordiale, penser clé primaire, clé étrangère & innodb…

PHPMySQL parcourt la base de données de géographie et présente par département la liste des villes.

PHPMySQL.php

```
<HTML>
 <HEAD>
  <TITLE>EXERCICE PHP MySQL</TITLE>
 </HEAD>

 <BODY>
<BR> <CENTER><H2>LES VILLES DU RESEAU INFO2</H2></CENTER>

<?php
// On pensera rapidement à définir utilisateur et mot de passe pour la bd
$connexion=mysql_connect("localhost","root","");
if (!$connexion)
    echo "Impossible de se connecter !";
mysql_selectdb("bd_geographie") or die("Connexion ratée");

// Ne pas rechigner à écrire des jointures, penser au tri pour l'affichage, éviter les SELECT *
$rqt_ville  = "SELECT d.nom AS nomdep, d.descriptif, v.code_insee, v.nom ";
$rqt_ville .= "FROM ville v, departement d ";
$rqt_ville .= "WHERE v.dep = d.numero ";
$rqt_ville .= "ORDER BY d.nom, v.nom" ;

/* Le curseur $lesvilles contient l'ensemble des villes de la base de données */
$lesvilles = mysql_query ( $rqt_ville ) ;

$departement_courant="";

/* Affichage des informations sur les villes */
while ( $uneville = mysql_fetch_object($lesvilles) )
{
 // Si dans la liste, on passe à un nouveau département, on crée un titre de rubrique avec le nom de ce département
 if  ($uneville->nomdep != $departement_courant )
 {
    //Récupération, mise en majuscule et affichage du nouveau département
    $departement_courant = $uneville->nomdep ;
    $depMajuscule = strtoupper($departement_courant);
    echo "<H2><BR>$depMajuscule  </H2>";
 }
 // Affichage de chaque ville avec le lien hypertexte vers son détail
 echo "<A HREF=\"pwd05.php?laville='$uneville->code_insee'&ledep=$departement_courant \">$uneville->nom </A><BR>";
}
mysql_close();

?>

 </BODY>
</HTML>
```

Le clic sur une ville déclenche PHPMySQL_details en lui passant pour paramètre une ville. Le script analyse les informations & récupère en base de données les infos adéquates (ex. 3 meilleures photos) et les affiche au besoin.

Dans la BD, on va pouvoir définir des attributs selon les fonctionnalités (ex att importance)

Dans l'application, on pourra afficher dynamiquement des rubriques : selon le résultat de la BD : IS_NULL, mysql_numrows..

PHPMySQL.php | PHPMySQL_details.php

```
<HTML>
 <HEAD>  <TITLE>EXERCICE PHP MySQL, plus de détails</TITLE>  </HEAD>

 <BODY> <BR>

<?php
// Zone de définition des constantes
DEFINE("NB_MAX_PHOTOS",3);

$connexion=mysql_connect("localhost","root","");
if (!$connexion)
    echo "Impossible de se connecter !";
mysql_selectdb("bd_geographie") or die("Problème sur cette base");

// Sécurité pour éviter les intrusions HTML
$idvilledétail = htmlentities($_GET["laville"]);
$ledep = htmlentities($_GET["ledep"]);

$rqt_villedétail  = "SELECT nom, dep, pop, alt, descriptif
                     FROM ville
                     WHERE code_insee = $idvilledétail ;" ;

/* Le curseur $laville contient les informations de la seule ville concernée*/
$laville = mysql_query ($rqt_villedétail) ;

while ( $infosville = mysql_fetch_object($laville) )
{
    echo "<CENTER><H1>$infosville->nom </H1></CENTER><HR>" ;
    echo "Département : $ledep<BR><BR>" ;

    //Test présence de valeur avant affichage
    echo "Population : ";
    if (IS_NULL($infosville->pop) ) echo "inconnue";    else    echo $infosville->pop." habitants<BR><BR>";
    echo "Altitude : ";
    if (IS_NULL($infosville->alt) ) echo "inconnue";    else    echo $infosville->alt." mètres<BR><BR>";
    echo "Descriptif : ";
    if (IS_NULL($infosville->descriptif) )  echo "inconnue";    else    echo $infosville->descriptif."<BR>";

    /* Requête qui retrouve les photos associées à une ville */
    $rqt_photos = "SELECT titre, fichier FROM photos WHERE codeville=".$idvilledétail." ORDER BY importance DESC"
    /* Possibilité de limiter le nombre de lignes retournées grâce à l'intruction MySQL LIMIT 0, lenombrevoulu */
    /* Le curseur $lesphotos contient l'ensemble des photos associées à une ville */
    $lesphotos = mysql_query ( $rqt_photos ) ;
```

```php

    /* Dans le cas où il existe des photos pour cette ville, on les présente */
    if ( mysql_num_rows($lesphotos )) {
            echo "<H4>En image : </H4></BR>" ;
        $nbphotosaffichees = 0 ;
        $nblibellesaffiches = 0 ;
        $chemin = "./images" ;
        /* 2 boubles pour générer un affichage ligne des photos/ ligne des libellés dessous */
        /* un tableau permet de stocker les libellés sans aller parcourir une deuxième fois la relation */
        while ( $unephoto = mysql_fetch_object ( $lesphotos ) AND ( $nbphotosaffichees < NB_MAX_PHOTOS ) )
            {
            $adressephoto = $chemin."/".$unephoto->fichier.".jpg" ;
            echo "<IMG SRC = \"$adressephoto \" heigth=\"150\" WIDTH=\"220\" >" ;
            echo "  " ;
            /* Stokage des libellés pour la boucle suivante */
            $tablibelles[$nbphotosaffichees]=$unephoto->titre ;
            $nbphotosaffichees++ ;
            }
        echo "<BR>" ;
        /* Affichage des libellés tant qu'il y a des photos */
        while ( $nblibellesaffiches < $nbphotosaffichees )
            {
            echo "        ".$tablibelles[$nblibellesaffiches]."      &
            $nblibellesaffiches++ ;
                }
        echo "<BR>" ;
    }
echo "<BR><HR>" ;
}

/* Fermeture de la base de données */
mysql_close() ;
?>
```

PWD06 PHP & la séparation Présentation / Traitement / Données

Impératif de séparer le code par nature => découpage en fonctions et en différents fichiers

Pour faciliter la maintenance on suivra une architecture de conception de notre code tel le MVC : Modèle Vue Contrôleur

affichage.inc.php | bd.inc.php | pilote.php

```php
<?php

include "affichage.inc.php" ;
include "bd.inc.php";

// On prépare une variable pour recevoir un ensemble de villes puis pour les communiquer
$ensembleVille = array();

//On va chercher dans la BD les données sur les villes selon nos critères (aucun)
$ensembleVille = cherche_bd_lesvilles_toutes() ;

//On choisit une fonction d'affichage pour présenter ces villes ($ensembleVille) en résultat
affiche_tableau ($ensembleVille);

?>
```

affichage.inc.php | bd.inc.php | pilote.php

```php
<?php
function affiche_tableau ($unTab)
/*présente sous la forme d'un cadre (un tableau graphique, une matrice) le tableau de données $unTab fourni en argument
(présentation simple : sans champ en gras ou autres formats différents dans le tableau) */
{

/* On prépare notre tableau graphique en comptant le nombre de lignes */
$nb = count($unTab);

echo "<BR>";

echo "<TABLE BORDER=1 >";
for ($i =0; $i < $nb; $i ++)
 {
    echo "<TR>";

    /* On va parcourir la ligne entière grace à foreach ($unTab[$i] as $cle => $valeur) */
    foreach ($unTab[$i] as $cle => $valeur)
        echo "<TD>".$unTab[$i][$cle]."</TD>";

    echo "</TR>";
  }
echo "</TABLE>";
}

?>
```

affichage.inc.php | bd.inc.php | pilote.php

```php
<?php
function cherche_bd_lesvilles_toutes ()
/* cherche les informations sur les différentes villes et les stocke dans une structure de données tableau */
/* le tableau des villes sera retournée a la fin de la function */
{
$tabvilles = array();

$host = "localhost";
$user = "root" ;    /*A changer !! */
$pwd  = "";         /*A changer !! */

$connexion = mysql_connect($host, $user, $pwd)
    or die("Impossible de se connecter : " . mysql_error());

$labase = mysql_select_db("bd_geographie", $connexion);
if (!$labase) {
   die ('Impossible de sélectionner la base de données : ' . mysql_error());
}

$requete= "SELECT ville,codeinsee,altitude,population FROM villes";
$resultat=mysql_query($requete);
$i = 0;
/*A la première ligne du tableau , on place le titre des colonnes,
On aurait pu travailler avec une "table d'association NomTechnique<->NomClair" */
$tabvilles[$i]['ville']      = "Ville";
$tabvilles[$i]['codeinsee'] = "Codeinsee";
$tabvilles[$i]['altitude']   = "Altitude";
$tabvilles[$i]['population']= "Population";
$i ++;

while($uneVille=mysql_fetch_object($resultat))
{
    $tabvilles[$i]['ville']      = $uneVille->ville;
    $tabvilles[$i]['codeinsee'] = $uneVille->codeinsee;
    $tabvilles[$i]['population']= $uneVille->population;
    $tabvilles[$i]['altitude']   = $uneVille->altitude;
    $i ++;
}

mysql_close($connexion);
return $tabvilles;
}
?>
```

PWD07 PHP & les sessions

Des sessions pour gérer identification/ authentification à des pages sécurisées

Des sessions pour conserver, de page en page, des variables sur un site, par exemple un panier

questions_avec_session.php | quizz.php

```php
<?php
session_start();
$_SESSION['verif']="ok";
echo "<HTML>";
echo "   <HEAD> ";
echo "      <TITLE> ENIGMAWEB </TITLE> </HEAD>";
echo "  <BODY>";
echo "      <CENTER><H2>ENIGMAWEB, LE JEU QU'IL TE FAUT !!</H2></CENTER><HR>";

  /* Connexion à MySQL, avec un compte et profil par défaut */
  $connect = mysql_connect ( "localhost", "root", "" )
                or die ("Connexion impossible") ;

  /* Connexion à la base de données */
  mysql_select_db("bd_enigmaweb") ;

  /* La requête préparée est statique, le tri facilite la consultation */

  $rqt_detail = "SELECT quizz.numquizz qinum, quizz.intitule qiint
                    FROM quizz
                      ORDER BY quizz.numquizz" ;

 /* Le curseur $lesvilles contient l'ensemble des villes de la base de données */
  $lesdetails = mysql_query ( $rqt_detail  ) ;
  $quizzcourant="";

echo "<FORM   METHOD= \"POST\"   ACTION=\"questions.php\">";

  /* Affichage des informations sur les villes */

echo "Ton pseudo :  <INPUT TYPE =\"TEXT\"   MAXLENGTH=40    VALUE=\"\"  NAME=\"pseudo\">   <BR><BR>";
echo "Ton age :        <INPUT TYPE =\"TEXT\"    MAXLENGTH=40    VALUE=\"\"  NAME=\"age\">
echo "<INPUT type=hidden name=\"verif\" value=\"ok\">";
echo "<BR><BR>Choisis ton quizz :";
echo "<SELECT name=\"rad_quizz\">";
while ( $uneligne = mysql_fetch_object ( $lesdetails)  )
  {              echo " <OPTION  VALUE=  \"".$uneligne->qinum."\">".$uneligne->qiint;
}
echo "</SELECT>";

echo "<BR><BR><BR><BR><INPUT TYPE =\"SUBMIT\" VALUE=\"VA JOUER\"><BR>";

echo "</FORM>";


  /* Fermeture de la base de données */
  mysql_close() ;
?>

</BODY>
</HTML>
```

questions_avec_session.php | quizz.php

```php
<?php
session_start();

// Version test & explications du 07/01/2011

echo "<HTML>";
echo "   <HEAD> ";
echo "      <TITLE> QUESTIONNAIRES </TITLE> </HEAD>";
echo "  <BODY>";
echo "      <CENTER><H2>ENIGMAWEB, LE JEU QU'IL TE FAUT !!</H2></CENTER><HR>";


if (isset( $_SESSION['verif']))
{
 if  (!empty($_POST['pseudo']) AND !empty($_POST['age']))
 {
 $lepseudo = $_POST['pseudo'];
 $lage=$_POST['age'];

 /* Connexion à MySQL, avec un compte et profil par défaut */
  $connect = mysql_connect ( "localhost", "root", "mysql" )
                  or die ("Connexion impossible") ;

  /* Connexion à la base de données */
  mysql_select_db("bd_enigmaweb") ;


  // Ajout d'un joueur
  //manipulation du auto-increment
  $rqt_nouveaujoueur ="INSERT INTO joueur (pseudo, age) VALUES ( '".$lepseudo."',".$lage.");";
  echo $rqt_nouveaujoueur ;

  $statut_joueur=mysql_query ( $rqt_nouveaujoueur  ) ;
  $lenumjoueur =  mysql_insert_id() ;
  echo $lenumjoueur ;

  //
  $quizz_choisi = $_POST["rad_quizz"];
  $rqt_quizztit= "SELECT quizz.intitule qiint
                  FROM quizz
                  WHERE quizz.numquizz ='".$quizz_choisi."';";

  $lequizz = mysql_query (  $rqt_quizztit ) ;
  $laligne = mysql_fetch_object ( $lequizz  );

  $lequizznom = $laligne->qiint;
  $lequizznum = $quizz_choisi;

  echo "<CENTER><H2>".$lequizznom."</H2></CENTER><HR>";
  echo "Bon courage ".$_POST["pseudo"];
```

PWD08 PHP & les javascripts

Un langage pour l'interactivité côté client => A utiliser sous forme de boîte à outils

Des sites spécialisés dans le recensement de fonctions & widgets javascript

```
<BODY>

<!---Le form f1--->

<FORM NAME="f1" METHOD="POST" ACTION="envoi.php">

Nom      : <INPUT TYPE="TEXT" NAME="nom">
<BR>
Prénom : <INPUT TYPE="TEXT" NAME="prenom">
<BR>
Mail      : <INPUT TYPE="TEXT" NAME="mail">
<BR>
  <INPUT TYPE="SUBMIT" NAME="SUBMIT" VALUE="Envoyer" ONCLICK="return verif_form()">


<!--Le script-->

<SCRIPT LANGUAGE="JavaScript">
function verif_form(){
   var Nom = document.f1.nom.value;
   var Prenom = document.f1.prenom.value;
   var Mail = document.f1.mail.value;
   if((!Nom) || (!Prenom) || (!Mail) ){
     alert("Toutes les informations sont obligatoires.");
     return false;
}
    if (Mail.indexOf("@",0)<0 )
    {
     alert("Vérifier l'adresse mail svp");
     return false;
    }
}
</SCRIPT>
```

PWD09 **PHP & la sécurité**

Sécurité & BD

On va **créer des utilisateurs**, définir des profils, **attribuer des droits** sur des éléments du schéma de BD. Et les scripts PHP accèderont à la base de données via ces identifications personnalisées.

On empêchera les accès BD sur les valeurs par défaut des comptes et mot de passe. Les comptes « root », « postgres » et les mots de passes courants «», mysql, pgsql seront proscrits.

Des pages réservées à **l'administration du site n'auront pas le même login** que les pages de consultation.

On évitera des attributs aux noms classiques tel admin, mdp, password, login…

Les mots de passe doivent être sécurisés : 10 caractères au moins avec des min MAJ nombres & caractères spéciaux.

Session

Les sessions constituent une façon de protéger le dialogue entre l'internaute et le serveur web.

Elles empêchent notamment que l'on puisse accéder intempestivement à une page sans être connecté.

Durée de vie d'une session ou d'un cookie

Il est possible de limiter dans le temps la connexion.

Exemple de la consultation des comptes bancaires

Chiffrement d'un mot de passe, le hachage

On ne doit pas stocker les mots de passe de manière lisible ou décodable.

Le hash = procédé de chiffrement irréversible

Sha-1 / 256 : principales techniques possibles et PHP propose la fonction correspondante

Secure Hash Algorithm de la National Security Agency États-Unis

Formulaires : Sécurité de passage des paramètres

Pour passer des paramètres d'une page à une autre, on exploite les méthodes GET et POST.

Syntaxe : script.php?var1=val1&var2=val2… Méthode GET & transmission via l'URL

La méthode POST les enlève.

Utiliser le POST et non le GET évite donc une interception des valeurs passées et la réécriture sous cette forme-là pour l'entrée des valeurs dans une base de données.

Formulaires : Attention aux champs cachés

Attention aux champs cachés :

< input type = "hidden" name = "".."" value = ".." >

Ils servent à passer une valeur via un formulaire sans qu'un utilisateur ne la voit.

Exemple : page1 -> page2 -> page3 & je veux exploiter des infos de page1 dans page3

Il ne faut pas mettre de valeur confidentielle dans un champ caché car l'utilisateur peut y avoir accès en demandant l'affichage du code source (le code HTML) de la page.

Pas de magicquote mais des addslashes et pas d'intrusions

Un problème de sécurité classique est : $rqt = SELECT .. WHERE password = ' " *RECUP PARAMETRE* "

Le langage HTML utilise des caractères spéciaux les < et > par exemple. Les requêtes SQL vont récupérer des arguments. **Il faut s'assurer que la récupération extérieure de valeurs n'interfère pas dans le code écrit dans les pages ou les accès bd.**

Lorsque l'on permet à des utilisateurs de soumettre un contenu dans un site, on donne un à ce site web. Il faut donc prendre les précautions appropriées, par exemple on va annihiler l'interprétation de balises html venant de saisies utilisateur. Des fonctions existent pour cela.

addslashes ( )	Ajoute des antislashs devant les apostrophes ('), les guillemets ("), les backslashs (\) et le caractère NULL. Cela évite une interprétation pouvant rendre le code illisible.
mysql_real_escape_string ( )	Protège les caractères spéciaux d'une commande SQL
htmlspecialchars ( )	Convertit les caractères spéciaux en entités HTML
htmlentities ( )	Identique à la fonction htmlspecialchars(), sauf que tous les caractères qui ont des équivalents en entités HTML sont effectivement traduits : accents, < > ...

Exemples de tentatives d'intrusion par du code :

⇨ Coller à des paramètres une chaîne du type « ; DROP TABLE newsletter »

⇨ SELECT Login, Password FROM identification WHERE Login=' 'or ''='# AND Password=' $password '
Ce qui revient à effectuer une requête de la forme :
SELECT Login, Password FROM identification WHERE Login= '' or ''=''

⇨ RECUP PARAMETRE reçoit *' OR password LIKE '%";*

La requête $rqt = SELECT .. WHERE password = *' ' OR password LIKE '%";* permet alors d'obtenir un accès.

Sécurité d'accès à une page, l'authentification HTTP

Il est possible avec le langage PHP (cf. fonction header) de travailler sur les entêtes des messages HTTP, le protocole utilisé pour communiquer entre le client et le site web. La fonction header

```
<?php

//Valeurs cachées dans un fichier crypté sur le serveur ou stockées dans une BD ; placer ces valeurs ailleurs
$user = "rouquier";
$pwd = "marcel";

function auth(){
   $realm="Authentification PHPindex";
   header("WWW-Authenticate: Basic realm='".$realm."'");
   header("HTTP/1.0 401 Unauthorized");
   echo "Vous ne pouvez pas accéder à cette page";
   exit;
}

if( !isset($_SERVER['PHP_AUTH_USER']) && !isset($_SERVER['PHP_AUTH_PW']) ) {
   auth();
}
else {
   if( $_SERVER['PHP_AUTH_USER']==$user && $_SERVER['PHP_AUTH_PW']==$pwd ) {
      echo "Bienvenue sur ce site";
   }
   else{
      auth();
   }
}
?>
```

Et les fichiers de protection .htaccess / .htpasswd

.htaccess : ce fichier contiendra l'adresse du .htpasswd et quelques autres options

.htpasswd : il contiendra une liste logins / pwd des personnes autorisées à accéder aux pages.

.htaccess :

```
AuthName "PageProtection"
AuthType Basic
AuthUserFile /cheminAbsolu/.htpasswd
Require valid-user
```

.htpasswd :

```
login:pavkLMUhOiqd3
```

SSL

Secure Socket Layer pour assurer la confidentialité des échanges de données

PWD10 Autres cas pratiques

PHP & la gestion des dates

- Attention au format de stockage des dates aaaa-mm-jj
- Attention au format d'affichage des dates jj/mm/aaaa
- Utiliser la fonction explode pour reconstituer une date au bon format

  Ex :

```
$date_tab = explode ("/", $date_ihm );
$date_php= $date_tab[2]."-".$date_tab[1]."-".$date_tab[0]   ;
```

PHP & la lecture d'un .ini

- .ini, l'intérêt des variables de configuration
- Lecture de fichier, cf. fonctions du langage C
- Lecture de ligne et extraction des données selon format (cf. explode)

Passage de tableaux et sérialisation

- Communication entre scripts et passage des variables
- Tableau de variables & fonctions php serialize()/unserialize()

PHP & MYSQL & la gestion d'erreurs BD

- L'envoi d'une requête de modification (INSERT, UPDATE, DELETE) doit s'accompagner d'un contrôle du code de retour de la BD.
- $err=mysql_errno() ;
- Dans le programme appelant, gérer les cas d'erreur , par exemple si erreur sur clé primaire 'num 1022)
- http://dev.mysql.com/doc/refman/5.5/en/error-messages-server.html

…