

Conception de Sites Web Dynamiques : TD 6

Premier exercice avec une base de données MySQL

Objectif de ce TD :

Dans ce TD, nous allons relier vos pages web à une base de données. Comme exemple, vous créez un article sur un sujet de votre choix, et vous permettez de laisser aux internautes de laisser des commentaires via un formulaire. Vous affichez également les commentaires laissés par internautes. Afin de ne pas afficher des commentaires ayant du contenu illicite, nous allons passer par une étape de validation des commentaires.

Pour ce faire, il y aura plusieurs étapes :

- Découverte de votre interface phpMyAdmin, disponible sur les serveurs du CREMI, ou via MAMP/Wampserver/Xampp.
- Création d'une base de données et d'une table `commentaire` dans votre interface phpMyAdmin.
- Liaison de vos pages web à votre base de données via un fichier `connexion_base.php` que vous incluez en tout début de vos pages web.
- Exécution de requêtes MySQL dans vos pages web.
- Administration de votre base de données via votre interface phpMyAdmin.

A respecter tout au long de ce TD :

- Si vous ne travaillez pas sur les postes du CREMI, vérifiez que votre logiciel de serveur Apache (MAMP ou Wampserver64 ou Xampp) est bien démarré..
- Les extensions des fichiers doivent être toujours `.php`
- Dans le navigateur, l'URL commence toujours par `http://` et non pas par `file://`.

Option 1 : Votre interface d'administration de base de données php-MyAdmin avec les serveurs du CREMI

Si vous travaillez sur vos propres ordinateurs, passez à directement à l'option 2.

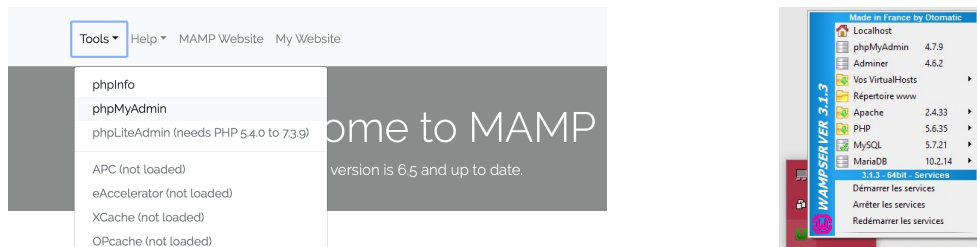
1. Il faut tout d'abord créer votre base de données MySQL au CREMI, et choisir un mot de passe. Vous pouvez faire cela sur le lien suivant : <https://services.emi.u-bordeaux.fr/mariadb/>
Choisissez un mot de passe que vous pouvez me montrer, car il apparaîtra en clair dans vos fichiers de configuration.

2. Ouvrez l'interface phpMyAdmin via le lien <https://services.emi.u-bordeaux.fr/mariadb/phpmyadmin/> et connectez-vous en mettant votre login ENT, ainsi que le mot de passe que vous avez choisi. Vous allez voir que votre base de données MySQL créé porte le nom de votre login ENT.

Passez maintenant directement au premier exercice, sans faire l'option 2.

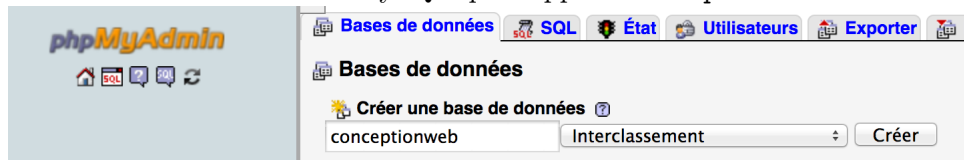
Option 2 : Votre interface d'administration de base de données php-MyAdmin avec MAMP/Wampserver

Ouvrez votre interface phpMyAdmin, disponible avec MAMP/Wampserver :



Si on vous demande un login, c'est **root**, et le mot de passe est également **root** si vous utilisez MAMP (il est vide pour Wampserver).

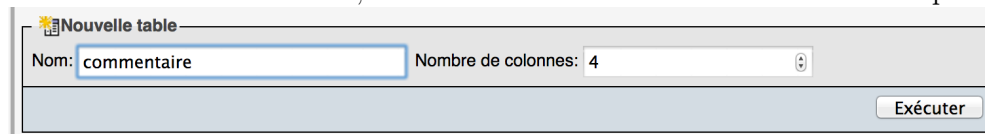
3. Créez une base de données MySQL qui s'appelle **conceptionweb**.



1 Création d'une table MySQL dans votre base de données et manipulation de données d'une table

Exercice 6.1 Création d'une table MySQL dans votre base de données

1. Sélectionnez votre base de données en cliquant sur son nom dans la barre latérale gauche (votre base de données s'appelle **conceptionweb** si vous travaillez avec MAMP, et si vous travaillez sur les serveurs du CREMI, elle porte le nom de votre login ENT).
2. Dans votre base de données, créez une nouvelle table **commentaire** avec quatre champs :



Les quatre champs sont les suivants :

- un champs INT qui s'appelle **id** avec un index de clé primaire (PRIMARY) ainsi que "auto-increment" (A_I),
- un champs TEXT qui s'appelle **nom**,
- un champs TEXT qui s'appelle **texte**,
- un champs DATETIME qui s'appelle **date**.

Afficher Structure SQL Rechercher Insérer Exporter Importer Privileges Opérations Suivi Déclencheurs									
Nom de la table: commentaire		Ajouter 1 colonne(s)		Exécuter					
Nom	Type	Taille/Valeurs	Défaut	Interclassement	Attributs	Null	Index	A.I	Comm
id	INT		Aucune			☐	PRIMARY	☒	
nom	TEXT		Aucune			☐	---	☐	
texte	TEXT		Aucune			☐	---	☐	
date	DATETIME		Aucune			☐	---	☐	

Exercice 6.2 Manipulation d'enregistrements dans une table d'une base de données MySQL

1. En cliquant sur "Insérer" dans votre interface, créez un nouvel enregistrement avec un nom, un texte, et une date de votre choix. Cliquez sur exécuter.

Colonne	Type	Fonction	Null	Valeur
id	int(11)			
nom	text			Paul
texte	text			Article très intéressant
date	datetime			2021-03-02 08:04:06

Regardez la requête MySQL résultante (commençant par "INSERT ...") dans le cadre ci-dessous qui ne tient pas compte des apostrophes de type accent-grave ` , ni du nom de la base de données. Notez l'importance des apostrophes ' qui délimitent les chaînes de caractères.

```
INSERT INTO commentaire (id, nom, texte, date) VALUES
(NULL, 'Paul', 'Article très intéressant', '2022-03-01 08:04:06');
```

2. Dans les liens bleus du haut de l'interface, cliquez sur "Afficher" (ou "Parcourir"). Notez la requête MySQL résultante (commençant par "SELECT ...") dans le cadre ci-dessous, sans mettre le nom de la base de données, ni les accents grave.

3. Effacer cet enregistrement (en affichant vos données, et en cliquant sur le petit x rouge à côté de votre enregistrement). Notez la requête MySQL (commençant par "DELETE ..."), sans mettre le nom de la base de données, ni les accents grave.

4. Créer de nouveau un enregistrement avec un nom, un texte, et une date de votre choix. Validez et affichez les enregistrements de votre table.
5. Modifier le texte de cet enregistrement selon votre goût (en affichant vos données, et en cliquant sur le petit crayon à côté de votre enregistrement). Notez la requête MySQL résultante (commençant par "UPDATE ...") dans le cadre ci-dessous (sans mettre le nom de la base de données, ni les accents grave).

Exercice 6.3 *Fichier connexion_base.php à inclure*

1. Créez un nouveau dossier `td6` dans votre dossier `cswd`.
2. Récupérez le code source du fichier `connexion_base.php` à partir de la page du cours <http://www.labri.fr/perso/preuter/cswd2022/>, collez-le dans un fichier avec votre éditeur de texte, et sauvegardez-le sous le nom `connexion_base.php` dans votre dossier `td6`. Ce fichier permet de se connecter à la base de données MySQL et de sélectionner votre base de données.

```
<?php
// définition des paramètres de connexion à la base de données
$config_base['hote']      = "localhost";
$config_base['utilisateur'] = "root";
$config_base['motdepasse'] = "root";    // si Wampserver, mettez "" ici
$config_base['nom_base']   = "conceptionweb";

if (substr($_SERVER['SERVER_NAME'],-17) == "emi.u-bordeaux.fr")
{
    $config_base['hote']      = "mariadb";
    $config_base['utilisateur'] = "loginENT";
    $config_base['motdepasse'] = "MOTDEPASSE";
    $config_base['nom_base']   = "loginENT";
}

// connexion à la base de données
try {
    $pdo = new PDO( "mysql:host={$config_base['hote']};
                    dbname={$config_base['nom_base']}",
                    "{$config_base['utilisateur']}",
                    "{$config_base['motdepasse']}");

    // afficher les messages d'erreurs pour trouver les erreurs
    $pdo->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_WARNING);

    // jeu de caractères : UTF-8
    $pdo->query("SET NAMES utf8");
    $pdo->query("SET CHARACTER SET utf8");
}
catch (PDOException $exception) {
    echo "Connexion échouée : " . $exception->getMessage();
}
?>
```

Remarque : Par la suite, tous les fichiers qui utilisent votre base de données, utiliseront le fichier `connexion_base.php` en l'incluant en tout début de page avec

```
<?php require_once("connexion_base.php"); ?>
```

Exercice 6.4 Page web d'un article avec commentaires

1. Créer une nouvelle page HTML5 dans le fichier `article.php` avec les balises comme d'habitude. Dans la partie `body`, ajoutez un titre `<h1>` de votre choix et créez les 3 paragraphes (avec la balise avec `<p>...</p>`) avec le texte issu de la version populaire de "Lorem ipsum" issue de la page <https://fr.wikipedia.org/wiki/Faux-texte>.
2. En tout début de votre fichier (donc même avant `DOCTYPE`), incluez la connexion à la base de données. Faites bien attention qu'il n'y ait ni d'espaces, ni de saut de ligne avant cette instruction !

```
<?php require_once("connexion_base.php"); ?>
```

3. Dans ce fichier `article.php`, après votre article (donc après les 3 paragraphes de lorem ipsum), affichez tous les enregistrement de votre table `commentaire`. Un exemple pour afficher tous les enregistrements (sans mise en forme) se trouve ci-dessous.

```
<?php
// exécuter une requete MySQL
$requete = "SELECT * FROM commentaire;";
$reponse = $pdo->prepare($requete);
$reponse->execute();

// récupérer tous les enregistrements dans un tableau
$enregistrements = $reponse->fetchAll();

// connaitre le nombre d'enregistrements
$nombreReponses = count($enregistrements);

// parcourir le tableau des enregistrements
for ($i=0; $i<count($enregistrements); $i++)
{
    echo $enregistrements[$i]['nom'];
    echo $enregistrements[$i]['texte'];
    echo $enregistrements[$i]['date'];
}
?>
```

4. Modifiez la mise en forme, p.ex. utilisez des tables HTML de 3 colonnes (nom, texte, date), ou, si vous voulez faire plus simple, une liste. Rappelez-vous, nous avons déjà fait cette partie de l'exercice dans les TDs précédents pour des tableaux associatifs ...

Exercice 6.5 Formulaires

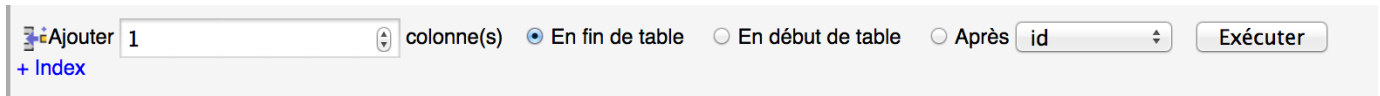
1. A la fin de votre page `article.php` créez un formulaire qui renvoie vers la page `ajouter.php`, et qui permet de renseigner un champs d'entrées du texte (appelé *nom*), d'une zone de texte (appelée *texte*) ainsi qu'un bouton qui envoie le formulaire (afin d'ajouter le commentaire dans la base de données).
2. Créer la page `ajouter.php` qui récupère et qui affiche les données du formulaire, ainsi que le texte dans un paragraphe `<p>` "Votre commentaire a bien été enregistré".
3. Cette page inclura le fichier `connexion_base.php`.
4. Ajoutez les données saisies dans votre base de données avec les instructions suivantes !

```
<?php
// exécuter une requete MySQL de type INSERT
$requete="INSERT INTO commentaire (nom, texte, date) VALUES (?, ?, NOW());";
$reponse=$pdo->prepare($requete);
$reponse->execute(array($nom, $texte));
?>
```

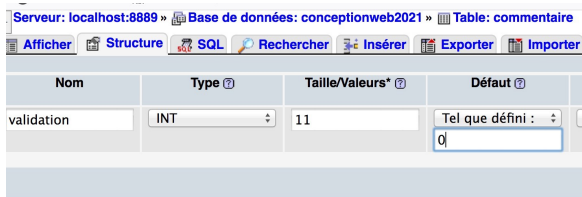
5. A la fin de votre page `ajouter.php`, créez un lien hypertexte avec le texte "retourner à l'article" qui permet de renvoyer à l'article `article.php` !

Exercice 6.6 Administration de commentaires

1. Dans votre table `commentaire`, ajouter en fin de table un champs INT qui s'appelle `validation`. Pour cela, sur phpMyAdmin, choisissez la table `commentaire` et cliquez sur "structure". Ensuite, ajouter une colonne en fin de table comme suit :



Comme valeur par défaut, spécifiez 0 :



2. Dans votre interface d'administration *phpMyAdmin*, sélectionnez les commentaires que vous validez en mettant le champs `validation` à 1.
3. Dans votre interface d'administration *phpMyAdmin*, dans les liens bleus du haut de l'interface, cliquez sur "Rechercher". Recherchez tous les commentaires ayant un champs `validation` de 1. Notez la requête MySQL résultante (commençant par "SELECT ...") dans le cadre ci-dessous.

4. Dans votre fichier `article.php`, modifiez votre requête que vous puissiez uniquement afficher les commentaires étant validés :

```
SELECT * FROM commentaire WHERE validation=1
```

2 Exercices optionnelles

Exercice 6.7 Fonctions PHP

1. Afin d'améliorer votre page `article.php`, étudiez les fonctions PHP `date()`, `mktime()`, et `substr()` sur le manuel de PHP (<http://www.php.net/manual/fr>). Essayez de trouver une meilleure mise en forme pour la date du commentaire postée, de type "Lundi, 5 Mars, à 10h00". Pour cela, soit vous utilisez les capacités de ces fonctions pleinement, mais vous pouvez aussi définir les deux tableaux suivants :

```
$jours = array("Dimanche", "Lundi", "Mardi", "Mercredi", "Jeudi",  
              "Vendredi", "Samedi");  
$mois = array("Janvier", "Février", "Mars", "Avril", "Mai", "Juin",  
              "Juillet", "Août", "Septembre", "Octobre", "Novembre",  
              "Décembre");
```

Aide : Dans un premier temps, découpez la chaîne de caractère de la date en année, mois, jour, heure, minute, secondes avec la fonction `substr()`. Ensuite, créez un timestamp avec la fonction `mktime()` à partir de ces informations. Finalement, vous pouvez récupérer les divers informations, comme le jour de la semaine, avec la fonction `date()`.