

Enoncé du cinquième TD/TP de l'enseignement  
**technologies du Web**  
d'IUP GMI 2/3  
**(PHP - MySQL)**

Pierre Pompidor

3 mai 2001

## Objectifs fonctionnels

Création d'une base de QCM accessible par le Web

- L'administrateur pourra
  - saisir des QCM (questionnaires à choix multiples) à partir du navigateur  
(on pourra envisager un système d'aide à la création de QCM à partir de cours, de sujets de TD ou de TP)
  - les mémoriser dans une base de données (nous utiliserons MySQL)
- Les utilisateurs pourront
  - répondre aux QCM
  - connaître le résultat de leur évaluation

## Objectifs pédagogiques

Création de pages **dynamiques** au moyen de **scripts "pages à trous" PHP**

- vous initier à l'écriture de scripts PHP
- réaliser la connexion d'un script PHP avec une base de données

# Gestion du SGBD MySQL

## Connexion à un serveur MySQL (un démon mysqld s'exécute)

```
mysql -h machine -u utilisateur -p [base_de_données]
```

-h : machine hôte

-u : utilisateur MySQL

-p : le mot de passe MySQL sera exigé

```
mysql -h a01 -u user1 -p
```

```
Enter password : *****
```

Certaines installations de MySQL permettent aux utilisateurs de se connecter comme "anonymes" sur le serveur s'exécutant sur la machine locale. Si c'est le cas, vous pouvez invoquer mysql sans options :

```
shell> mysql
```

Pour vous déconnecter tapez QUIT après le prompt

```
mysql> QUIT
```

Bye

(ou control-D).

Les commandes SQL suivantes peuvent être exécutées en interactif ou dans un script PHP. N'oubliez pas de clore les requêtes SQL par un ;.

## Création d'une base de données (create database)

```
CREATE DATABASE mabase;
```

Quelques types :

TINYINT	1 byte	SMALLINT	2 byte
MEDIUMINT	3 bytes	INT	4 bytes
BIGINT	8 bytes		
FLOAT	4 bytes	DOUBLE	8 bytes
REAL	8 bytes		
DATE	3 bytes	DATETIME	8 bytes
TIME	3 bytes	YEAR	1 byte
CHAR(M)	M bytes, 1 <= M <= 255		
VARCHAR(M)	L+1 bytes, où L <= M et 1 <= M <= 255		

## Chargement d'une base de données (use)

```
mysql> USE mabase;
```

ou directement lors de la connexion :

```
mysql -h a01 -u user1 -p mabase
```

## Listing des tables existantes (show)

```
SHOW TABLES;
```

## Description de la structure d'une table (describe)

```
DESCRIBE matable;
```

## Insertion d'un tuple (insert)

```
INSERT INTO matable VALUES (...);
```

## Importation des données d'un fichier texte (load data local infile)

```
LOAD DATA LOCAL INFILE monfichier INTO TABLE matable;
```

**Attention, MySQL ne gère pas les clefs étrangères.**

# Exemples "types" de scripts PHP

## Rappel succinct de la syntaxe de PHP

Déclaration des variables à la volée :	<code>\$variable = ...</code>
Transtypage par un opérateur de transtypage :	<code>chaîne = (string)entier;</code>
-> par défaut les variables sont locales	
Accès aux variables d'environnement :	directement par leurs noms
Accès aux variables des formulaires :	directement par leurs noms
Opérateurs arithmétiques et de comparaison du C	
Expressions régulières :	<code>preg_match ("expression reg.", \$chaîne);</code>
-> motifs capturés :	<code>\$matches[1], \$matches[2], ...</code>
Création des tableaux simples :	à la volée <code>\$tab[\$i] = ...</code>
Création des tableaux associatif :	à la volée <code>\$tableau["..."] = ...</code>
Structures conditionnelles et itératives du C +	
	<code>foreach (\$tableau as \$val) {...} (en php4)</code>
	<code>foreach (\$tableau as \$clef =&gt; \$val) {...} (en php4)</code>
	<code>while (list(\$clef,\$valeur) = each(\$tableau)){...}</code>

## Insertion d'un tuple dans une base de données PHP

```
<html>

<head>
<title> Voici un exemple d'insertion d'un tuple dans une base de données en PHP
</head>

<body>
<?php
    echo "<h2> Le nom et le prénom que vous avez saisis dans le formulaire sont $nom et $prenom </h2>";

    // connexion à mySQL et à "mabase"
    $lien = mysql_connect( "localhost", "user", "pass") or die ( "Erreur de connexion sur mySQL");
    mysql_select_db("mabase", $lien) or die ( "Erreur de connexion sur la base");

    // Si l'insertion réussie, $resultat vaut vrai, sinon faux
    $requete = "INSERT INTO matable (nom, prenom) VALUES ('$nom', '$prenom')";
    $resultat = mysql_query ($requete);

    if ($resultat) {echo "<p>Vos données son bien enregistrées.";}
    else
        {echo "<p>Warning: erreur d'insertion dans la base de données.</p>\n";
          // Retour de l'erreur PHP
          echo "<p>Erreur retournée par mySQL : " . mysql_error() . "\n";
        }
    }
?>

</body>
</html>
```

## Lecture d'une table dans une base de données en PHP

```
<html>

<head>
<title> Voici un exemple de lecture d'une table dans une base de données en PHP
</head>

<body>
<?
  // connexion à mySQL et à "mabase"
  $lien = mysql_connect( "host", "user", "pass") or die ( "Erreur de connexion sur mySQL");
  mysql_select_db("mabase", $lien) or die ( "Erreur de connexion sur la base");

  $requete = "select * from matable";
  $resultat = mysql_query($requete);
?>

<table bgcolor='lightyellow'>
<tr><th>Nom</th><th>Prénom</th></tr>
<?
  $i = 0;
  while ($i < mysql_num_rows($resultat))
  {
    echo "<tr><td>"; echo mysql_result($resultat,$i,'nom');
    echo "</td><td>"; echo mysql_result($resultat,$i,'prenom');
    echo "</td></tr>";
    $i++;
  }
?>
</table>

// OU

<table bgcolor='lightyellow'>
<tr><th>Nom</th><th>Prénom</th></tr>
<?
  while ($row = mysql_fetch_row($resultat))
  {
    echo "<tr>";
    for ($i=0; $i<mysql_num_fields($resultat); $i++)
    {
      echo "<td> $row[$i] </td>";
    }
    echo "</tr>";
  }
?>
</table>

</body>
</html>
```