

TP3

Modèle de base de données matos4
Pages PHP pour ajouter un nouveau type de matériel
Programme C pour envoyer des courriels

Ce TP correspond à la mise en œuvre des notions SQL et PHP abordées en cours. Vous devez rendre votre travail en fin de séance.

Partie I : On utilise la base matos3 !

La première partie consiste à utiliser le script **matos3_ddl.sql** et le script **totale_dml.sql** associés pour remplir la base avec des enregistrements valides. Récupérer tous les fichiers nécessaires sur Moodle. Créer un utilisateur **matos3** et remplir la base avec les bons fichiers.

Partie II : Mise à jour de la raspberry PI !


Pour pouvoir compiler l'application de courriel, il va falloir mettre à jour la raspberry pi

```
$ sudo apt-get update  
$ sudo apt-get install libssl-dev  
$ sudo apt-get install graphviz
```

Partie III : Insertion d'un nouveau type de matériel dans la base !

Insérer un nouveau type de matériel dans la base **matos3** est une opération relativement complexe qui se fait en 4 étapes.

L'étape 1 consiste à demander à l'utilisateur le nom du type de matériel qu'il souhaite insérer. Avant de passer à l'étape 2, il faut vérifier que le nom est valide et absent de la table **materiel**.



The screenshot shows a web browser window with the title 'Page permettant d'ajouter un nouveau type de materiel dans la base - Mozilla Firefox'. The address bar shows 'localhost/ajouternouveau.php'. The page content is titled 'Page d'ajout d'un nouveau type de materiel 1/4'. Below the title, there is a note '* champs requis.' followed by a form field labeled 'Nom du nouveau type de materiel:' with an asterisk indicating it is required. A 'Valider' button is located below the input field.

L'étape 2 consiste à demander à l'utilisateur toutes les colonnes spécifiques au nouveau matériel, ainsi que leurs types associés.

Page permettant d'ajouter un nouveau type de materiel dans la base - Mozilla Firefox

Page permettant d'ajouter un nouveau type de materiel dans la base - Mozilla Firefox

localhost/ajouternouveau2.php

Page d'ajout d'un nouveau type de materiel 2/3

ici pingouin

La base de données s'apprête à créer une table **tm_pingouin**.

Cliquer sur "Ajouter une nouvelle colonne à la table" pour modifier la structure de **tm_pingouin**.

Cliquer sur "Valider la structure de la table" pour passer à l'étape suivante.

Nouvelle colonne

nom varchar

Nouvelle colonne

taille int

Nouvelle colonne

couleur varchar

L'étape 3 consiste à afficher la requête SQL qui va procéder à la création d'une nouvelle table **tm_X** où **X** représente le nom du nouveau type de matériel. Un bouton **Créer la table** permet de passer à l'étape 4.

Page permettant d'ajouter un nouveau type de materiel dans la base - Mozilla Firefox

localhost/ajouternouveau3.php

Page permettant d'ajouter un nouveau type de materiel dans la base 3/4

pingouin

0 nom VARCHAR(100)

1 taille INT

2 couleur VARCHAR(100)

```
CREATE TABLE IF NOT EXISTS `matos2`.`tm_pingouin` (
  `idtm_pingouin` INT NOT NULL AUTO INCREMENT,
  `instance_materiel_idinstance_materiel` INT NOT NULL,
  `nom` VARCHAR(100),
  `taille` INT,
  `couleur` VARCHAR(100),
  PRIMARY KEY (`idtm_pingouin`),
  UNIQUE INDEX `idtm_pingouin_UNIQUE` (`idtm_pingouin` ASC),
  INDEX `fk_tm_pingouin_instance_materiel1_idx` (`instance_materiel_idinstance_materiel` ASC),
  CONSTRAINT `fk_tm_pingouin_instance_materiel1` FOREIGN KEY (`instance_materiel_idinstance_materiel`) REFERENCES `matos2`.`instance_materiel` (`idinstance_materiel`) ON DELETE NO ACTION ON UPDATE NO ACTION) ENGINE = InnoDB
```

La requête SQL de création de la table a été construite.

L'étape 4 consiste à exécuter l'ordre SQL de l'étape 3. La nouvelle table est alors créée, et un nouvel enregistrement est fait dans **materiel**.

Page permettant d'ajouter un nouveau type de materiel dans la base - Mozilla Firefox

localhost/ajouternouveau4.php

Page permettant d'ajouter un nouveau type de materiel dans la base 4/4

pingouin

```
CREATE TABLE IF NOT EXISTS `matos2`.`tm_pingouin` (
  `idtm_pingouin` INT NOT NULL AUTO INCREMENT,
  `instance_materiel_idinstance_materiel` INT NOT NULL,
  `nom` VARCHAR(100),
  `taille` INT,
  `couleur` VARCHAR(100),
  PRIMARY KEY (`idtm_pingouin`),
  UNIQUE INDEX `idtm_pingouin_UNIQUE` (`idtm_pingouin` ASC),
  INDEX `fk_tm_pingouin_instance_materiel1_idx` (`instance_materiel_idinstance_materiel` ASC),
  CONSTRAINT `fk_tm_pingouin_instance_materiel1` FOREIGN KEY (`instance_materiel_idinstance_materiel`) REFERENCES `matos2`.`instance_materiel` (`idinstance_materiel`) ON DELETE NO ACTION ON UPDATE NO ACTION) ENGINE = InnoDB
```

INSERT INTO `matos2`.`materiel` (`nom`) VALUES ('pingouin')

La table du nouveau matériel a été créée.

Pour simplifier les choses, ces 4 étapes correspondent à 4 fichiers PHP.

Récupérer ensuite les fichiers **totale_nl.txt**, **ajouternouveau.php**, **ajouternouveau2.php**, **ajouternouveau3.php**, **ajouternouveau4.php**, **ajouternouveau.js** et **style4.css** sur Moodle, et modifier les pour être sûr que les informations de connexion (login, password, nom de la base) sont correctes.

En utilisant la numérotation des lignes données dans le fichier **totale_nl.txt** :

- modifier les lignes 2 à 4 du fichier **ajouternouveau.php**

Il faut ici démarrer une nouvelle session PHP

- modifier la ligne 23 du fichier **ajouternouveau.php**

Il faut vérifier ici les infos de login dans la base

- modifier la ligne 56 du fichier **ajouternouveau.php**

Il faut ici déclarer **nomMaterielErr**, **nomMateriel**, et **errors**.

- modifier la ligne 62 du fichier **ajouternouveau.php**

Comment tester si le nom rentré n'est pas vide ?

- modifier la ligne 67 du fichier **ajouternouveau.php**

Puis tester si le nom entré n'est pas déjà présent dans la base ?

- modifier la ligne 76 du fichier **ajouternouveau.php**

Si il n'y a pas d'erreurs, il faut utiliser la fonction **header()** de PHP pour rediriger l'utilisateur vers la page suivante. Il faut également utiliser la supervariable **\$_SESSION** pour passer à la page suivante le nom du nouveau matériel.

- modifier la ligne 82 du fichier **ajouternouveau.php**

Il faut ici créer le champ texte et le préremplir si on a déjà un nom de matériel.

- modifier les lignes 90 à 92 du fichier **ajouternouveau2.php**

Un peu comme avant.

- modifier la ligne 111 du fichier **ajouternouveau2.php**

Les infos de connexion.

- modifier les lignes 184 à 186 du fichier **ajouternouveau3.php**

Un peu comme avant.

- modifier la ligne 205 du fichier **ajouternouveau3.php**

Les infos de connexion.

- En vous inspirant du fichier **matos3_ddl.sql**, modifier la ligne 239 du fichier **ajouternouveau3.php** (attention, question difficile !)
- modifier les lignes 249 à 251 du fichier **ajouternouveau4.php**
- modifier la ligne 270 du fichier **ajouternouveau4.php**
- modifier les lignes 288 à 289 du fichier **ajouternouveau4.php**

Partie IV : On envoie des courriels !

En vous inspirant du code fourni dans totale_nl.txt (à partir de la ligne 339), **example_simple.c**, écrire un programme C permettant de prévenir tous les élèves de la base de données que la version 3 de la base de données **matos3** est maintenant disponible.

Ce programme utilise une bibliothèque qui est fournie dans le fichier **smtp-client.tgz**. Examiner le script **cmd1.sh** pour savoir comment recompiler l'application **example_simple.c**.