



المدرسة العليا للتكنولوجيا الناصور
École Supérieure de Technologie de Nador
ⵜⴰⵎⴰⵔⵜ ⵜⴰⵏⵓⵔⴰⵢⵜ ⵜⴰⵖⵔⴰⵎⵜ ⵜⴰⵙⵉⵎⵓⵔⵜ ⵜⴰⵏⵓⵔⴰⵢⵜ

Développement web avancé

Filière : Ingénierie Logicielle et Cybersécurité (ILCS)

Projet PHP-Mysql



Exercice 0 : Mise en place

Partie 1 : Création de la BD

- 1) Créer une base de données « **upb_bd_td techno_web** ». Vous pouvez utiliser phpmyadmin.
- 2) En utilisant le script SQL ci-dessous, Créer la table « **etudiants** ».

```
DROP TABLE IF EXISTS `etudiants`;  
CREATE TABLE IF NOT EXISTS `etudiants` (  
  `id` int(11) NOT NULL AUTO_INCREMENT,  
  `nom` varchar(45) NOT NULL,  
  `prenoms` varchar(45) NOT NULL,  
  `sexe` enum('M','F') DEFAULT NULL,  
  `email` varchar(45) NOT NULL,  
  `password` varchar(45) DEFAULT NULL,  
  `contact` varchar(20) DEFAULT NULL,  
  `quartier` varchar(45) DEFAULT NULL,  
  `presentation` longtext,  
  `photo` varchar(100) DEFAULT NULL,  
  `id_filiere` int(11) DEFAULT NULL,  
  `date_ajout` datetime NOT NULL DEFAULT CURRENT_TIMESTAMP,  
  `date_modif` datetime DEFAULT NULL,  
  PRIMARY KEY (`id`),  
  UNIQUE KEY `email` (`email`),  
  KEY `id_filiere` (`id_filiere`)  
) ENGINE=MyISAM AUTO_INCREMENT=8 DEFAULT CHARSET=utf8;
```

- 3) En utilisant le script SQL ci-dessous, Créer la table « **filiere** ».

```
DROP TABLE IF EXISTS `filiere`;  
CREATE TABLE IF NOT EXISTS `filiere` (  
  `id_filiere` int(11) NOT NULL AUTO_INCREMENT,  
  `nom_filiere` varchar(100) NOT NULL,  
  `description_filiere` varchar(255) DEFAULT NULL,  
  PRIMARY KEY (`id_filiere`)  
) ENGINE=MyISAM AUTO_INCREMENT=3 DEFAULT CHARSET=utf8;
```

- 4) Renseigner la table « **etudiants** » avec des informations fictives/réelles. Vous pouvez utiliser le script ci dessous :



المدرسة العليا للتكنولوجيا الناصور
École Supérieure de Technologie de Nador
ⵜⴰⵎⴰⵔⵜ ⵜⴰⵏⵓⵔⵜ ⵜⴰⵖⵓⵔⵜ ⵜⴰⵙⵉⵎⵓⵔⵜ ⵜⴰⵏⵓⵔⵜ

```
INSERT INTO `etudiants` (`id`, `nom`, `prenoms`, `sexe`, `email`,  
`password`, `contact`, `quartier`,  
`presentation`, `id_filiere`, `date_ajout`, `date_modif`) VALUES  
(1, 'ACKA', 'KOUAKOU JEAN YVES', 'M', 'acka.kjy@upb.ci', '2022L2',  
NULL, 'COCODY', 'TEST TEST TEST TEST  
TEST TEST TEST KOUAKOU JEAN YVES', 1, '2022-02-16 20:39:56', NULL),  
(2, 'AGBEDJE', 'OSSOHO ABACUC', 'M', 'agbedje.oab@upb.ci', '2022L2',  
NULL, 'YOPOUGON', 'TEST TEST  
TEST TEST TEST TEST TEST', 1, '2022-02-16 20:39:56', NULL),  
(3, 'AGNIN', 'JOEL JEAN DAVID', 'M', 'agni.jd@upb.ci', '2022L2',  
NULL, 'PORT-BOUET', 'TEST TEST TEST TEST  
TEST TEST TEST', 1, '2022-02-16 20:42:39', NULL),  
(4, 'KOUAKOU', 'ANNAYAH AKISSI', 'F', 'kouakou.annayah@upb.ci',  
'2022L2', NULL, 'ABOBO', 'TEST TEST TEST  
TEST TEST TEST TEST', 1, '2022-02-16 20:46:18', NULL),  
(5, 'COULIBALY', 'ROKIA', 'F', 'coulibaly.rokia@upb.ci', '2022L2',  
NULL, 'ANYAMA', 'TEST TEST TEST TEST  
TEST TEST TEST', 1, '2022-02-16 20:46:18', NULL),  
(6, 'SAKO', 'KADER', 'M', 'sako.kader@upb.ci', '2022L2', NULL,  
'KOUMASSI', 'TEST TEST TEST TEST TEST TEST  
TEST', 1, '2022-02-16 20:49:14', NULL),  
(7, 'TESSOUGUE', 'AWA', 'F', 'tessougue.awa@upb.ci', '2022L2', NULL,  
'BINGERVILLE', 'TEST TEST TEST TEST  
TEST TEST TEST', 1, '2022-02-16 20:49:14', NULL);
```

5) Renseigner la table « **filières** » avec des informations fictives/réelles. Vous pouvez utiliser le script ci dessous :

```
INSERT INTO `filières` (`id_filiere`, `nom_filiere`, `description_filiere`) VALUES  
(1, 'MIAGE', 'Méthodes Informatiques Appliquées à la Gestion des Entreprises'),  
(2, 'SEA', NULL);
```

```
INSERT INTO `filières` (`id_filiere`, `nom_filiere`,  
`description_filiere`) VALUES  
(1, 'MIAGE', 'Méthodes Informatiques Appliquées à la Gestion des  
Entreprises'),  
(2, 'SEA', NULL);
```

Partie 2: Connexion à la BD

PDO (PHP Data Object) est une bibliothèque de fonctions PHP permettant d'accéder à n'importe quelle base de données.

Dans ce projet, pour se connecter à la base de données, **nous utiliserons PDO**.

Syntaxe : Exemple de script PHP permettant d'afficher le contenu de la table « **etudiants** ».

1) Pour se connecter à une base de données avec PDO, on utilise la syntaxe suivante :

```
$db = new PDO('mysql:host=localhost;dbname=nom-base-de-données;charset=utf8', 'user',  
'Motde-passe');
```

NB :

User : utilisateur de la base de données, ici dans notre cas on utilisera l'utilisateur **root**

2) Utilisation des blocs try et catch pour tester la présence d'erreurs :

En cas d'erreur, PDO renvoie ce qu'on appelle une **exception**, qui permet de « capturer » l'erreur.

```
<?php  
try  
{  
    /* $db = new PDO('mysql:host=localhost;dbname=nom-base-de-données;charset=utf8', 'user', 'M  
    $db = new PDO('mysql:host=localhost;dbname=upb_bd_td techno_web;charset=utf8', 'root', '');  
}  
catch (Exception $e)  
{  
    die('Erreur : ' . $e->getMessage());  
}
```

3) Faire des requêtes avec l'objet PDO :

Pour effectuer une requête à l'aide de l'objet PDO, on utilise premièrement la méthode **prepare()**

Exemple :

```
$etudiantsStatement = $db->prepare('SELECT * FROM etudiants');
```

Ensuite on utilise la méthode **execute()** sur le résultat de la méthode **prepare()**

Dans le cas d'une requêtes SELECT, on utilise ensuite la méthode **fetch()** ou **fetchAll()** pour récupérer les données sous forme de tableau.

4) Pratique : Créer, tester et expliquer le script « prepa_tp3.php » ci-dessous :

```
<?php  
try  
{  
    /* $db = new PDO('mysql:host=localhost;dbname=nom-base-de-données;charset=utf8', 'user', 'Mot-de  
    passe'); */  
    $db = new PDO('mysql:host=localhost;dbname=upb_bd_td techno_web;charset=utf8', 'root', '');  
}  
catch (Exception $e)  
{
```



المدرسة العليا للتكنولوجيا الناصور
École Supérieure de Technologie de Nador
ⵜⴰⵎⴰⵔⵜ ⵜⴰⵎⴰⵔⵜ ⵜⴰⵖⴰⵏⵜ ⵜⴰⵙⴰⵎⴰⵏⵜ ⵜⴰⵖⴰⵏⵜ

```
die('Erreur : ' . $e->getMessage());
}
// On récupère tout le contenu de la table etudiants
$query = 'SELECT * FROM etudiants';
$stmt = $db->prepare($query);
$stmt->execute();
$etudiants = $stmt->fetchAll();
// On affiche nom et prenom chaque étudiant
foreach ($etudiants as $etudiant) {
    ?>
    <p><?php echo $etudiant['nom']. ' '. $etudiant['prenoms']; ?></p>
    <?php
}
?>
```

5) Pratique : Exemple de script d'insertion

```
// Ecriture de la requête
$query_2 = 'INSERT INTO filieres(nom_filiere, description_filiere) VALUES
(:nom_filiere, :description_filiere)';
// Préparation
$insertFiliere = $db->prepare($query_2);
// Exécution ! La filière est maintenant en base de données
$insertFiliere->execute([
    'nom_filiere' => 'MP',
    'description_filiere' => 'Maths Physique',
]);
```

6) Pratique : Exemple de script de mise à jour

```
// Ecriture de la requête
$query_2 = 'UPDATE filieres SET nom_filiere = : nom_filiere, description_filiere = : description_filiere,
WHERE id =
:id';
// Préparation
$insertEtudiant = $db->prepare($query_2);
// Exécution
$insertEtudiant->execute([
    'nom_filiere' => 'MPSI',
    'prenoms' => 'Maths Physique Science Industrielle',
    'id' => 3,
]);
```

7) Pratique : Exemple de script de suppression

```
$query_3 = 'DELETE FROM etudiants WHERE id=:id';
$stmt = $db->prepare($query_3);
$stmt->execute(['id' => 3]);
```

Exercice 1 : PHP – Connexion à une Base de Données

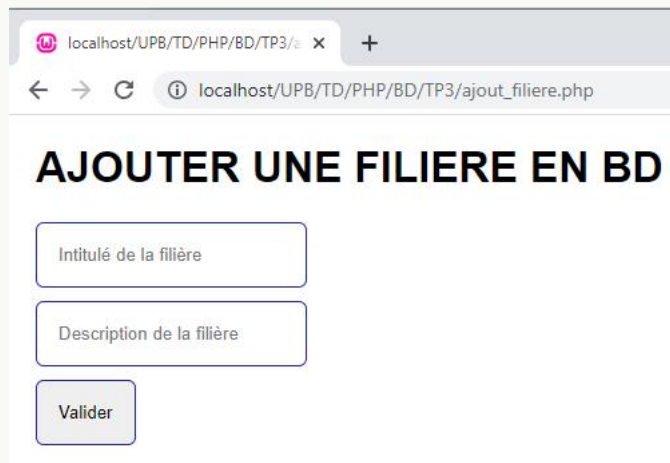
1) Créer la page « **ajout_filiere.php** » contenant un formulaire permettant d'ajouter une nouvelle filière dans la base de données.

- Vérifier que les champs ne sont pas vides :

Exemple :

```
if(isset($_POST["valider"]) AND !empty($_POST["nom_filiere"] AND $_POST["description_filiere"]))
```

- Vérifier que la filière n'est pas déjà présente dans la base.



localhost/UPB/TD/PHP/BD/TP3/ x +

localhost/UPB/TD/PHP/BD/TP3/ajout_filiere.php

AJOUTER UNE FILIERE EN BD

Intitulé de la filière

Description de la filière

Valider

2) Créer la page « **ajout_etudiant.php** » permettant d'ajouter un étudiant dans la base de données

- N'ajouter pas le mot de passe en clair dans la base de données ! Encrypter le en md5.

Exemple :

```
$password = md5($_POST["password"]);
```

- Vérifier que les champs ne sont pas vides
- Vérifier que l'utilisateur n'est pas déjà présent dans la base.

Indication :

On pourra faire une requête pour vérifier si l'email renseigné n'est pas déjà dans la base de données.

- Empêcher l'utilisateur d'injecter du code dans la base de données.
- Si l'insertion fonctionne, afficher une notification de succès. Sinon afficher une notification d'erreur



المدرسة العليا للتكنولوجيا الناصور
École Supérieure de Technologie de Nador
ⵜⴰⵎⴰⵔⵜ ⵜⴰⵏⵓⵙⴰⵢⵜ ⵜⴰⵖⵔⴰⵏⵜ ⵜⴰⵙⴰⵎⴰⵏⵜ

AJOUTER UN ETUDIANT EN BD

[[Consulter la liste des étudiants](#)]

3) Créer une page « **liste_etudiant.php** » qui permet d'afficher la liste des étudiants dans un tableau (Afficher tous les étudiants enregistrés dans la table « **etudiants** »).

- Ajouter un lien vers la page **ajout_etudiant.php** au-dessus du tableau
- Prévoir les liens ou boutons permettant d'accéder aux pages :

details_etudiant.php , **modifier_etudiant.php** et **supprimer_etudiant.php**

Indication :

`<a href="details_etudiant.php?id_etudiant=<?php echo $etudiant['id']; ?>">Voir plus `



المدرسة العليا للتكنولوجيا الناصور
École Supérieure de Technologie de Nador
ⵜⴰⵎⴰⵔⵜ ⵜⴰⵎⴰⵔⵜ ⵜⴰⵖⵔⴼⴰⵏⵜ ⵜⴰⵏⵓⵔⵜ ⵜⴰⵖⵔⴼⴰⵏⵜ

LISTE DES ETUDIANTS

[\[Ajouter un étudiant\]](#)

Id	Nom	Prenoms	Genre	Email	Quartier	Contact	Actions		
1	ACKA	KOUAKOU JEAN YVES	M	acka.kjy@upb.ci	COCODY		Voir plus	Modifier	Supprimer
3	AGNIN	JOEL JEAN DAVID	M	agni.jd@upb.ci	PORT-BOUET		Voir plus	Modifier	Supprimer
4	KOUAKOU	ANNAYAH AKISSI	F	kouakou.annayah@upb.ci	ABOBO		Voir plus	Modifier	Supprimer
5	COULIBALY	ROKIA	F	coulibaly.rokia@upb.ci	ANYAMA		Voir plus	Modifier	Supprimer
12	COULIBALYDD	Ssss	F	sssadmin@mffe.ci	TestQG	045555630011	Voir plus	Modifier	Supprimer
7	TESSOUGUE	AWA	F	tessougue.awa@upb.ci	BINGERVILLE		Voir plus	Modifier	Supprimer
8	GODE	Testdemo 22	F	egode.icci@gmail.com	TestQG	045555630011	Voir plus	Modifier	Supprimer


4) Créer une page **details_etudiant.php** qui affiche les informations d'un seul étudiant, cette page doit également afficher un formulaire d'ajout d'une image

- Ajouter un input type= 'file' dans le formulaire
- Sauvegarder la photo dans un dossier « uploads»
- Sauvegarder le nom du fichier image dans la BD

localhost/UPB/TD/PHP/BD/TP3/c x +

localhost/UPB/TD/PHP/BD/TP3/details_etudiant.php?id_etudiant=2

INFORMATION ETUDIANT



Enregistrement d'une image

Choisir un fichier

Aucun fichier choisi

Valider

Id	2
Nom	AGBEDJE
Prenoms	OSSOHOU ABACUC
Genre	M
Email	agbedje.oab@upb.ci
Quartier	YOPOUGON
Contact	
	Modifier informations étudiant

- 5) Créer une page **modifier_etudiant.php** qui permet d'afficher un formulaire pré rempli (avec les informations de l'étudiant qu'on souhaite modifier).
- Vérifier la présence de la variable `$_GET['id_etudiant']`, si cette variable n'existe pas alors rediriger sur la page « **liste_etudiant.php** »
 - Faire la mise à jour des informations dans la BD.
- 6) Créer la page **supprimer_etudiant.php** qui permet de supprimer les informations d'un étudiant en BD
- Vérifier la présence de la variable `$_GET['id_etudiant']`, si cette variable n'existe pas alors rediriger sur la page « **liste_etudiant.php** »
 - Supprimer les informations de l'étudiant correspondant à l'`id_etudiant` reçu.
- 7) Créer le script « **traitement_image.php** » permettant d'enregistrer la photo de l'étudiant dans un dossier « uploads » et sauvegarder le chemin du fichier image dans la base de données en mettant à jour de la colonne « photo » de l'étudiant concerné.
- Assurez-vous que le formulaire de la page **details_etudiant.php** permettant d'uploader la photo de l'étudiant possède l'attribut « `enctype="multipart/form-data"` »
 - Dans le script « **traitement_image.php**, vous pouvez :
 - Vérifier si le fichier image est une image réelle ou une fausse image en utilisant la méthode `getimagesize()` qui Retourne la taille d'une image
Ex : `getimagesize($_FILES["fileToUpload"]["tmp_name"])`
 - Vérifier si le fichier existe déjà grâce à la fonction `file_exists()` qui Vérifie si un fichier ou un dossier existe
 - Télécharger le fichier avec la methode `move_uploaded_file()` qui permet de Déplace un fichier téléchargé

Syntaxe :

`move_uploaded_file(string$from,string$to):bool`

Exercice 2 : PHP – MySQL—Bootstrap – POO

On souhaite refaire l'exercice précédent en utilisant le Framework Bootstrap, la Programmation Orientée Objet (POO) en PHP et un système d'authentification utilisant la Base de donnée.

NB : - La liste des étudiants, l'ajout, la modification, la suppression doivent être protégés par une authentification.

Faire un CRUD (Create Read Update Delete) sur la table étudiant.