

# Partie 4

---

# Correction Entity Diagram (Partie 2)

*Cours*

<https://jamboard.google.com/d/1r1LQ95BPJkR1ihUeEADxoH0Bk4d7-NLfVIPp8oQRWKM/edit?usp=sharing>

# Database design

*Cours*

Pour avoir une base de données construite de manière efficace il y a plusieurs règles de design que nous devons appliquer:

Voici un article qui résume les idées principales:

<https://www.codementor.io/@nigelpeck/design-principles-for-relational-data-tzzujzkiq>

Voici un article plus exhaustif sur le design de bases de données relationnelles.

[https://www.lucidchart.com/pages/database-diagram/database-design/#section\\_4](https://www.lucidchart.com/pages/database-diagram/database-design/#section_4)

Le concept de **normalization**, et ses trois premières formes y est particulièrement important.

# Table options on foreign key handling

*Cours*

<https://openclassrooms.com/fr/courses/1959476-administrez-vos-bases-de-donnees-avec-mysql/1965264-options-des-cles-etrangees>

# Correction queries (Partie 3)

*Cours*

Les corrections des queries sont dans le fichier Correction3.sql

Utilisez la base de données *sql\_commandes* liée au fichier *sql\_commandes.sql*

## Exercices globaux

1. Obtenir l'utilisateur ayant le prénom "Muriel" et le mot de passe "test11", sachant que l'encodage du mot de passe est effectué avec l'algorithme Sha1.
2. Obtenir la liste de tous les produits qui sont présent sur plusieurs commandes.
3. Obtenir la liste de tous les produits qui sont présent sur plusieurs commandes et y ajouter une colonne qui liste les identifiants des commandes associées.
4. Enregistrer le prix total à l'intérieur de chaque ligne des commandes, en fonction du prix unitaire et de la quantité
5. Obtenir le montant total pour chaque commande et y voir facilement la date associée à cette commande ainsi que le prénom et nom du client associé
6. (difficulté très haute) Enregistrer le montant total de chaque commande dans le champ intitulé "cache\_prix\_total"
7. Obtenir le montant global de toutes les commandes, pour chaque mois
8. Obtenir la liste des 10 clients qui ont effectué le plus grand montant de commandes, et obtenir ce montant total pour chaque client.
9. Obtenir le montant total des commandes pour chaque date
10. Ajouter une colonne intitulée "category" à la table contenant les commandes. Cette colonne contiendra une valeur numérique
11. Enregistrer la valeur de la catégorie, en suivant les règles suivantes :
  - o "1" pour les commandes de moins de 200€
  - o "2" pour les commandes entre 200€ et 500€
  - o "3" pour les commandes entre 500€ et 1.000€
  - o "4" pour les commandes supérieures à 1.000€
12. Créer une table intitulée "commande\_category" qui contiendra le descriptif de ces catégories
13. Insérer les 4 descriptifs de chaque catégorie au sein de la table précédemment créée
14. Supprimer toutes les commandes (et les lignes des commandes) inférieur au 1er février 2019. Cela doit être effectué en 2 requêtes maximum

# BDD Avancé

## *Exercices globaux*

Faites un planning de vos tâches pour le projet, pour les 6 séances restantes.

# Concepts de SQL Avancé

---



# Therorème acid et Transactions

*Cours*

Je vous invite tout d'abord à regarder la vidéo suivant:

<https://www.youtube.com/watch?v=AcqtAEzuojo>

Pour aller plus loin autour des transactions je vous conseille les articles suivants:

- <https://openclassrooms.com/fr/courses/1959476-administrez-vos-bases-de-donnees-avec-mysql/1970063-transactions>
- <https://www.mysqltutorial.org/mysql-transaction.aspx>
-

# Variables

<https://www.mysqltutorial.org/mysql-variables/>

*Cours*

# Procedures

*Cours*

<https://openclassrooms.com/fr/courses/1959476-administrez-vos-bases-de-donnees-avec-mysql/1971667-procedures-stockees>

# Error Handling

[https://www.mysqltutorial.org/mysql-error-handling-in-stored-procedure  
s/](https://www.mysqltutorial.org/mysql-error-handling-in-stored-procedures/)

*Cours*

# Views

*Cours*

<https://openclassrooms.com/fr/courses/1959476-administrez-vos-bases-de-donnees-avec-mysql/1973943-vues>

# Tables temporaires

*Cours*

<https://openclassrooms.com/fr/courses/1959476-administrez-vos-bases-de-donnees-avec-mysql/1975522-vues-materialisees>

# Materialized views

*Cours*

<https://openclassrooms.com/fr/courses/1959476-administrez-vos-bases-de-donnees-avec-mysql/1975522-vues-materialisees>

# Functions

*Cours*

- Built-in functions:  
[https://www.w3schools.com/mysql/mysql\\_ref\\_functions.asp](https://www.w3schools.com/mysql/mysql_ref_functions.asp)
- User defined functions:  
<https://www.mysqltutorial.org/mysql-stored-function/>



# Cursors

*Cours*

<https://openclassrooms.com/fr/courses/1959476-administrez-vos-bases-de-donnees-avec-mysql/1972565-gestionnaires-de-reseaux-cursors-et-utilisation-avancee> (2e partie)

# Triggers

*Cours*

<https://openclassrooms.com/fr/courses/1959476-administrez-vos-bases-de-donnees-avec-mysql/1973090-triggers>

# Variables locales & structures conditionnelles

*Cours*

<https://openclassrooms.com/fr/courses/1959476-administrez-vos-bases-de-donnees-avec-mysql/1972254-structurez-vos-instructions>

# User handling

*Cours*

<https://openclassrooms.com/fr/courses/1959476-administrez-vos-bases-de-donnees-avec-mysql/1975880-gestion-des-utilisateurs>

# Dynamic SQL / Prepared Statements

*Cours*

<https://www.databasestart.com/dynamic-sql/> (section avec MySQL)

[https://www.w3schools.com/php/php\\_mysql\\_prepared\\_statements.asp](https://www.w3schools.com/php/php_mysql_prepared_statements.asp)