

Atelier 5 : Technologie Web: PHP & MySQL

Série 5

L'enregistrement dans une base de données

PHP et MySQL

Premières requêtes

Les informations vous permettant de vous y connecter (nom du serveur, nom de la base, identifiant, mot de passe) doivent vous être fournies par votre hébergeur.

Si vous travaillez sur votre propre machine, les informations suivantes (qui correspondent à celles par défaut) devraient convenir...

- Serveur SGBD : localhost.
- Nom de la base : test.
- Identifiant : root.
- Mot de passe : vide.

Avant d'envoyer des requêtes, il convient tout d'abord de se connecter à MySQL via `mysqli` sur la base de données `test` :

```
<?php
//simple conection:
$liendb = mysqli_connect("localhost", "root", "", "test") or
die("Error " . mysqli_error($liendb));
?>
```

Voici comment on doit faire pour se connecter à MySQL via l'extension **orientée objet** PDO sur la base `test` :

```
<?php
//simple conection:
$liendb = new pdo("mysql:host=localhost;dbname=test", "root", "");
//Le premier paramètre (qui commence par mysql) s'appelle le DSN : Data Source Name.
//C'est généralement le seul qui change en fonction du type de base
//de données auquel on se connecte.
?>
```

Remarque : Lorsque votre site sera en ligne, vous aurez sûrement un nom d'hôte différent ainsi qu'un login et un mot de passe comme ceci :

```
<?php
$liendb = new PDO('mysql:host=sql.hebergeur.com;dbname=maBase;charset=utf8', 'etudinat.isil',
'isil2016');
?>
```

Pour tester la présence d'erreurs, voici le code à exécuter

```
<?php
//connexion
try
/*PHP essaie d'exécuter ces instructions de ce bloc try*/
{
$liendb = new PDO('mysql:host=localhost;dbname=test', 'root', '');
//crée un objet $liendb qui représente la connexion à la base de données.
}
catch (Exception $e)
{
die('Erreur : ' . $e->getMessage());
//S'il y a une erreur, PHP rentre ici pour arrêter l'exécution de la page
//en affichant un message décrivant l'erreur.
}
?>
```

Vous pouvez utiliser la requête suivante pour choisir la base de données avec laquelle vous allez travailler :

```
mysqli_select_db("test");
```

Il faut ensuite formater une requête en SQL :

```
$sql = "CREATE TABLE eleve (
ideleve int(10) unsigned NOT NULL auto_increment,
nom varchar(64) NOT NULL default '',
prenom varchar(64) NOT NULL default '',
adresse varchar(128) NOT NULL default '',
ville varchar(64) NOT NULL default '',
cp varchar(8) NOT NULL default '',
pays varchar(32) NOT NULL default 'france',
sexe varchar(8) binary NOT NULL default '',
naissance date NOT NULL default '0000-00-00',
taille int(10) unsigned NOT NULL default '0',
email varchar(64) NOT NULL default '',
telephone varchar(16) NOT NULL default '',
lv varchar(16) NOT NULL default '',
PRIMARY KEY (ideleve),
KEY nom (nom)
)";
```

Envoyez-la ensuite à la base :

```
mysqli_query($liendb,$sql);
```

Lorsque vous n'avez plus besoin de la base de données dans le *script*, il est conseillé de fermer la connexion au serveur :

```
mysqli_close($liendb);
```

Voici les étapes à suivre pour exécuter une requête :

Étape 1 : création d'une connexion au serveur.

Étape 2 : sélection de votre base.

Étape 3 : préparation de la requête.

Étape 4 : envoi de la requête à la base.

Étape 5 : fermeture de votre connexion.

Exemple avec Style procédural :mysql_

```
<?php
//conection:
$liendb = mysqli_connect("localhost", "root", "", "test");

/* Vérification de la connexion */
if (mysqli_connect_errno()) {
    printf("Échec de la connexion : %s\n", mysqli_connect_error());
    exit();
}

/* "Create table eleve" */
$sql = "CREATE TABLE eleve (
ideleve int(10) unsigned NOT NULL auto_increment,
nom varchar(64) NOT NULL default '',
prenom varchar(64) NOT NULL default '',
adresse varchar(128) NOT NULL default '',
ville varchar(64) NOT NULL default '',
cp varchar(8) NOT NULL default '',
pays varchar(32) NOT NULL default 'france',
sexe varchar(8) binary NOT NULL default '',
naissance date NOT NULL default '0000-00-00',
taille int(10) unsigned NOT NULL default '0',
email varchar(64) NOT NULL default '',
telephone varchar(16) NOT NULL default '',
lv varchar(16) NOT NULL default '',
PRIMARY KEY (ideleve),
KEY nom (nom)
)";

if (mysqli_query($liendb, $sql) == TRUE) {
    printf("Table eleve créée avec succès.\n");
}

/*Fermeture de la connexion*/
mysqli_close($liendb);
?>
```

Même exemple avec Style orienté objet :mysqli_

```
<?php
//connexion
$liendb = new mysqli("localhost", "root", "", "test");
```

```

/* Vérification de la connexion */
if (mysqli_connect_errno()) {
    printf("Échec de la connexion : %s\n", mysqli_connect_error());
    exit();
}
/* "Create table eleve" */
$sql = "CREATE TABLE eleve (
ideleve int(10) unsigned NOT NULL auto_increment,
nom varchar(64) NOT NULL default '',
prenom varchar(64) NOT NULL default '',
adresse varchar(128) NOT NULL default '',
ville varchar(64) NOT NULL default '',
cp varchar(8) NOT NULL default '',
pays varchar(32) NOT NULL default 'france',
sexe varchar(8) binary NOT NULL default '',
naissance date NOT NULL default '0000-00-00',
taille int(10) unsigned NOT NULL default '0',
email varchar(64) NOT NULL default '',
telephone varchar(16) NOT NULL default '',
lv varchar(16) NOT NULL default '',
PRIMARY KEY (ideleve),
KEY nom (nom)
)";
$result = $liendb->query($sql);
if ($result = $liendb->query($sql)) {
    printf("Table eleve créée avec succès.\n");
}

/*Fermeture de la connexion*/
$liendb->close();
?>

```

Même exemple avec Style orienté objet : PDO

```

<?php
//connexion et vérification
try
{
    $liendb = new PDO('mysql:host=localhost;dbname=test', 'root', '');
    print("Connecté avec succès... \n");
}
catch (Exception $e)
{
    //En cas d'erreur, on affiche un message et on arrête tout
    die('Erreur : ' . $e->getMessage());
}
/* "Create table eleve" */
$sql = "CREATE TABLE eleve (
ideleve int(10) unsigned NOT NULL auto_increment,
nom varchar(64) NOT NULL default '',

```

```

prenom varchar(64) NOT NULL default '',
adresse varchar(128) NOT NULL default '',
ville varchar(64) NOT NULL default '',
cp varchar(8) NOT NULL default '',
pays varchar(32) NOT NULL default 'france',
sexe varchar(8) binary NOT NULL default '',
naissance date NOT NULL default '0000-00-00',
taille int(10) unsigned NOT NULL default '0',
email varchar(64) NOT NULL default '',
telephone varchar(16) NOT NULL default '',
klv varchar(16) NOT NULL default '',
PRIMARY KEY (ideleve),
KEY nom (nom)
)";
if ($result = $liendb->query($sql)) {
printf("Table eleve créée avec succès.\n");
}
/*Déconnexion*/
$liendb = NULL;//Pour fermer la connexion, on détruit l'objet:
print 'Déconnecté avec succès.';
?>

```

Remarque :

Pour faire une requête, voici la syntaxe générale :

```
$result = $liendb->query('Tapez votre requête SQL ici');
```

On récupère ce que la base de données nous a renvoyé dans un autre objet que l'on a appelé ici *\$result*.

CREATE TABLE, INSERT INTO

Exercice 1

1. Écrivez le script *creer_table_eleve.php*, qui va vous permettre de créer votre table **eleve** :

```

TABLE eleve (
ideleve int(10) unsigned NOT NULL auto_increment,
nom varchar(64) NOT NULL default '',
prenom varchar(64) NOT NULL default '',
adresse varchar(128) NOT NULL default '',
ville varchar(64) NOT NULL default 'Essaouira',
cp varchar(8) NOT NULL default '',
pays varchar(32) NOT NULL default 'Maroc',
sexe varchar(8) binary NOT NULL default '',
naissance date NOT NULL default '0000-00-00',
taille int(10) unsigned NOT NULL default '0',
email varchar(64) NOT NULL default '',
telephone varchar(16) NOT NULL default '',
lv varchar(16) NOT NULL default '',

```

```
PRIMARY KEY (ideleve),  
KEY nom (nom))
```

2. Ecrire une requête SQL qui permet d'insérer l'élève suivant dans cette table :

Nom : Bidaoui.
Prénom : Anas.
Adresse : 12, rue Nouaceur.
Ville : Casablanca.
Code postal : 20640.
Pays : Maroc.
Sexe : masculin.
Date de naissance : 11/04/1992.
Taille : 170 cm.
Courriel : bidaoui.anas@gmail.com.
Téléphone : 0123456790.
Langue vivante : arabe.

Rappels :

```
INSERT INTO nom_table (liste de colonnes) VALUES (liste des valeurs des colonnes);
```

Les dates doivent être enregistrées à l'anglo-saxonne : année-mois-jour. Pour une date avec l'heure : AAAA-MM-JJ HH-MM-SS.

3. Écrire le script *enregistre_bidaoui.php* qui permet d'enregistrer cet élève dans la base de données :

```
<?php  
//connexion et vérification  
try  
{  
$liendb = new PDO('mysql:host=localhost;dbname=test', 'root', '');  
print("Connecté avec succès... \n");  
}  
catch (Exception $e)  
{//En cas d'erreur, on affiche un message et on arrête tout  
die('Erreur : ' . $e->getMessage());  
}  
/* "Create table eleve" */  
$sql='INSERT INTO eleve(nom, prenom, adresse, ville,cp, pays, sexe, naissance, taille,  
email, telephone, lv)  
VALUES (\'Bidaoui\', \'Anas\', \'12, rue Nouaceur\', \'Casablanca\',20640, \'Maroc\',  
\'masculin\', \'1992-04-11\', 170,  
\'bidaoui.anas@gmail.com\', 01234567890, \'Arabe\')';  
$liendb->exec($sql);  
//La méthode exec() exécute une requête et retourne le nombre d'enregistrements affectés.  
// $nombreDeLigne = $liendb->exec($sql); on insère et on en profite pour  
//récupérer le nombre de lignes insérées (ici 1)  
/*Déconnexion*/  
$liendb = NULL;  
print 'Déconnecté avec succès.';  
echo "eleve < Anas Bidaoui > enregistré";  
?>
```

Version mysqli_

```
<?php
/*connexion et vérification*/
$liendb = mysqli_connect('localhost', 'root', '', 'test');
print("Connecté avec succès... <br/>");
/* Vérification de la connexion */
if (mysqli_connect_errno()) {
    printf("Échec de la connexion : %s\n", mysqli_connect_error());
    exit();
}
/* "Create table eleve" */
$sql='INSERT INTO eleve(nom, prenom, adresse, ville,cp, pays, sexe, naissance, taille,
email, telephone, lv)
VALUES (\Rbati\, \Mohamed\, \14, rue Med V\, \Rabat\,10640, \Maroc\,
\masculin\, \1993-04-11\, 165,
\rbati.mohamed@uca.ma\, 01234567891, \Arabe\)\';
/* "Create table" ne retournera aucun jeu de résultats */
if (mysqli_query($liendb, $sql) === TRUE) {
    printf("Eleve ajouté avec succès.<br/>");
}

/*Déconnexion*/
mysqli_close($liendb);
print 'Déconnecté avec succès.<br/>';

echo "eleve < Mohamed Rbati > enregistré";
?>
?>
```

La commande SQL SELECT

Pour récupérer des données d'une table, on utilise la commande *SQL SELECT* :

SELECT liste des colonnes sélectionnées FROM nom de la table WHERE clause

Exercice 2

1. Écrire le script *voir_bidaoui.php* qui permet de récupérer les données présentes dans les colonnes nom et prenom de cette table.

```
<?php
//connexion et vérification
try{
    $liendb = new PDO('mysql:host=localhost;dbname=test', 'root', '');
    print("Connecté avec succès... \n");
}
catch (Exception $e)
{
    //En cas d'erreur, on affiche un message et on arrête tout
    die('Erreur : ' . $e->getMessage());
}
```

```

}
// Si tout va bien, on peut continuer
// On récupère les deux colonnes nom et prenom de la table eleve
/* commande SELECT*/
$sql='SELECT nom, prenom FROM eleve';
$result = $liendb->query($sql);
//$dta = $result->fetch(); //contient la première entrée
while ($dta = $result->fetch())
//On affiche chaque entrée une à une
{
?>
<p>
<strong>Nom: </strong><?php echo $dta['nom']; ?><br />
<strong>Prénom: </strong><?php echo $dta['prenom'];?>,
</p>
<?php
}
/*Déconnexion*/
$result->closeCursor(); // Termine le traitement de la requête
print 'Déconnecté avec succès.<br/>';
?>

```

Version mysqli_ : Style procédural

```

<?php
$liendb = mysqli_connect("localhost", "root", "", "test") or
die("Error " . mysqli_error($liendb));
$sql = "SELECT ideleve, nom, prenom FROM eleve";
$resultat = mysqli_query ($liendb,$sql);

while ($eleve = mysqli_fetch_array ($resultat)){
$id = $eleve['ideleve'];
$nom = $eleve['nom'];
$prenom = $eleve['prenom'];
echo "eleve [$id], nom = $nom, prenom = $prenom<br>";
}
mysqli_close($liendb);
?>

```

Version mysqli_ : Style orienté objet

```

<?php
$liendb = new mysqli("localhost", "root", "", "test")
or die("Error " . mysqli_error($liendb));
/* Vérification de la connexion */
if ($liendb->connect_errno) {
    printf("Échec de la connexion : %s\n", $liendb->connect_error);
    exit();
}
$sql = "SELECT ideleve, nom, prenom FROM eleve";
$resultat = $liendb->query($sql);

```



```

while ($eleve = $resultat->fetch_array(MYSQLI_ASSOC)){
    $id = $eleve['ideleve'];
    $nom = $eleve['nom'];
    $prenom = $eleve['prenom'];
    echo "eleve [$id], nom = $nom, prenom = $prenom<br>";
}
/* Fermeture de la connexion */
$liendb->close();
?>

```

2. Remplir la table par les trois nouveaux élèves suivants :

Premier élève :

Nom : Souiri.
 Prénom : Ahmed.
 Adresse : 15, rue Yassamine.
 Ville : Essaouira.
 Code postal : 2440.
 Pays : Maroc.
 Sexe : masculin.
 Date de naissance : 11/05/1993.
 Taille : 164 cm.
 Courriel : souiri.ahmed@gmail.com.
 Téléphone : 0123456792.
 Langue vivante : arabe.

Deuxième élève :

Nom : Mesfioui.
 Prénom : Roumaissae.
 Adresse : 1, rue 12.
 Ville : Safi.
 Code postal : 21640.
 Pays : Maroc.
 Sexe : féminin.
 Date de naissance : 12/04/1993.
 Taille : 170 cm.
 Courriel : mesfioui.roumaissae@yahoo.com.
 Téléphone : 0123456794.
 Langue vivante : espagnole.

Troisième élève :

Nom : Jdidi.
 Prénom : Khawla.
 Adresse : 2, boulevard Hassan II.
 Ville : El Jadida.
 Code postal : 22640.
 Pays : Maroc.
 Sexe : féminin.
 Date de naissance : 10/06/1992.

Taille : 166 cm.
Courriel : jdidi.khawla@outlook.com.
Téléphone : 0123456795.
Langue vivante : anglais.

3. Ajouter d'autres élèves dans la table `eleve`.

Exercice 3

Comptez le nombre d'enregistrements dans la table `eleve`. La fonction permettant d'obtenir le nombre d'enregistrements est `$num_rows` (ou `mysqli_num_rows()` pour le style procédural). Elle prend en argument le pointeur sur les données renvoyées par *MySQL* : `$resultat`.

Exercice 4

1. Ecrire une requête pour lister les élèves de taille supérieure ou égale à 150 cm.
2. Ecrire une requête pour lister les garçons d'Essaouira.
3. Ecrire une requête pour sélectionner les élèves étudiant l'espagnol ou l'anglais
4. Lister les élèves nés après 1989 et n'habitant pas à Rabat.
5. Écrire maintenant le script complet qui permet d'afficher les noms et les prénoms des garçons marakchis, en utilisant la boucle *while*.
6. Écrire maintenant le script complet qui permet d'afficher les noms et les prénoms des garçons marakchis, en utilisant la boucle *for*.
Indication : `mysqli_num_rows()/num_rows, mysqli_fetch_array()/fetch_array()`.
7. Écrire maintenant le script complet qui permet d'afficher les noms et les prénoms des garçons marakchis, en utilisant la boucle *foreach*.
8. Ecrire une requête pour compter le nombre de champs.
Indication : utiliser la fonction `mysql_num_fields()/field_count`
9. Ecrire une requête qui permet de compter le nombre de personnes.

Enregistrement d'une fiche

Rappel :

1) `empty` - Détermine si une variable est vide.

Syntaxe:

```
bool empty ( mixed $var )
```

`var`: Variable à vérifier.

Valeurs de retour:

Retourne `FALSE` si `var` existe et est non-vide, et dont la valeur n'est pas zéro.

Description:

Détermine si une variable est considérée comme vide. Une variable est considérée comme vide si elle n'existe pas, ou si sa valeur équivaut à `FALSE`.

La fonction `empty()` ne génère pas d'alerte si la variable n'existe pas.

Ce qui suit est considéré comme étant vide :

- `""` (une chaîne vide)
- `0` (0 en tant qu'entier)
- `0.0` (0 en tant que nombre à virgule flottante)

- "0" (0 en tant que chaîne de caractères)
 - NULL
 - FALSE
 - array() (un tableau vide)
 - \$var; (une variable déclarée, mais sans valeur)
- 2) isset - Détermine si une variable est définie et est différente de NULL

Réalisez maintenant le script *eleve_enregistre.php* qui va permettre d'enregistrer un élève dans la base.

Commencez par créer le formulaire *eleve.html* :

Fichier *eleve.php*

```
<html>
<head><title>Ajouter un élève</title></head>
<body>
<h1>Ajouter un élève :</h1>
<form action="eleve_enregistre.php" method="post">
<label>nom</label>
<input type="text" name="nom" /><br/>
<label>prénom</label>
<input type="text" name="prenom" /><br/>
<label>adresse</label>
<textarea name="adresse"></textarea><br/>
<label>ville</label>
<input type="text" name="ville" /><br/>
<label>code postal</label>
<input type="text" name="codepostal" /><br/>
<label>pays</label>
<input type="text" name="pays" /><br/>
<label>sexe</label>
<input type="radio" name="sexe" value="masculin" /> M -
<input type="radio" name="sexe" value="feminin" /> F<br/>
<label>date naissance</label>
<input type="text" name="naissance" /><br/>
<label>taille (cm)</label>
<input type="text" name="taille" /><br/>
<label>email</label>
<input type="text" name="email" /><br/>
<label>téléphone</label>
<input type="text" name="telephone" /><br/>
<label>langue vivante</label>
<select name="lv">
<option value="arabe">arabe</option>
<option value="anglais">anglais</option>
<option value="espagnol">espagnol</option>
<option value="allemand">allemand</option>
</select>
<br/>
```

```
<br/>
<input type="submit" value="enregistrer" />
</form>
</body>
</html>
```

Script eleve_enregistre.php

```
<?php
if (empty($_REQUEST['nom']) || empty($_REQUEST['prenom']) ||
empty($_REQUEST['adresse']) || empty($_REQUEST['ville']) ||
empty($_REQUEST['codepostal']) || empty($_REQUEST['pays']) ||
empty($_REQUEST['naissance']) || empty($_REQUEST['telephone']) ||
empty($_REQUEST['lv']))
die("ERREUR : tous les champs doivent être remplis.");
if ($_REQUEST['sexe']!="masculin" && $_REQUEST['sexe']!="feminin")
die("ERREUR : choisissez votre sexe.");
if (preg_match("/^[\\w\\.-]+@[\\w\\.-]+\\. [a-z]{2,3}$/i",
$_REQUEST['email']) == false)
die("ERREUR : adresse e-mail non valide.");
if ($_REQUEST['taille']<=100 || $_REQUEST['taille']>=220)
die("ERREUR : la taille n'est pas valide.");
$liendb = mysqli_connect("localhost", "root", "", "test");
$sql = "INSERT INTO eleve (nom, prenom, adresse, ville, cp, pays, sexe, naissance, taille,
email, telephone, lv)
VALUES ('".$_REQUEST['nom']."' , '".$_REQUEST['prenom']."' , '"
".$_REQUEST['adresse']."' , '".$_REQUEST['ville']."' , '"
".$_REQUEST['codepostal']."' , '".$_REQUEST['pays']."' , '"
".$_REQUEST['sexe']."' , '".$_REQUEST['naissance']."' , '"
".$_REQUEST['taille']."' , '".$_REQUEST['email']."' , '"
".$_REQUEST['telephone']."' , '"
".$_REQUEST['lv']."' )";
if (mysqli_query($liendb,$sql)!=false)
echo("eleve < '".$_REQUEST['nom']."' > enregistré.");
else
print("Echec lors de la création de la fiche.");
mysqli_close($liendb);
?>
```

Modifiez votre script de manière à ce qu'il affiche aussi le numéro de l'*ideleve* dans la réponse :

```
if (mysqli_query($liendb,$sql)!=false) {
$ideleve = mysqli_insert_id($liendb);
//Retourne l'identifiant automatiquement généré par la dernière requête
print("eleve [$ideleve] < '".$_REQUEST['nom']."' > enregistré.");
}
else
echo("Echec lors de la création de la fiche.");
```

En analysant bien ce script et en connaissant le problème de l'échappement de caractère, vous pourriez vous demander si ce que vous écrivez est valide. En effet, si l'internaute a tapé "34 rue

de l'Isle" comme adresse dans le formulaire, la requête reste-t-elle correcte malgré la présence des guillemets ? Est-il nécessaire d'échapper, avec une barre oblique inversée, toutes les variables provenant du formulaire ?

Exemple de requête entièrement échappée

```
$sql = "INSERT INTO eleve (nom, prenom, adresse, ville,
cp, pays, sexe, naissance, taille, email, telephone, lv)
VALUES ('" . addslashes($_REQUEST['nom']) ."', '" .
addslashes($_REQUEST['prenom']) ."', '" .
addslashes($_REQUEST['adresse']) ."', '" .
addslashes($_REQUEST['ville']) ."', '" .
addslashes($_REQUEST['codepostal']) ."', '" .
addslashes($_REQUEST['pays']) ."', '" .
addslashes($_REQUEST['sexe']) ."', '" .
addslashes($_REQUEST['naissance']) ."', '" .
addslashes($_REQUEST['taille']) ."', '" .
addslashes($_REQUEST['email']) ."', '" .
addslashes($_REQUEST['telephone']) ."', '" .
addslashes($_REQUEST['lv']) ."'");
```

Par défaut, en PHP, toutes les variables d'un *script* provenant d'un formulaire ou d'un cookie sont déjà échappées : on parle de *magic quotes* à propos de cet échappement automatique.

Ainsi, lorsque l'internaute envoie "34 rue de l'Isle", le *script* reçoit une variable `$_REQUEST['adresse']` contenant "34 rue de l\'Isle". Il est donc indispensable de savoir si votre système fonctionne sur ce modèle car si vous échappez une telle variable, son contenu est alors doublement échappé et contient "34 rue de l\\\'Isle". La fonction *PHP* qui vous permet de statuer sur la question est `get_magic_quotes_gpc()` :

```
if (get_magic_quotes_gpc())
echo "le système utilise les 'magic quotes'";
else
echo "le système n'utilise pas le système des 'magic quotes'";
```

Si ce script affiche "le système utilise les 'magic quotes'", votre requête d'origine est valide ; sinon, c'est la deuxième solution (avec les `addslashes()`) qui doit être utilisée.

Modification de la structure d'une table

Exercice 5

1. Ajoutez une colonne `essai` à la table `eleve` :
`ALTER TABLE eleve ADD essai INT(10) unsigned NOT NULL`
2. La colonne `essai` est donc un entier positif non nul. Faites en sorte maintenant qu'elle devienne un nombre flottant :
`ALTER TABLE eleve CHANGE essai essai FLOAT`
3. Modifiez maintenant son nom, et appelez-la `essai2` :
`ALTER TABLE eleve CHANGE essai essai2 FLOAT`
4. Finalement, supprimez cette colonne de la table `eleve` :
`ALTER TABLE eleve DROP essai2`

Envoi de fichier

Vous allez maintenant enrichir votre fiche et ajouter une photo au profil des élèves. Pour se faire, une colonne photo doit donc être insérée dans la table. C'est la commande *SQL ALTER* qui va permettre de modifier la structure de la table :

```
ALTER TABLE eleve ADD photo VARCHAR(64) NULL
```

Envoi de fichier

L'envoi d'un fichier impose l'utilisation de la méthode *POST* et d'un encodage spécial des données transmises. Le changement d'encodage est rendu possible grâce à l'attribut *enctype* qui prend alors la valeur "*multipart/form-data*".

Il vous est possible de choisir la photo sur votre disque dur en utilisant un *widget* spécifique pour le champ photo : l'*INPUT* de type *FILE*.

Votre formulaire devient donc :

form.html

```
<html>
<head><title>Ajouter un élève</title></head>
<body>
<h1>Ajouter un élève :</h1>
<form action="eleve_enregistre.php" method="post" enctype="multipart/form-data">
<label>nom</label>
<input type="text" name="nom" /><br/>
<label>prénom</label>
<input type="text" name="prenom" /><br/>
<label>adresse</label>
<textarea name="adresse"></textarea><br/>
<label>ville</label>
<input type="text" name="ville" /><br/>
<label>code postal</label>
<input type="text" name="codepostal" /><br/>
<label>pays</label>
<input type="text" name="pays" /><br/>
<label>sexe</label>
<input type="radio" name="sexe" value="masculin" /> M -
<input type="radio" name="sexe" value="feminin" /> F<br/>
<label>date naissance</label>
<input type="text" name="naissance" /><br/>
<label>taille (cm)</label>
<input type="text" name="taille" /><br/>
<label>email</label>
<input type="text" name="email" /><br/>
<label>téléphone</label>
<input type="text" name="telephone" /><br/>
<label>langue vivante</label>
<select name="lv">
<option value="arabe">arabe</option>
<option value="anglais">anglais</option>
```

```

<option value="espagnol">espagnol</option>
<option value="allemand">allemand</option>
</select><br/>
<label>photo</label>
<input type="file" name="photo" /><br/>
<br/>
<input type="submit" value="enregistrer" />
</form>
</body>
</html>

```

PHP rend l'envoi de fichier très facile. Deux fonctions sont utilisées :

`is_uploaded_file()` et `move_uploaded_file()`.

La fonction *is_uploaded_file()* permet de vérifier que l'envoi du fichier s'est déroulé sans encombre. Elle prend en paramètre un nom temporaire de fichier et retourne un booléen en cas de succès (true) ou d'échec (false). Le nom temporaire se récupère via la super-globale `$_FILES`. Cette variable est un tableau dont les clés correspondent aux noms des *widgets* de type file. Dans le cas présent, `$_FILES['photo']` est créé et propose deux éléments :

- `$_FILES['photo']['tmp_name']` qui contient le nom temporaire du fichier sur le serveur ;
- `$_FILES['photo']['name']` qui contient le nom du fichier que vous avez sélectionné sur votre machine.

La fonction s'utilise donc de la manière suivante :

```

if (is_uploaded_file($_FILES['photo']['tmp_name']))==true)
// on passe à la suite

```

Le déplacement du fichier temporaire est réalisé par la fonction *move_uploaded_file()*. Son premier paramètre correspond au nom du fichier temporaire (`$_FILES['photo']['tmp_name']`) et le second à l'emplacement final du fichier. Cet emplacement contient à la fois le répertoire qui contiendra le fichier et son nom. Généralement, le nom du fichier est celui d'origine : `$_FILES['photo']['name']`. Le répertoire en revanche peut se situer n'importe où. Si vous voulez déplacer le fichier dans un répertoire *images* se trouvant au même niveau que votre script, utilisez le chemin :

```
"images/".$_FILES['photo']['name']
```

Si le répertoire images se situe un niveau au-dessus de votre script, le chemin devient :

```
"../images/".$_FILES['photo']['name']
```

Contentez-vous pour l'instant de placer le fichier au même niveau que le *script eleve_enregistre.php* :

```

<?php
if (is_uploaded_file($_FILES['photo']['tmp_name'])) {
echo "File ". $_FILES['photo']['name'] ." téléchargé avec succès.\n";
move_uploaded_file($_FILES['photo']['tmp_name'],$_FILES['photo']['name']);
echo("<center><img src='".$_FILES['photo']['name']."' /></center></hr>");
}
else {
echo "Attaque possible par téléchargement de fichier : ";

```

```

echo "Nom du fichier : '". $_FILES['photo']['tmp_name'] . "'.";
die("Problème d'envoi du fichier.");
}
if (empty($_REQUEST['nom']) || empty($_REQUEST['prenom']) ||
empty($_REQUEST['adresse']) || empty($_REQUEST['ville']) ||
empty($_REQUEST['codepostal']) || empty($_REQUEST['pays']) ||
empty($_REQUEST['naissance']) || empty($_REQUEST['telephone']) ||
empty($_REQUEST['lv']))
die("ERREUR : tous les champs doivent être remplis.");
if ($_REQUEST['sexe']!="masculin" && $_REQUEST['sexe']!="feminin")
die("ERREUR : choisissez votre sexe.");
if (preg_match("/^[\\w\\.-]+@[\\w\\.-]+\\. [a-z]{2,3}$/i",
$_REQUEST['email']) == false)
die("ERREUR : adresse e-mail non valide.");
if ($_REQUEST['taille']<=100 || $_REQUEST['taille']>=220)
die("ERREUR : la taille n'est pas valide.");
$liendb = mysqli_connect("localhost", "root", "", "test");
$sql = "INSERT INTO eleve (nom, prenom, adresse, ville, cp, pays, sexe, naissance, taille,
email, telephone, lv)
VALUES ('".$_REQUEST['nom']. "', '".$_REQUEST['prenom']. "', '$_REQUEST['adresse']. "', '$_REQUEST['ville']. "', '$_REQUEST['codepostal']. "', '$_REQUEST['pays']. "', '$_REQUEST['sexe']. "', '$_REQUEST['naissance']. "', '$_REQUEST['taille']. "', '$_REQUEST['email']. "', '$_REQUEST['telephone']. "', '$_REQUEST['lv']. "')";
/*if (mysqli_query($liendb,$sql)!=false)
echo("eleve < '".$_REQUEST['nom']. "' > enregistré.");
else
print("Echec lors de la création de la fiche.");
*/
if (mysqli_query($liendb,$sql)!=false) {
$ideleve = mysqli_insert_id($liendb);
//Retourne l'identifiant automatiquement généré par la dernière requête
print("eleve [$ideleve] < '".$_REQUEST['nom']. "' > enregistré.");
}
else
echo("Echec lors de la création de la fiche.");
mysqli_close($liendb);
?>

```