

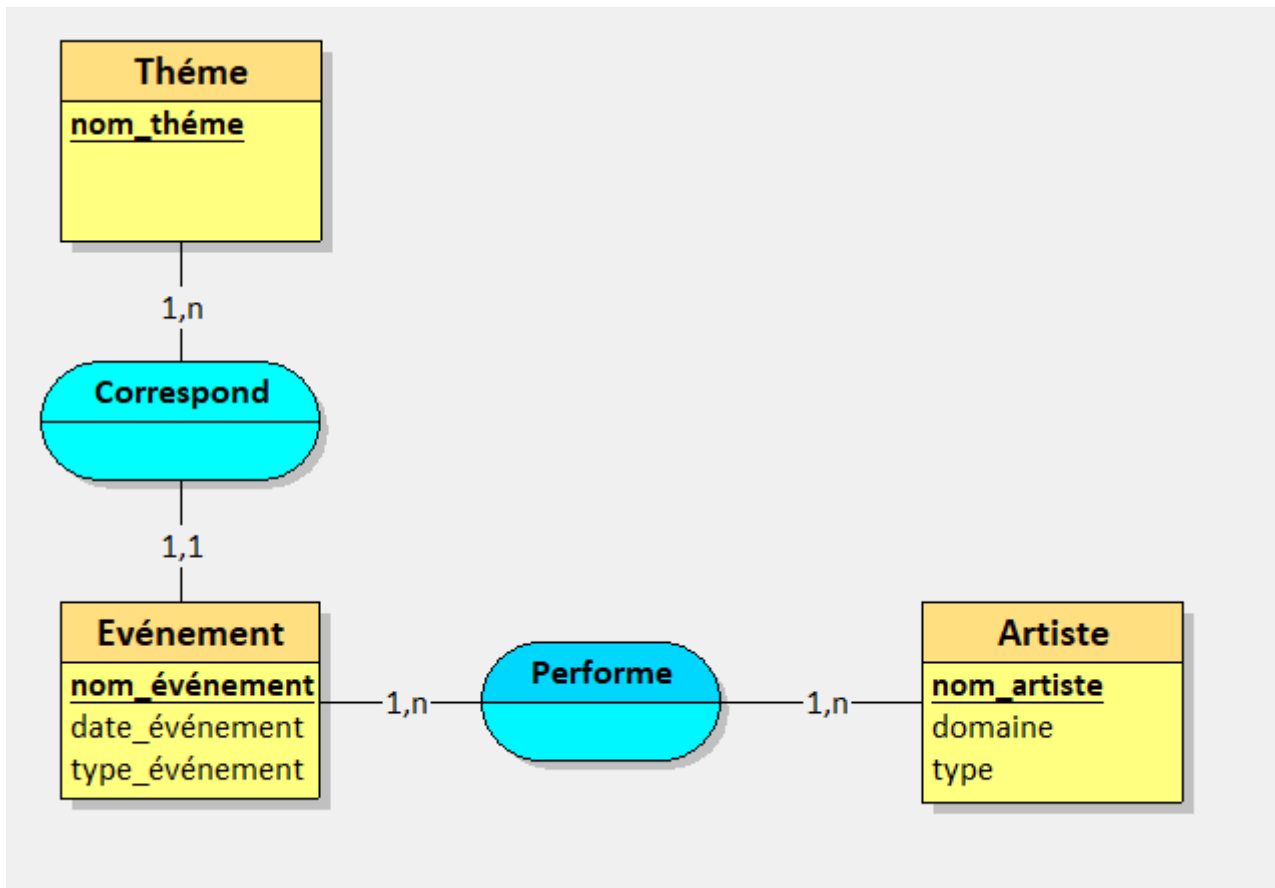
Compte Base de données :

15/11/2024 :

Objectif :

Créer une base de données et une table avec des instructions SQL.

Le MCD de notre première table (créer avec looping):



Son MLD :

Thème = (nom_thème VARCHAR(50));

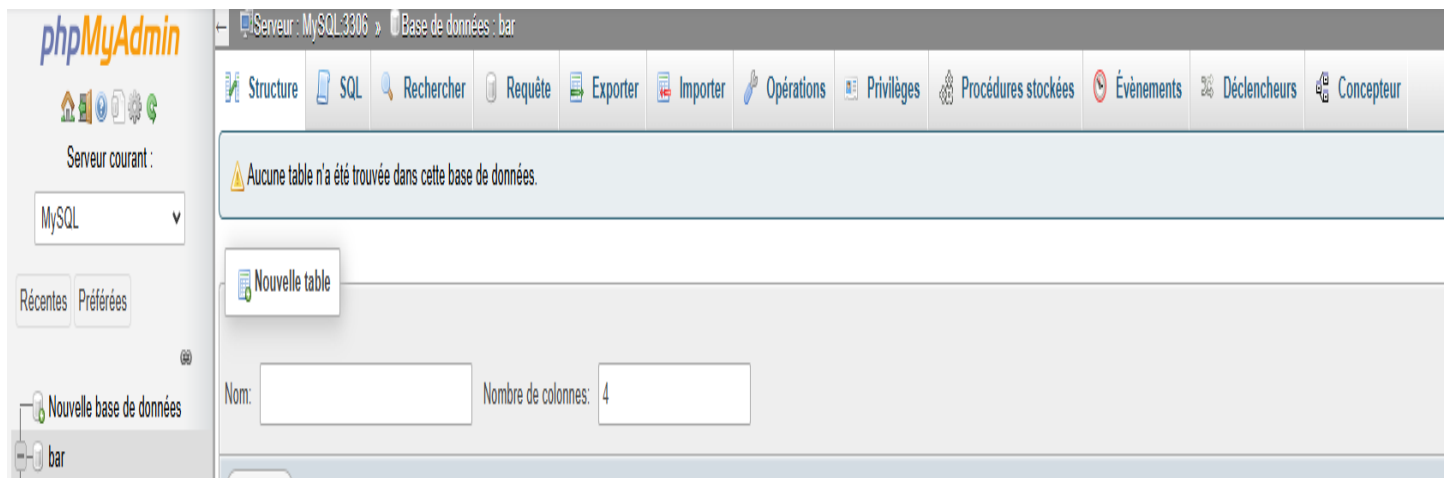
Artiste = (nom_artiste VARCHAR(50), domaine VARCHAR(50), type VARCHAR(50));

Événement = (nom_événement VARCHAR(50), date_événement DATE, type_événement VARCHAR(50), #nom_thème);

Performe = (#nom_événement, #nom_artiste);

On ouvre Wamp server et on crée une nouvelle base de données avec la commande **create database bar;**

On a notre base mais elle est vide pour l'instant :



L'avantage de looping c'est qu'il nous donne la traduction de notre MCD en langage SQL on va créer chaque table en recopiant les instructions :

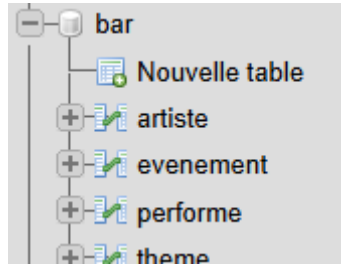
```
SQL
CREATE TABLE Théme(
  nom_thème VARCHAR(50),
  PRIMARY KEY(nom_thème)
);

CREATE TABLE Artiste(
  nom_artiste VARCHAR(50),
  domaine VARCHAR(50),
  type VARCHAR(50),
  PRIMARY KEY(nom_artiste)
);

CREATE TABLE Événement(
  nom_événement VARCHAR(50),
  date_événement DATE,
  type_événement VARCHAR(50),
  nom_thème VARCHAR(50) NOT NULL,
  PRIMARY KEY(nom_événement),
  FOREIGN KEY(nom_thème) REFERENCES Théme(nom_thème)
);

CREATE TABLE Performe(
  nom_événement VARCHAR(50),
  nom_artiste VARCHAR(50),
  PRIMARY KEY(nom_événement, nom_artiste),
  FOREIGN KEY(nom_événement) REFERENCES Événement(nom_événement),
  FOREIGN KEY(nom_artiste) REFERENCES Artiste(nom_artiste)
);
```

On a notre base de données avec nos tables :



A chaque fois qu'on va créer une nouvelle base de données on va d'abord créer son MCD sur looping. Puis looping nous traduira les requêtes SQL, pour qu'on puisse créer chaque table.

22/11/2024 :

Objectif :

On va rajouter des valeurs dans notre base de données.

On va commencer par réfléchir aux valeurs qu'on veut ajouter dans chaque table :

Pour la table artiste : On va mettre un nom (on va en inventer un), le domaine de performance (musique par exemple), le type (genre musical par exemple).

Pour la table thème : On va juste créer un nom de thème.

Pour la table evenement : On va créer un nom d'événement, une date, le type d'événement (un concert pour un seul groupe, une scène ouverte), le nom du thème (même nom que dans notre table thème).

Pour la table performe : On va mettre le nom de l'événement (même nom que dans notre table evenement), le nom de l'artiste (même nom que dans notre table artiste), la date.

Voici les instructions SQL :

Pour la table thème :

```
1 INSERT INTO theme (nom_thème)
2 VALUES ('Nouveau talents')
```

Pour la table evenement :

```
1 INSERT INTO evenement (nom_événement, date_événement, type_événement, nom_thème)
2 VALUES ('New stars', '2025/06/07', 'Concert', 'Nouveau talents')
```

Pour la table artiste :

```
1 INSERT INTO artiste (nom_artiste, domaine, type)
2 VALUES ('rubber band', 'musique', 'electro')
```

Pour la table performe :

```
INSERT INTO performe (nom_événement, nom_artiste, date_performance)
VALUES ('New stars', 'rubber band', '2025/06/07')
```

29/11/2024 :

Objectif :

On va faire trois requêtes SQL dans notre base de donnée.

Pour notre première requête on va commencer par trier tous les artistes qui performeront à l'événement New stars :

L'instruction SQL est :

```
SELECT * FROM `performe` WHERE nom_événement = "New stars";
```

Cette instructions affiche 3 résultats :

Options







|  | | | | nom_événement | nom_artiste | date_performance | | | |
|---|---|--------|---|---------------|---|------------------|-----------|-------------|------------|
| <input type="checkbox"/> |  | Éditer |  | Copier |  | Supprimer | New stars | MILI | 2025-06-07 |
| <input type="checkbox"/> |  | Éditer |  | Copier |  | Supprimer | New stars | rubber band | 2025-06-07 |
| <input type="checkbox"/> |  | Éditer |  | Copier |  | Supprimer | New stars | théa | 2025-06-07 |

Pour notre deuxième requête on veut voir tous les artistes qui appartiennent au genre musical rock.

L'instruction SQL est :

```
SELECT * FROM `artiste` WHERE type = "rock";
```

Cette instruction affiche 2 résultats :

| | | | | | nom_artiste | domaine | type | | |
|--------------------------|---|--------|---|--------|---|-----------|--------------|---------|------|
| <input type="checkbox"/> |  | Éditer |  | Copier |  | Supprimer | rubber band | musique | rock |
| <input type="checkbox"/> |  | Éditer |  | Copier |  | Supprimer | Jeff Buckley | musique | rock |

Pour notre troisième requête on veut voir tous les événements qui auront lieu au mois de juin 2025.

L'instruction SQL est :

```
SELECT * FROM evenement WHERE date_événement BETWEEN "2025-06-01" AND "2025-06-30";
```

Cette instruction affiche 2 résultats :

| <div><div><div></div><div></div><div></div></div></div> | | | | nom_événement | date_événement | type_événement | nom_thème | | | |
|---|--|--------|--|---------------|--|----------------|-----------|------------|---------------|-----------------|
| <input type="checkbox"/> | <div><div></div><div></div><div></div></div> | Éditer | <div><div></div><div></div><div></div></div> | Copier | <div><div></div><div></div><div></div></div> | Supprimer | New stars | 2025-06-07 | Scène ouverte | Nouveau talents |
| <input type="checkbox"/> | <div><div></div><div></div><div></div></div> | Éditer | <div><div></div><div></div><div></div></div> | Copier | <div><div></div><div></div><div></div></div> | Supprimer | Dopamine | 2025-06-13 | Scène ouverte | Electric joy |

06/12/2024 :

Objectif :

On va faire une requête jointe dans notre base de données.

Une requête jointe fait appel à plusieurs tables différentes. Par exemple on va savoir le genre musical de tous les artistes qui vont performer à l'événement dopamine.

Notre requête SQL est la suivante :

```
SELECT artiste.type FROM artiste, performe WHERE performe.nom_événement = "Dopamine" AND artiste.nom_artiste = performe.nom_artiste.
```

Notre résultats est le suivant :

| type |
|--------------|
| electronique |
| hyperpop |

Avec cette requête on sait qu'à l'événement Dopamine les genres musicaux représentés seront l'électronique et l'hyperpop.

06/12/2024 :

Objectif :

On va faire une requête GROUP BY.

On va faire une requête GROUP BY pour savoir le nombre d'artiste qui vont performer dans chaque événement.

Notre requête SQL est la suivante :

```
SELECT nom_événement, COUNT(*) FROM performe GROUP BY nom_événement
```

Le résultat est le suivant :

| nom_événement | COUNT(*) |
|---------------|----------|
| Blue nights | 1 |
| Dopamine | 2 |
| New stars | 3 |

Avec cette instruction on voit que l'événement Blue nights aura 1 artiste, Dopamine aura 2 artistes, New stars aura 3 artistes.

17/01/2025 :

Objectif :

On va faire une requête Insert dans chaque table, puis une requete update et delete.

On doit mettre à jour les donnés de notre table pour ça on utilise une requête INSERT

Imaginons qu'on veut ajouter un artiste :

La requête SQL est la suivante :

```
INSERT INTO artiste (nom_artiste, domaine, type) VALUES ('Tei shi','musique','pop');
```

Le résultat est le suivant :

| | | | | | | | nom_artiste | domaine | type |
|--------------------------|---|--------|---|--------|---|-----------|--------------|---------|--------------|
| <input type="checkbox"/> |  | Éditer |  | Copier |  | Supprimer | rubber band | musique | rock |
| <input type="checkbox"/> |  | Éditer |  | Copier |  | Supprimer | Jeff Buckley | musique | rock |
| <input type="checkbox"/> |  | Éditer |  | Copier |  | Supprimer | MILI | musique | electronique |
| <input type="checkbox"/> |  | Éditer |  | Copier |  | Supprimer | théa | musique | hyperpop |
| <input type="checkbox"/> |  | Éditer |  | Copier |  | Supprimer | Tei shi | musique | pop |

On veut créer un nouvel événement :

La requête SQL :

```
INSERT INTO evenement (nom_événement, date_événement, type_événement, nom_thème)  
VALUES ('Enkephaline', '2025/12/12', 'Concert', 'Trans-pop')
```

Le résultat :

| ← T → | | | nom_événement | date_événement | type_événement | nom_thème | |
|--------------------------|--|--|---|----------------|----------------|---------------|-----------------|
| <input type="checkbox"/> |  Éditer |  Copier |  Supprimer | New stars | 2025-06-07 | Scène ouverte | Nouveau talents |
| <input type="checkbox"/> |  Éditer |  Copier |  Supprimer | Blue night | 2025-05-19 | Concert | Real folk blues |
| <input type="checkbox"/> |  Éditer |  Copier |  Supprimer | Dopamine | 2025-06-13 | Scène ouverte | Electric joy |
| <input type="checkbox"/> |  Éditer |  Copier |  Supprimer | Enkephaline | 2025-12-12 | Concert | Trans-pop |

Deux de nos artistes vont performer au nouvel événement.

La requête SQL :

```
INSERT INTO performe (nom_événement, nom_artiste, date_performance) VALUES ('Enkephaline', 'Tei shi', '2025/12/12'), ('Enkephaline', 'Théa', '2025/12/12')
```

Le résultat :

| | | | nom_événement | nom_artiste | date_performance | | | | |
|--------------------------|---|--------|---|-------------|---|-----------|-------------|--------------|------------|
| <input type="checkbox"/> |  | Éditer |  | Copier |  | Supprimer | New stars | rubber band | 2025-06-07 |
| <input type="checkbox"/> |  | Éditer |  | Copier |  | Supprimer | New stars | théa | 2025-06-07 |
| <input type="checkbox"/> |  | Éditer |  | Copier |  | Supprimer | Blue nights | Jeff Buckley | 2025-05-19 |
| <input type="checkbox"/> |  | Éditer |  | Copier |  | Supprimer | New stars | MILI | 2025-06-07 |
| <input type="checkbox"/> |  | Éditer |  | Copier |  | Supprimer | Dopamine | MILI | 2025-06-13 |
| <input type="checkbox"/> |  | Éditer |  | Copier |  | Supprimer | Dopamine | théa | 2025-06-13 |
| <input type="checkbox"/> |  | Éditer |  | Copier |  | Supprimer | Enkephaline | Tei shi | 2025-12-12 |
| <input type="checkbox"/> |  | Éditer |  | Copier |  | Supprimer | Enkephaline | Théa | 2025-12-12 |

On va ajouter le nouveau thème.

La requête SQL :

```
INSERT INTO theme (nom_thème) VALUES ('Trans-pop')
```

On c'est tromper dans la date de l'événement Enkephaline. On va utiliser une requête update.

La requête :

```
UPDATE `evenement` SET `date_événement` = '2025-12-19' WHERE `evenement`.`nom_événement` = 'Enkephaline';
```

Le résultat :

| T→ | | | nom_événement | date_événement | type_événement | nom_thème | |
|--------------------------|--|--|---|----------------|----------------|---------------|-----------------|
| <input type="checkbox"/> |  Éditer |  Copier |  Supprimer | New stars | 2025-06-07 | Scène ouverte | Nouveau talents |
| <input type="checkbox"/> |  Éditer |  Copier |  Supprimer | Blue night | 2025-05-19 | Concert | Real folk blues |
| <input type="checkbox"/> |  Éditer |  Copier |  Supprimer | Dopamine | 2025-06-13 | Scène ouverte | Electric joy |
| <input type="checkbox"/> |  Éditer |  Copier |  Supprimer | Enkephaline | 2025-12-19 | Concert | Trans-pop |

Mili ne participe plus à l'événement . On va supprimer sa ligne.

La requête :

```
DELETE FROM `performe` WHERE `performe`.`nom_événement` = 'New stars' AND  
`performe`.`nom_artiste` = 'MILI'
```

24/01/2025

Objectif :










On fait une requête SQL avec une sous requête.

On veut connaître le nom de tous les thèmes auxquelles un artiste participe.

La requête :

```
SELECT nom_thème  
FROM evenement  
WHERE nom_événement IN  
(  
    SELECT nom_événement  
    FROM performe  
    WHERE nom_artiste = 'théa'  
);
```

Le résultat :

| | | | | nom_thème |
|--------------------------|--|--|---|-----------------|
| <input type="checkbox"/> |  Éditer |  Copier |  Supprimer | Nouveau talents |
| <input type="checkbox"/> |  Éditer |  Copier |  Supprimer | Electric joy |
| <input type="checkbox"/> |  Éditer |  Copier |  Supprimer | Trans-pop |

31/01/2025

Objectif :

On rajoute une table pour récupérer les résultats de notre enquête de satisfaction.

La requête :

```
CREATE TABLE satisfaction  
(  
    ID_satisfaction INT,  
    nom_pseudo VARCHAR(50),
```

```
role VARCHAR (50),  
contact VARCHAR (50),  
date_visite DATE,  
accueil VARCHAR (50),  
qualite VARCHAR (50),  
prix VARCHAR (50),  
spectacle VARCHAR (50),  
locaux VARCHAR (50),  
ambiance VARCHAR (50),  
suggestions VARCHAR (50),  
PRIMARY KEY (ID_Satisfaction)  
);
```

Voilà notre nouvelle table :

| nom_pseudo | role | contact | date_visite | accueil | qualité | prix | spectacle | locaux | ambiance | suggestions |
|------------|------|---------|-------------|---------|---------|------|-----------|--------|----------|-------------|
|------------|------|---------|-------------|---------|---------|------|-----------|--------|----------|-------------|