

phpMyAdmin

Créer une base de données utilisable
dans un programme



Compétence demandée :
**Savoir créer des bases de données et
des tables**

Une base de données relationnelle
est un ensemble de tables

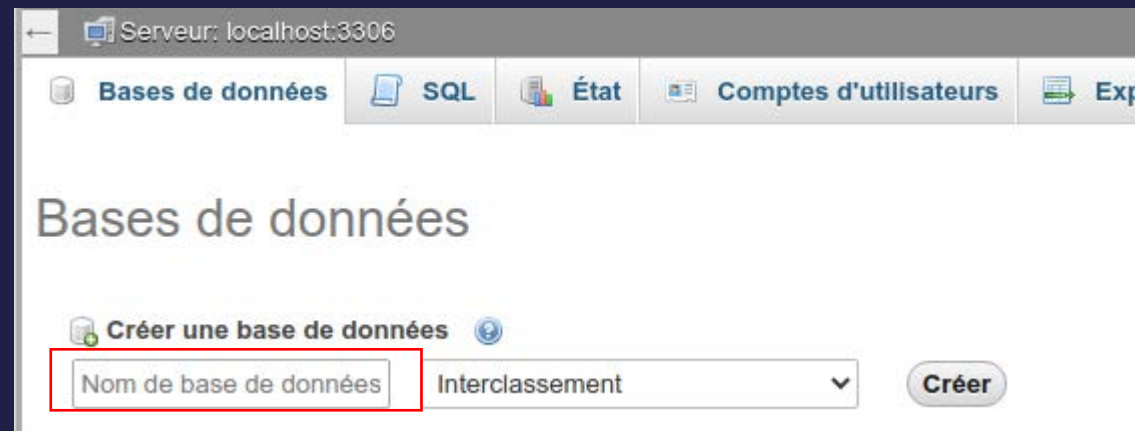
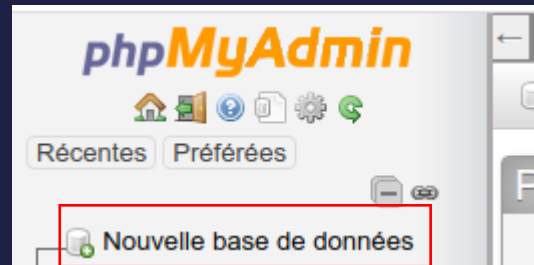
1. Créer une base de données vide
2. Créer une table
3. Ajouter les colonnes
4. Ajouter les contraintes FK (innoDB)
5. Vérifier sa base de données

1. Créer une base de données vide
2. Créer une table
3. Ajouter les colonnes
4. Ajouter les contraintes FK (innoDB)
5. Vérifier sa base de données

Créer une base de données est créer **un ensemble vide dans lequel nous créerons nos tables** par la suite.

Habituellement, en MySQL, **nous créons 1 base de données par projet**. Ainsi, pour chaque projet que vous gérez, vous devez avoir une base de données distincte pour chaque projet.

1. Créer une nouvelle base de données



Il n'existe pas de convention de nommage en base de données, mais nous recommandons d'utiliser **les minuscules et le snake case** dans le nom de la base de données.

Par ailleurs nous recommandons également que **le projet et la base de données portent le même nom.**

Il existe de nombreux interclassement qui dicte comment les caractères seront encodés avant d'être stockés.

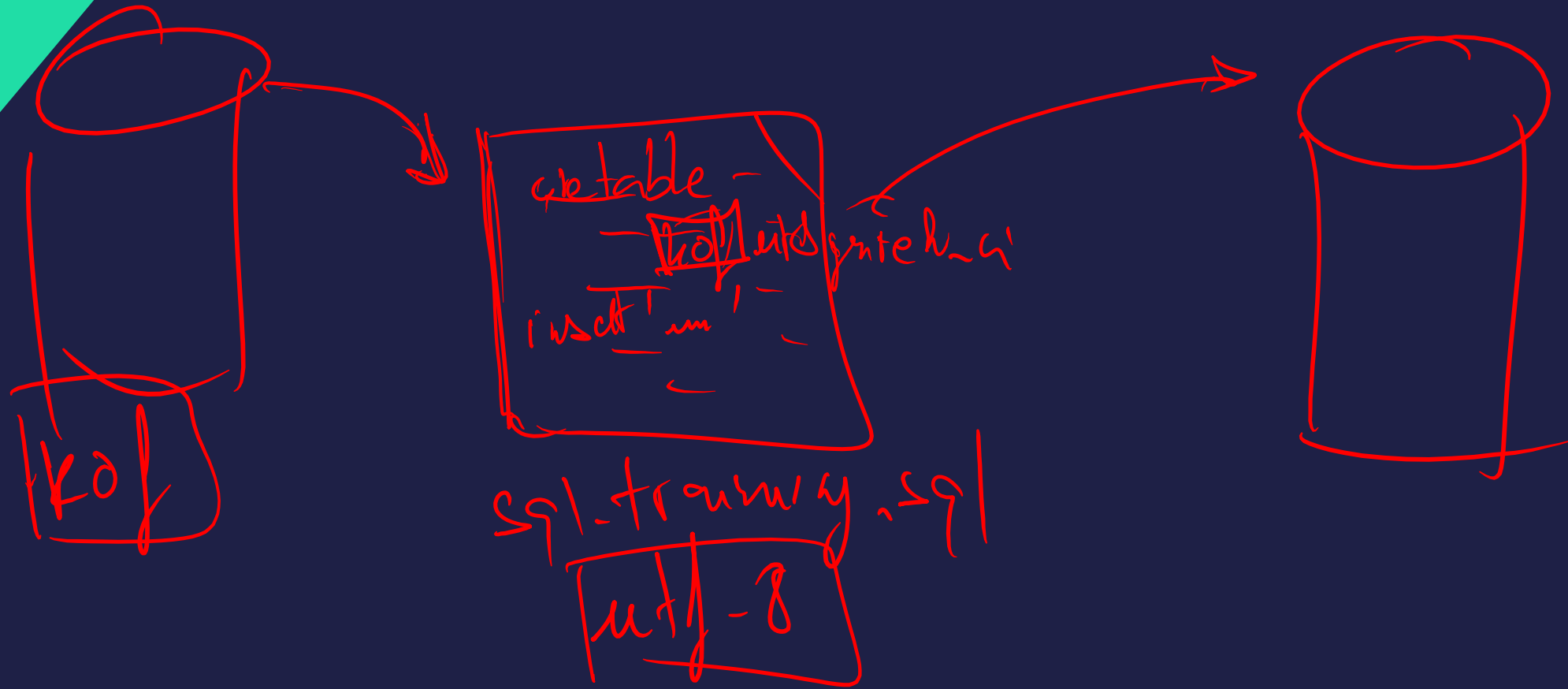
Les 2 les plus utilisés sont :

- `utf8_general_ci`
- `utf8mb4_0900_ai_ci` (qui se veut être une version améliorée de `utf8_general_ci`)

Vous pouvez utiliser soit l'un soit l'autre ! 😊

`utf8_general_ci` : est beaucoup plus populaire dans les projets et sur internet

`utf8mb4_0900_ai_ci` se veut être une version améliorée de `utf8_general_ci` en performance



1. Créer une base de données vide
2. Créer une table
3. Ajouter les colonnes
4. Ajouter les contraintes FK (innoDB)
5. Vérifier sa base de données

Afin de créer une table en base de données il faut connaître à l'avance le **nom de la table** et également le **nombre de colonnes** dans cette table.

N'oubliez pas d'abord de cliquer sur la base de données dans laquelle on veut créer la table afin de la sélectionner

→ use database;

2. Créer une nouvelle table

Structure SQL Rechercher Requête Export plus

⚠ Aucune table n'a été trouvée dans cette base de données.

Nouvelle table

Nom: Nombre de colonnes:

Exécuter

1. Créer une base de données vide
2. Créer une table
3. Ajouter les colonnes
4. Ajouter les contraintes FK (innoDB)
5. Vérifier sa base de données

Cela peut paraître déroutant mais chaque colonne est ajoutée en ligne. Dans la page de création de table, chaque ligne correspond à une colonne.

3. Ajouter les colonnes

Serveur: localhost:3306 » Base de données: airbnb » Table: logements

Afficher Structure SQL Rechercher Insérer Export Import Privileges Opérations Suivi Déclencheurs

Nom de table: logements Ajouter 1 colonne(s) Exécuter

Nom	Type	Taille/Valeurs*	Valeur par défaut	Interclassement	Attributs	Null	Index	A_I	Commentaires
<input type="text"/> <small>Choisissez à partir des colonnes centrales</small>	INT	<input type="text"/>	Aucune	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	---	<input type="checkbox"/>	<input type="text"/>
<input type="text"/> <small>Choisissez à partir des colonnes centrales</small>	INT	<input type="text"/>	Aucune	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	---	<input type="checkbox"/>	<input type="text"/>
<input type="text"/> <small>Choisissez à partir des colonnes centrales</small>	INT	<input type="text"/>	Aucune	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	---	<input type="checkbox"/>	<input type="text"/>
<input type="text"/> <small>Choisissez à partir des colonnes centrales</small>	INT	<input type="text"/>	Aucune	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	---	<input type="checkbox"/>	<input type="text"/>

Commentaires sur la table :

Interclassement :

Moteur de stockage : InnoDB

Rajouter les colonnes il faut faire attention aux choses suivantes :

- a. Nom des colonnes
- b. Type de valeurs
- c. Tailles maximales des valeurs
- d. Mise en place d'index (PRIMARY KEY)
- e. A_I = AUTO_INCREMENT

Type	Description	Exemple
INT	Nombre entier	3
VARCHAR	Texte à longueur maximal (rapide)	Hello ca va ?
TEXT	Texte sans limite (lent)	Article 1. Nous ...
DATE	Date (mais pas l'heure)	2021-04-03
FLOAT	Nombre à virgule	45,9398493
DOUBLE	Nombre à virgule avec beaucoup de chiffres après la virgule	67,03984039892039200393
DECIMAL	Nombre a virgule avec un nombre fixe après la virgule	123,23
DATETIME	Date et heure	2021-02-05 16:23:45
TIMESTAMP	Nombre de secondes depuis le 1 ^{er} janvier 1970	3847988

Type	Description	Exemple
INT	Nombre entier	3
VARCHAR	Texte à longueur maximal (rapide)	Hello ca va ?
TEXT	Texte sans limite (lent)	Article 1. Nous ...
DATE	Date (mais pas l'heure)	2021-04-03
FLOAT	Nombre à virgule	45,9398493
DOUBLE	Nombre à virgule avec beaucoup de chiffres après la virgule	67,03984039892039200393
DECIMAL	Nombre a virgule avec un nombre fixe après la virgule	123,23
DATETIME	Date et heure	2021-02-05 16:23:45
TIMESTAMP	Nombre de secondes depuis le 1 ^{er} janvier 1970	3847988

3. Ajouter les colonnes

Serveur: localhost:3306 » Base de données: airbnb » Table: logements

Afficher Structure SQL Rechercher Insérer Export Import Privileges Opérations Suivi Déclencheurs

Nom de table: logements Ajouter 1 colonne(s) Exécuter

Nom	Type	Taille/Valeurs*	Valeur par défaut	Interclassement	Attributs	Null	Index	A_I	Commentaires
<input type="text"/> <small>Choisissez à partir des colonnes centrales</small>	INT	<input type="text"/>	Aucune	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	---	<input type="checkbox"/>	<input type="text"/>
<input type="text"/> <small>Choisissez à partir des colonnes centrales</small>	INT	<input type="text"/>	Aucune	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	---	<input type="checkbox"/>	<input type="text"/>
<input type="text"/> <small>Choisissez à partir des colonnes centrales</small>	INT	<input type="text"/>	Aucune	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	---	<input type="checkbox"/>	<input type="text"/>
<input type="text"/> <small>Choisissez à partir des colonnes centrales</small>	INT	<input type="text"/>	Aucune	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	---	<input type="checkbox"/>	<input type="text"/>

Commentaires sur la table :

Interclassement :

Moteur de stockage : InnoDB

En bas de page il y a également la possibilité de sélectionner le moteur de stockage de la table.

Les 2 choix les plus répandus sont :

- MyISAM
- InnoDB

Un **moteur de stockage** est l'intermédiaire entre le système de gestion de base de données et la table qui comporte les enregistrements. Il **gère la lecture et l'écriture de la table**.

Un moteur de stockage est un **sous-programme** du SGBDR.

A votre avis, pourquoi existe-t-il plusieurs moteurs de stockage pour la table de la base de données ?



Différents moteurs de stockage existent
afin de mieux coller aux besoins de
chacun !

MyISAM est un moteur de stockage qui est **très rapide en lecture mais assez lent en écriture**.

InnoDB est un moteur de stockage qui est **rapide en écriture mais assez lent en lecture**.

Par ailleurs innoDB a la capacité d'effectuer des **vérifications supplémentaires** et de manière automatique pour chaque enregistrement de notre table.

Si vous avez un doute sur le moteur de
stockage à utiliser vous pouvez choisir
innoDB !



1. Créer une base de données vide
2. Créer une table
3. Ajouter les colonnes
4. Ajouter les contraintes FK (innoDB)
5. Vérifier sa base de données

Nous avons vu dans le cours précédent qu'il était possible de stocker des relations entre les tables soit sous forme de colonnes (1-n) soit sous la forme d'une nouvelle structure de table (n-n).

Dans les 2 cas elle comporte 1 (cas de la colonne) ou 2 clés étrangères (cas de la nouvelle structure de table) pour **identifier l'enregistrement de la contrepartie de la relation.**

En utilisant **innoDB**, vous serez dans la capacité de demander au moteur de stockage d'ajouter une mesure de sécurité : de vérifier automatiquement et à chaque changement de la base de données, de vérifier que toutes les relations soient bien respectées.

Cette vérification automatique s'appelle une contrainte de clé étrangère.

d'intégrité

Une contrainte de clé étrangère est utilisée pour empêcher les actions qui viendraient à détruire les relations entre les enregistrements.

C'est bien une vérification faite sur les enregistrements ! C'est-à-dire si un jour, nous décidons de supprimer un enregistrement, innoDB regardera d'abord si celui-ci n'est pas référencé ailleurs !

4. Ajouter les contraintes FK (InnoDB)

Serveur: localhost:3306 » Base de données: banque » Table: carte

Afficher Structure SQL Rechercher Insérer Export Import Privilèges Opérations Suivi Déclencheurs

Structure de table **Vue relationnelle**

Contraintes de clé étrangère

Actions Propriétés de la contrainte

Nom de la contrainte

ON DELETE RESTRICT ON UPDATE RESTRICT

+ Ajouter une contrainte

Colonne

Contrainte de clé étrangère (INNODB)

Base de données	Table	Colonne
banque	compte	id

+ Ajouter une colonne

FK de la table courante de la relation

PK de la table étrangère de la relation

+ Relations internes

Colonne descriptive : ---

Aperçu SQL Sauvegarder

4. Ajouter les contraintes FK (innoDB)

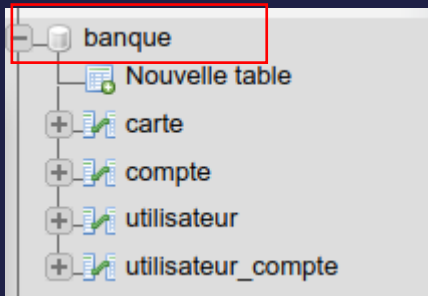
Si vous n'avez pas la vue relationnelle,
vous pouvez utiliser innoDB comme
moteur de stockage (rendez-vous dans
les “Opérations”)

1. Créer une base de données vide
2. Créer une table
3. Ajouter les colonnes
4. Ajouter les contraintes FK (innoDB)
5. Vérifier sa base de données

Vous pouvez voir le **schéma de votre base de données** en cliquant sur votre base de données pour la sélectionner et en consultant le **Concepteur**.

Le Concepteur vous permet d'avoir une **représentation graphique** de votre base de données, ce qui vous permettra de la vérifier aisément.

5. Vérifier sa base de données (concepteur)



Choisir “Concepteur”

