

PHP - Interfaçage avec un SGBD (1)

■ Principales étapes (ex. de *mySQL*)

- 1 se connecter au serveur de la BD :

```
mysql_connect($host, $user, $password)
```

- 2 choisir la base : `mysql_selectdb($bdd)`

- 3 exprimer une requête : `$id_rq=mysql_query($ch_req)`

- 4 lire une ligne du résultat (pour une requête sélection) :

```
mysql_fetch_row($id_rq) OU
```

```
mysql_fetch_object($id_rq)
```

- 5 se déconnecter : `mysql_close()`

1

PHP - Interfaçage avec un SGBD (2)

■ Exemple

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01
Transitional//FR">
<html>
<head><title>Promotion</title></head>
<body>
<table border="1" cellpadding="0" cellspacing="0">
<tr><th>Nom</th><th>Adresse mail</th></tr>
<?php
//Paramètres de connexion
$host="localhost"; //machine hébergeant le script
$user="lo"; //votre login
$password="cmoi"; //mot de passe
$bdd="etudiants"; //nom de la base
mysql_connect($host, $user, $password) or
die("erreur de connexion au serveur");
mysql_selectdb($bdd) or
die("erreur de connexion à la base");
```

2

PHP - Interfaçage avec un SGBD (3)

```
// Création et envoi de la requête
$ch_req ="SELECT nom,mail FROM Promo ORDER BY nom";
$result = mysql_query($ch_req);

// Récupération des résultats
while($ligne = mysql_fetch_row($result)){
    $nom=$ligne[0];
    $mail=$ligne[1];
    print "<tr>\n<td>";
    print "<a href=\"\$mail\">$nom</a>";
    print "</td>\n<td>$mail</td>\n</tr>\n";
}
// Déconnexion de la base de données
mysql_close();
?>
</tr>
</table>
</body>
</html>
```

3

PHP - Sessions (1)

■ Utilité

- ◆ Transmettre des variables de page en page (ex. login)
- ◆ Conserver des informations le temps d'une visite

■ Mécanisme

- ◆ Session = fichier sur le serveur
- ◆ Accessible de tous les scripts

■ Caractérisée par

- ◆ un identifiant généré à la création de la session
- ◆ une période de validité limitée
 - maximale
 - détruite à la fermeture du navigateur du client

4

PHP - Sessions (2)

■ Tableau et fonctions

◆ `$_SESSION`

tableau associatif global contenant toutes les variables de session pour la sessions courante (clés = noms)

◆ `session_start()`

- démarre une session ou en rappelle une existante
- doit être appelée dans tous les scripts, avant tout écho

◆ `session_destroy()`

- détruit une session (mais pas les variables associées)
- destruction explicite des variables (*ex.* deconnexion.php)

```
$_SESSION = array();
```

5

PHP - Compléments (1)

■ Fonctions de protection de chaînes

◆ `addslashes()`

- Échappe les caractères (' , " , \ , NUL) qui doivent l'être pour être utilisés dans une requête de base de données
- Fonction inverse : `stripslashes()`
- *Rem.* la directive `magic_quotes_gpc` est à *on* par défaut, et elle appelle `addslashes()` sur toutes les données GET, POST, COOKIE

◆ `htmlspecialchars()`

- Convertit les caractères spéciaux HTML (& , " , ' , < , >) en leur entité HTML (& amp; , & quot; , & #039; , & lt; , & gt;)
- `htmlspecialchars()` traduit en entité HTML tous les caractères ayant un équivalent. Inverse : `htmlspecialchars_decode()`

6

PHP - Compléments (2)

■ Dates et heures

◆ Timestamp UNIX

- nb secondes écoulées depuis le 01/01/1970 00:00:00 GMT
- générer un timestamp : `time()`, `mktime()` ou `strtotime()`

◆ Fonctions de dates/heures

- ajout de périodes : `strtotime()`
- formatage : `date()` ou `strftime()`
- test de validité de date : `checkdate()`
- retour d'un tableau associatif (clés)
 - `getdate (seconds, minutes, hours, mday, wday, mon, year, yday, weekday, month)`
 - `gettimeofday (sec,usec, minuteswest, dsttime)`

7

PHP - Compléments (3)

■ Affichage de variables complexes

- ◆ `print_r` affiche récursivement les clés et valeurs d'un tableau
 - Rem.* place le pointeur en fin de tableau (faire un `reset()`)
- ◆ `var_dump` affiche les types en plus
- ◆ `var_export` retourne le code PHP utilisé pour générer une variable

■ Divers

◆ redirection d'une page (**avant tout écho**)

```
header('location:http://www.site.fr/page.html');
exit(); // évite de stocker la page dans l'historique
```

8