

TP3 : Interaction avec une BD via PHP

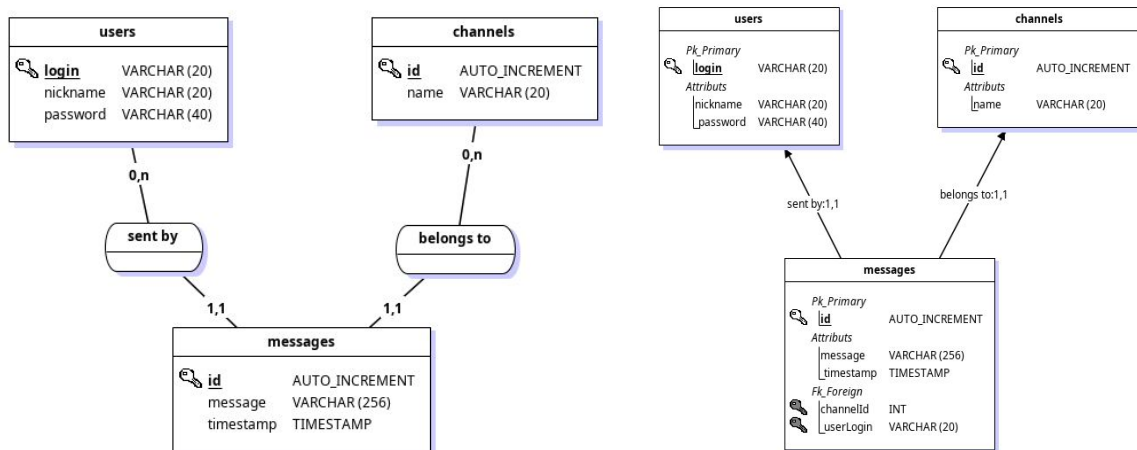
Ayoub KARINE (ayoub.karine@isen-ouest.yncrea.fr)

Objectifs

Dans ce TP vous allez :

- Créer une base de données
- Mettre en place une connexion et puis une interaction avec la base de données. Pour cela, vous utiliserez le langage PHP associé à l'extension PHP Data Objects (PDO)
- Récupérer des valeurs à partir d'une URL

Les modèles conceptuel et physique de données sont respectivement :



Création et remplissage de la BD

1. Via la WSL, déplacez vous vers le chemin du serveur apache et tapez code .

```
akarine@DESKTOP-FTU8G8F:/var/www/html$ code .
```

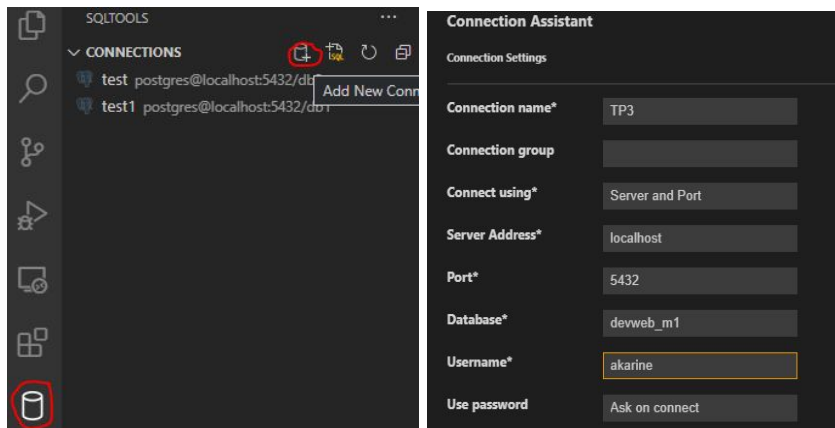
2. Créer un dossier, sur VS Code, nommé TP3
3. Télécharger le dossier ["BDD"](#) tout en le mettant dans le dossier TP3

4. Créer une base de donnée, via le terminal WSL intégré dans VS Code, appelée "devweb_m1"

```
akarine@DESKTOP-FTU8G8F:/var/www/html$ sudo service postgresql start
* Starting PostgreSQL 12 database server
akarine@DESKTOP-FTU8G8F:/var/www/html$ sudo -u postgres psql
psql (12.2 (Ubuntu 12.2-4))
Type "help" for help.

postgres=# create database devweb_m1;
WARNING: could not flush dirty data: Function not implemented
CREATE DATABASE
```

5. Connectez la BD "devweb_m1" avec VS Code



6. Exécutez, dans VS Code, les codes "models.sql" puis "data.sql" du dossier BDD
7. Visualiser le contenu des tables : "channels", "messages" et "users"

Constantes de connexion à la base de données

8. Créez un dossier "php" dans lequel vous créez un fichier "constants.php"
9. Afin de se connecter à la base de données, définissez, dans le fichier "constants.php", les constantes de connexion suivantes :
 - a. DB_USER qui vaut "votrelogin" ;
 - b. DB_PASSWORD qui vaut "votreMotdePasse" ;
 - c. DB_NAME qui vaut devweb_m1 ;
 - d. DB_SERVER qui vaut 127.0.0.1
 - e. DB_PORT qui vaut 5432

Connexion à la base de données

10. Dans le dossier "php", créez un fichier "database.php" dans lequel vous écrirez la fonction nommée dbConnect. Celle-ci doit tenter d'établir la connexion et doit renvoyer l'objet PDO créé (ou false en cas d'erreur). Vous utiliserez les exceptions pour traiter les erreurs de connexions. Pour pouvoir avoir accès aux constantes définies précédemment, pensez à inclure le fichier "constants.php"

11. Pour établir la connexion à la base de données, créez un fichier "chat.php" dans le dossier "php". Dans celui-ci,

- a. activer le débogage PHP en insérant le code suivant :

```
1 ini_set('display_errors', 1);
2 error_reporting(E_ALL);
```

- b. appelez la fonction dbConnect.

12. Vérifiez que la connexion à la base de données est fonctionnelle en effectuant la requête suivante dans votre navigateur (rien ne va s'afficher en cas de succès de connexion):

GET php/chat.php Test la connexion à la base de données

Requête pour la liste des salons (channels)

13. Créez un fichier sql dans le dossier BD dans lequel vous écrivez une requête qui affiche la liste des salons (channels). Affichez le résultat dans VS Code. Pour écrire la requête SQL, référez-vous au modèle physique de données (c.f. début de TP)

14. Ecrivez, dans le fichier "database.php", la fonction nommée dbGetChannels. Celle-ci prend en paramètre l'objet PDO, représentant la base de données, créée à l'étape précédente. Elle renvoie la liste des salons en cas de succès, false sinon.

15. Pour récupérer la liste des salons du chat, l'utilisateur devra effectuer la requête suivante à l'aide de son navigateur :

GET php/chat.php?request=channels Liste des salons en JSON

Pour traiter cette requête, modifiez le fichier "chat.php" pour renvoyer le résultat de la fonction "dbGetChannels". Notez que l'appel de cette fonction ne doit se faire que lorsque le paramètre "request" de la requête, qui est de type GET, vaut "channels"

Le résultat doit être comme suit :

```
Array ( [0] => Array ( [id] => 1 [name] => General ) [1] => Array ( [id] => 2 [name] => Musique ) [2] => Array ( [id] => 3 [name] => Voyage ) [3] => Array ( [id] => 4 [name] => Cinema ) [4] => Array ( [id] => 5 [name] => Bricolage ) )
```

16. Modifiez "chat.php" pour renvoyer, cette fois-ci, la liste des salons dans le format JSON.

Pour un bon affichage du format JSON, installez, sur chrome, l'extension [JSONVIEW](#).

Le résultat doit être comme suit :

```
localhost/M1/TP2/php/chat.php?request=channels
[
  {
    id: 1,
    name: "General"
  },
  {
    id: 2,
    name: "Musique"
  },
  {
    id: 3,
    name: "Voyage"
  },
  {
    id: 4,
    name: "Cinema"
  },
  {
    id: 5,
    name: "Bricolage"
  }
]
```