P\_106-SpaceInvaders

Une image contenant capture d’écran, Graphique, pixel

Description générée automatiquement

Hadrien Aymon– Cin2a

ETML - Vennes

32 périodes

AMG

ETML-INF

[1 Introduction 3](#_Toc187246378)

[1.1 Contexte : 3](#_Toc187246379)

[1.2 Objectifs : 3](#_Toc187246380)

[2 Conception 3](#_Toc187246381)

[2.1 MCD/MLD 3](#_Toc187246382)

[3 Base de données 4](#_Toc187246383)

[3.1 Création de la base de données 4](#_Toc187246384)

[3.2 Création de données dans les tables 5](#_Toc187246385)

[**3.2.1** **Tables** 5](#_Toc187246386)

[3.3 Requêtes demandées 5](#_Toc187246387)

[3.4 Créations des utilisateurs/rôles 10](#_Toc187246388)

[3.5 Création des index 11](#_Toc187246389)

[3.6 Backup/Restore 11](#_Toc187246390)

[4 Conclusion 11](#_Toc187246391)

[4.1 Avis personnel : 11](#_Toc187246392)

[5 Annexes 11](#_Toc187246393)

[5.1 Sources : 11](#_Toc187246394)

# 

# Introduction

## Contexte :

Travailler avec une base de données sur la thématique « SpaceInvaders »

## Objectifs :

Crée et gérer une base de données en y ajoutant différents droits pour différents rôles.

# Conception

## MCD/MLD

Une image contenant texte, capture d’écran, diagramme, ligne

Description générée automatiquement

Une image contenant texte, capture d’écran, diagramme, affichage

Description générée automatiquement

# Base de données

## Création de la base de données

Pour crée la base de donnée, il faut d’abord se connecter à MySQL en utilisant cette commande : mysql -u root -proot

Ensuite, il faut exécuter cette commande afin de crée la base de données : CREATE DATABASE db\_space\_invaders ;

Après avoir exécuter ces deux commandes, ouvrez le MCD et assurer vous que le SGBD cible soit MySQL.

Une image contenant texte, logiciel, Icône d’ordinateur, Page web

Description générée automatiquement

Ensuite, copier le script SQL dans MySQL.



## Création de données dans les tables

### **Tables**

Pour crée des données pour les tables t\_joueur, t\_arme, t\_avoir, t\_commande et t\_suivre [Mockaroo](https://www.mockaroo.com) a été utilisé. Pour accéder aux fichiers d’insertions des données, rendez-vous sur Github et copier l’intérieur des fichiers sql t\_arme, t\_commande, t\_joueur, t\_suivre et t\_avoir dans mysql. C’est important de copier les fichiers dans cet ordre-là car t\_joueur, t\_suivre et t\_avoir dépendent des tables t\_arme et t\_commande. [Accéder aux fichiers d'insertions des données](https://github.com/hadri123soleil/Space_Invaders_Aymon/tree/main/Scripts/Insertions-données)

Voici des captures d’écran des informations de Mockaroo pour la création des données dans MySQL :

Une image contenant capture d’écran, texte, Police

Description générée automatiquementUne image contenant capture d’écran, texte

Description générée automatiquementUne image contenant capture d’écran, texte, logiciel

Description générée automatiquementUne image contenant capture d’écran, texte, Logiciel multimédia, logiciel

Description générée automatiquementUne image contenant texte, capture d’écran, nombre, logiciel

Description générée automatiquement

## Requêtes demandées

Requête n°1 :

La première requête que l’on vous demande de réaliser est de sélectionner les 5 joueurs qui ont le meilleur score c’est-à-dire qui ont le nombre de points le plus élevé. Les joueurs doivent être classés dans l’ordre décroissant

SELECT pseudonyme, nombrePoint FROM t\_joueur ORDER BY nombrePoint DESC LIMIT 5;

Une image contenant texte, capture d’écran, Police

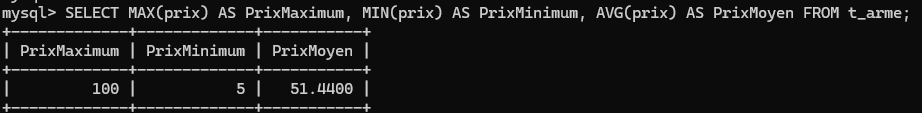
Description générée automatiquement

Requête n°2 :

Trouver le prix maximum, minimum et moyen des armes.

Les colonnes doivent avoir pour nom « PrixMaximum », « PrixMinimum » et « PrixMoyen)

SELECT MAX(prix) AS PrixMaximum, MIN(prix) AS PrixMinimum, AVG(prix) AS PrixMoyen FROM t\_arme ;



Requête n°3 :

Trouver le nombre total de commandes par joueur et trier du plus grand nombre au plus petit.

La 1ère colonne aura pour nom "IdJoueur", la 2ème colonne aura pour nom "NombreCommandes"

SELECT t\_joueur.id\_joueur AS IdJoueur, COUNT(t\_commande.id\_commande) AS NombreCommandes

FROM t\_joueur

JOIN t\_commande ON t\_joueur.fk\_commande = t\_commande.id\_commande

GROUP BY t\_joueur.id\_joueur

ORDER BY NombreCommandes DESC;

Une image contenant texte, capture d’écran

Description générée automatiquement

Requête n°4 :

Trouver les joueurs qui ont passé plus de 2 commandes.

La 1ère colonne aura pour nom "IdJoueur", la 2ème colonne aura pour nom "NombreCommandes"

SELECT t\_joueur.id\_joueur, COUNT(t\_commande.id\_commande) AS NombreCommandes

FROM t\_joueur

INNER JOIN t\_commande ON t\_joueur.fk\_commande = t\_commande.id\_commande

GROUP BY t\_joueur.id\_joueur

HAVING NombreCommandes > 2;

Une image contenant texte, capture d’écran, Police, noir

Description générée automatiquement

Le résultat est vide car aucun de mes joueurs n’a passer plus de deux commandes.

Requête n°5 :

Trouver le pseudo du joueur et le nom de l'arme pour chaque commande.

SELECT t\_joueur.pseudonyme, t\_arme.nom

FROM t\_joueur

INNER JOIN t\_commande ON t\_joueur.fk\_commande = t\_commande.id\_commande

INNER JOIN t\_suivre ON t\_commande.id\_commande = t\_suivre.fk\_commande

INNER JOIN t\_arme ON t\_suivre.fk\_arme = t\_arme.id\_arme;Une image contenant texte, capture d’écran

Description générée automatiquement

Requête n°6 :

Trouver le total dépensé par chaque joueur en ordonnant par le montant le plus élevé en premier, et limiter aux 10 premiers joueurs.

La 1ère colonne doit avoir pour nom "IdJoueur" et la 2ème colonne "TotalDepense"

SELECT t\_joueur.id\_joueur, SUM(t\_arme.prix) AS total\_prix

FROM t\_joueur

INNER JOIN t\_commande ON t\_joueur.fk\_commande = t\_commande.id\_commande

INNER JOIN t\_suivre ON t\_commande.id\_commande = t\_suivre.fk\_commande

INNER JOIN t\_arme ON t\_suivre.fk\_arme = t\_arme.id\_arme

GROUP BY t\_joueur.id\_joueur

ORDER BY total\_prix DESC

LIMIT 10;

Une image contenant texte, capture d’écran, Police

Description générée automatiquement

Requête n°7 :

Récupérez tous les joueurs et leurs commandes, même s'ils n'ont pas passé de commande.

Dans cet exemple, même si un joueur n'a jamais passé de commande, il sera quand même listé, avec des valeurs `NULL` pour les champs de la table `t\_commande`.

SELECT t\_joueur.pseudonyme, t\_commande.id\_commande

FROM t\_joueur

LEFT JOIN t\_commande ON t\_joueur.fk\_commande = t\_commande.id\_commande;

Une image contenant texte, capture d’écran

Description générée automatiquement

Requête n°8 :

Récupérer toutes les commandes et afficher le pseudo du joueur s’il existe, sinon montrer `NULL` pour le pseudo.

SELECT t\_joueur.pseudonyme, t\_commande.id\_commande

FROM t\_joueur

RIGHT JOIN t\_commande ON t\_joueur.fk\_commande = t\_commande.id\_commande;

Une image contenant texte, capture d’écran

Description générée automatiquement

Requête n°9 :

Trouver le nombre total d'armes achetées par chaque joueur (même si ce joueur n'a acheté aucune Arme).

SELECT t\_joueur.pseudonyme, COUNT(t\_suivre.fk\_arme) AS total\_armes\_achetees

FROM t\_joueur

LEFT JOIN t\_commande ON t\_joueur.fk\_commande = t\_commande.id\_commande

LEFT JOIN t\_suivre ON t\_commande.id\_commande = t\_suivre.fk\_commande

GROUP BY t\_joueur.id\_joueur, t\_joueur.pseudonyme;

Une image contenant texte, capture d’écran, Police

Description générée automatiquement

Requête n°10 :

Trouver les joueurs qui ont acheté plus de 3 types d'armes différentes

SELECT t\_joueur.id\_joueur, t\_joueur.pseudonyme, COUNT(DISTINCT t\_suivre.fk\_arme)

FROM t\_joueur

JOIN t\_commande ON t\_joueur.fk\_commande = t\_commande.id\_commande

JOIN t\_suivre ON t\_commande.id\_commande = t\_suivre.fk\_commande

GROUP BY t\_joueur.id\_joueur, t\_joueur.pseudonyme

HAVING COUNT(DISTINCT t\_suivre.fk\_arme) > 3;

Une image contenant texte, capture d’écran, Police, noir

Description générée automatiquement

Aucun de mes joueurs a plus de trois types d’armes différentes

## Créations des utilisateurs/rôles

Pour créer les utilisateurs et les rôles, j’ai utilisé les supports de cours que l’on avait à disposition.

Pour créer les utilisateurs et les rôles, veuillez copier l’intérieur de ces fichiers texte et les coller dans MySQL. [Accéder aux fichiers utilisateurs/rôles](https://github.com/hadri123soleil/Space_Invaders_Aymon/tree/main/Scripts/Créations-UtilisateursRôles)

## Création des index

En étudiant le dump MySQL **db\_space\_invaders.sql** vous constaterez que vous ne trouvez pas le mot clé **INDEX**.

1. Pourtant certains index existent déjà. Pourquoi ?

Parce que les clés primaires et les clés étrangères sont indexer pas défaut dans MySQL.

1. Quels sont les avantages et les inconvénients des index ?

Avantages : Cela accélère les recherches dans la base de données (c’est comme des raccourcis pour accéder à des données).

Désavantages : Cela prend de la place, cela ralenti les requêtes d’insertion, de modification et de suppression car dès que l’on ajoute une table, il faut remettre l’index à jour. Il est conseiller de ne pas mettre d’index lorsque ce n’est pas vraiment utile.

3. Sur quel champ (de quelle table), cela pourrait être pertinent d’ajouter un index ? Justifier votre réponse.

J’ai indexé le nom des armes sur la table t\_arme car si l’on recherche une arme par son nom, MySQL trouvera plus rapidement cette arme car il n’aura pas à parcourir toute la table t\_arme. Je trouve cependant qu’il y a très peu de données donc crée un index sur peu de données ne sert pas à grand-chose.

Commande pour indexer le nom des armes :

CREATE INDEX idx\_t\_arme\_nom ON t\_arme (nom);

## Backup/Restore

Commande pour backup :

mysqldump -u root -proot db\_space\_invaders **>** db\_space\_invaders-backup.sql

Commande pour restore:

mysql -u root -p db\_space\_invaders **<** db\_space\_invaders-backup.sql

# Conclusion

## Avis personnel :

J’ai bien aimé ce projet car comparer à d’autre projet sur les bases de données, il ne fallait pas seulement insérer des données mais également créer des rôles et leur attribuer des droits pour gérer les données. Les utilisateurs avaient donc des rôles et l’on pouvait vraiment gérer toute la base de données et voir en profondeur comment cela se passe à ce niveau-là.

# Annexes

## Sources :

Les supports de cours mis à disposition sur teams.

[Mockaroo](https://www.mockaroo.com) pour la création des données.

[Chat GPT](https://chatgpt.com) m’a aider pour les requêtes numéro 6, 9 et 10.

[Stackoverflow](https://stackoverflow.com) pour les autres requêtes