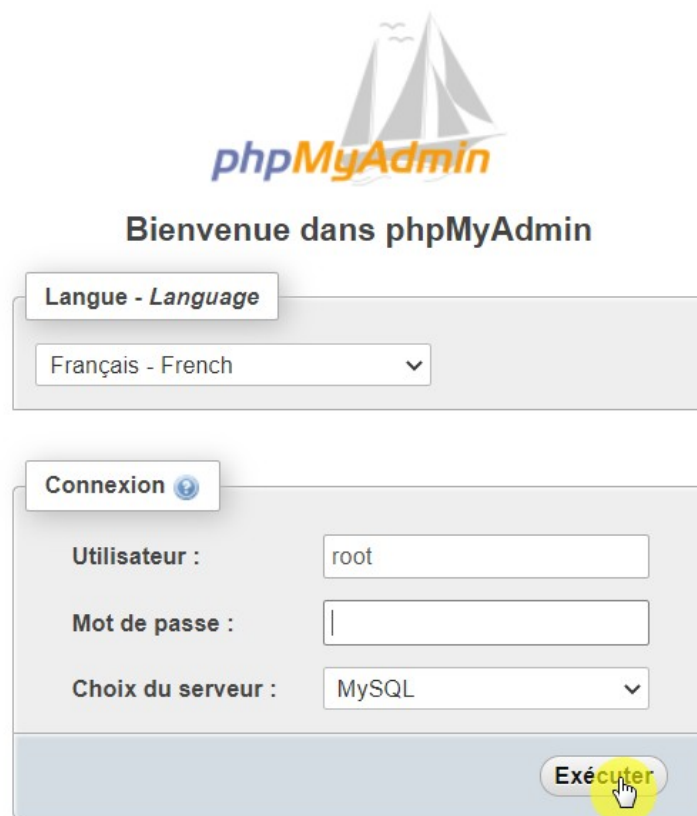


TD PHP MVC

Étape 7 : La base de données

Nous allons maintenant voir comment insérer des données dans notre template home/index.view.php

Nous allons donc créer une base de données en utilisant PHPMYAdmin, choisir MySQL avec Utilisateur : root et Mot de passe : vide.



The image shows the phpMyAdmin welcome screen. At the top is the phpMyAdmin logo. Below it, the text "Bienvenue dans phpMyAdmin" is displayed. There are two main sections: "Langue - Language" with a dropdown menu set to "Français - French", and "Connexion" (Connection) with fields for "Utilisateur :" (root), "Mot de passe :" (empty), and "Choix du serveur :" (MySQL). A yellow circle highlights the "Exécuter" button at the bottom right of the connection section.

Nous créons une base de donnée que nous appelons « tdphpmvc_db », avec une interclassement : utf8mb4_general_ci



The image shows the phpMyAdmin interface for creating a new database. The left sidebar shows the "Bases de données" (Databases) tab selected. The main area is titled "Bases de données" and shows the "Création d'une base de données" (Create a new database) section. The "Nom de la base de données" (Database name) field contains "tdphpmvc_db" and the "Collation" (Collation) dropdown is set to "utf8mb4_general_ci". A yellow circle highlights the "Créer" (Create) button.

Nous créons une première table que nous appelons « category » avec 4 colonnes

Structure SQL Rechercher Requête Exporter

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

Nouvelle table

Nom: category Nombre de colonnes: 4

Exécuter

Nous allons maintenant créer les 4 colonnes de cette table « category »

1. Choisir le moteur de stockage InnoDB (pour permettre l'ajout de clés étrangères)
2. Saisir le nom des colonnes :
id est la colonne permettant l'identification de manière unique d'une ligne
name sera le nom de la catégorie
description correspondra à sa description
image_path servira à stocker le chemin vers l'image représentant la catégorie
3. Renseigner le type des données
4. Renseigner la taille pour le type VARCHAR
5. Cocher A_I (auto-incrément) pour la colonne id (clé primaire de la table)
6. Cocher Null sauf pour l'id
7. Mettez NULL comme valeur par défaut sauf pour l'id

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

Nom	Type	Taille/Valeurs	Valeur par défaut	Interclassement	Attributs	Null	Index	A_I
id	INT		Aucun(e)			<input type="checkbox"/>	PRIMARY	<input checked="" type="checkbox"/>
name	VARCHAR	255	NULL			<input checked="" type="checkbox"/>	---	<input type="checkbox"/>
description	TEXT		NULL			<input checked="" type="checkbox"/>	---	<input type="checkbox"/>
image_path	VARCHAR	255	NULL			<input checked="" type="checkbox"/>	---	<input type="checkbox"/>

Commentaires de table : Interclassement : Moteur de stockage : InnoDB

Enregistrer

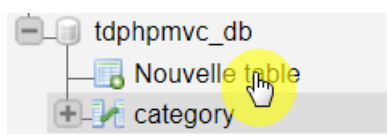
La table « category » est créée :-). Elle va nous servir à stocker les catégories de produits (Pour une pizzeria par exemple : Pizza, Pasta, Boissons, Desserts, etc ...)

Structure de table Vue relationnelle

#	Nom	Type	Interclassement	Attributs	Null	Valeur par défaut	Commentaires	Extra	Action
<input type="checkbox"/> 1	id	int(11)			Non	Aucun(e)		AUTO_INCREMENT	M
<input type="checkbox"/> 2	name	varchar(255)	utf8mb4_general_ci		Oui	NULL			M
<input type="checkbox"/> 3	description	text	utf8mb4_general_ci		Oui				M
<input type="checkbox"/> 4	image_path	varchar(255)	utf8mb4_general_ci		Oui	NULL			M

☐ Tout cocher Avec la sélection : Parcourir Modifier Supprimer Primaire Unique Index

Nous créons de la même manière une table « products »



Structure SQL Rechercher Requête Exporter Importer Opérations

Nom de table: Ajouter colonne(s)

Ne pas oublier de mettre le moteur de stockage sur InnoDB et de cocher l'auto-increment pour l'id.

Nom de table: Ajouter colonne(s)

Nom	Type	Taille/Valeurs*	Valeur par défaut	Interclassement	Attributs	Null	Index	Auto-incr.	Commentaires
<input type="text" value="id"/>	<input type="text" value="INT"/>	<input type="text"/>	<input type="text" value="Aucun(e)"/>	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	<input type="text" value="PRIMARY"/>	<input checked="" type="checkbox"/>	
<input type="text" value="name"/>	<input type="text" value="VARCHAR"/>	<input type="text" value="255"/>	<input type="text" value="NULL"/>	<input type="text"/>	<input type="text"/>	<input checked="" type="checkbox"/>	<input type="text" value="---"/>	<input type="checkbox"/>	
<input type="text" value="description"/>	<input type="text" value="TEXT"/>	<input type="text"/>	<input type="text" value="NULL"/>	<input type="text"/>	<input type="text"/>	<input checked="" type="checkbox"/>	<input type="text" value="---"/>	<input type="checkbox"/>	
<input type="text" value="image_path"/>	<input type="text" value="VARCHAR"/>	<input type="text" value="255"/>	<input type="text" value="NULL"/>	<input type="text"/>	<input type="text"/>	<input checked="" type="checkbox"/>	<input type="text" value="---"/>	<input type="checkbox"/>	
<input type="text" value="price"/>	<input type="text" value="FLOAT"/>	<input type="text"/>	<input type="text" value="NULL"/>	<input type="text"/>	<input type="text"/>	<input checked="" type="checkbox"/>	<input type="text" value="---"/>	<input type="checkbox"/>	
<input type="text" value="category_id"/>	<input type="text" value="INT"/>	<input type="text"/>	<input type="text" value="NULL"/>	<input type="text"/>	<input type="text"/>	<input checked="" type="checkbox"/>	<input type="text" value="---"/>	<input type="checkbox"/>	

Commentaires de table : Interclassement : Moteur de stockage :

La colonne « price » servira à stocker le prix des produits.

La colonne « category_id » correspondra à l'id de la catégorie à laquelle le produit appartient. C'est une référence vers la table category, on parle alors de clé étrangère (qu'il va falloir créer une fois la table product créée)



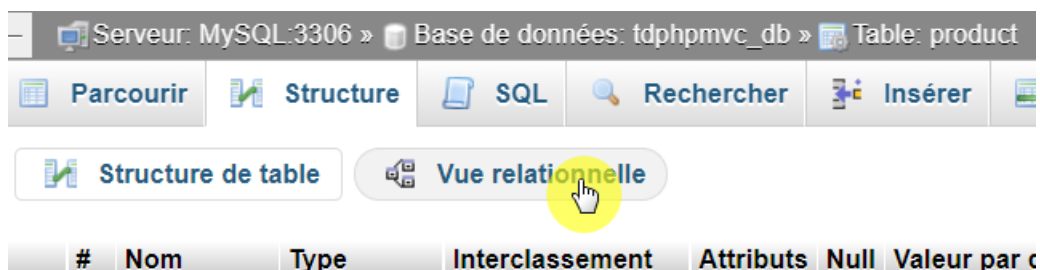
La table « product » est maintenant créée.

Structure de table Vue relationnelle

#	Nom	Type	Interclassement	Attributs	Null	Valeur par défaut	Commentaires	Extra	Action
<input type="checkbox"/> 1	id	int(11)			Non	Aucun(e)		AUTO_INCREMENT	M
<input type="checkbox"/> 2	name	varchar(255)	utf8mb4_general_ci		Oui	NULL			M
<input type="checkbox"/> 3	description	text	utf8mb4_general_ci		Oui				M
<input type="checkbox"/> 4	image_path	varchar(255)	utf8mb4_general_ci		Oui	NULL			M
<input type="checkbox"/> 5	price	float			Oui	NULL			M
<input type="checkbox"/> 6	category_id	int(11)			Oui	NULL			M

☐ Tout cocher Avec la sélection : Parcourir Modifier Supprimer Primaire Unique Index

Il nous reste à créer la relation entre product et category, pour cela nous allons ajouter une contrainte en cliquant sur Vue relationnelle



Nous créons une contrainte avec la nom FK_product_category comme indiqué ci-dessous

Contraintes de clé étrangère

Actions	Propriétés de la contrainte	Colonne	Contrainte de clé étrangère (INNODB)		
		Base de données	Table	Colonne	
	FK_product_category	category_id	tdphpmvc_db	category	id
	ON DELETE: RESTRICT	ON UPDATE: RESTRICT	+ Ajouter une colonne		

La colonne de la table product sur laquelle nous mettons la contrainte est category_id. Elle fait référence à la colonne id de la table category dans notre base de données tdphpmvc_db



La requête SQL a été exécutée avec succès.

Pour plus d'infos sur la conception de base de données, je vous invite à consulter ce tuto sur la méthode MERISE : <http://cours.pise.info/modelisation/index.htm>

Pour créer quelques lignes dans ces 2 tables, vous pouvez exécuter directement en ligne de commande les scripts suivant en cliquant sur SQL



Commencez par inserer des données dans la table category :

```
INSERT INTO `category` (`id`, `name`, `description`, `image_path`) VALUES  
(1, 'Les Pizza', 'des pizza ...', 'assets/img/category1.jpg'),  
(2, 'Les Boissons', 'des boissons ...', 'assets/img/category2.jpg'),  
(3, 'Les Desserts', 'des desserts ...', 'assets/img/category3.jpg');
```



Puis pour la table product :

```
INSERT INTO `product` (`id`, `name`, `description`, `price`, `image_path`, `category_id`) VALUES  
(1, '4 Fromages', 'sapien urna pretium nisl ut volutpat sapien arcu sed augue aliquam erat volutpat  
in congue etiam justo etiam pretium iaculis justo', 10.64, 'img/product1.jpg', 1),  
(2, 'Reine', 'libero non mattis pulvinar nulla pede ullamcorper augue a suscipit nulla elit ac nulla sed  
vel enim sit amet nunc viverra dapibus nulla', 10.02, 'img/product2.jpg', 1),  
(3, 'Margarita', 'maecenas ut massa quis augue luctus tincidunt nulla mollis molestie lorem quisque  
ut erat curabitur gravida nisi at nibh in', 10.57, 'img/product3.jpg', 1),  
(4, 'Napolitaine', 'volutpat sapien arcu sed augue aliquam erat volutpat in congue etiam justo etiam  
pretium iaculis justo in hac habitasse platea', 9.46, 'img/product4.jpg', 1),  
(5, 'Regina', 'habitasse platea dictumst maecenas ut massa quis augue luctus tincidunt nulla mollis  
molestie lorem quisque ut erat curabitur gravida nisi at', 10.01, 'img/product5.jpg', 1),  
(6, 'Orientale', 'ipsum dolor sit amet consectetur adipiscing elit proin interdum mauris non ligula  
pellentesque ultrices phasellus id sapien in sapien iaculis congue vivamus metus arcu adipiscing  
molestie hendrerit at', 9.75, 'img/product6.jpg', 1),  
(7, 'Chèvre-Miel', 'phasellus id sapien in sapien iaculis congue vivamus metus arcu adipiscing  
molestie hendrerit at vulputate vitae nisl aenean lectus pellentesque eget nunc donec quis orci  
eget orci', 10.56, 'img/product7.jpg', 1),  
(8, 'Tartiflette', 'in faucibus orci luctus et ultrices posuere cubilia curae mauris viverra diam vitae  
quam suspendisse potenti nullam porttitor lacus at turpis donec posuere', 8.19, 'img/product8.jpg',  
1),  
(9, 'Ch'ti', 'sem fusce consequat nulla nisl nunc nisl dui bibendum felis sed interdum venenatis  
turpis enim blandit mi in porttitor pede justo eu massa donec dapibus', 9.13, 'img/product9.jpg', 1);
```



Nous venons de créer 3 categories de produits et 9 produits appartenant à la catégorie 1 (Les Pizza)