

Documentation de la base de données Gretris

La base de données Gretris est conçue pour une application de jeu. Elle suit un schéma relationnel et utilise le moteur InnoDB de MySQL pour bénéficier des fonctionnalités de gestion des transactions et des clés étrangères. Voici une description détaillée des tables créées, leurs champs, ainsi que leurs relations.

1. Base de données Gretris.

Création :

La base de données est créée avec la commande « CREATE DATABASE Gretris; ».

Elle commence par une vérification si elle existe déjà, pour éviter les doublons avec « DROP DATABASE IF EXISTS Gretris ; ».

2. Table Avatar.

Description : La table Avatar stocke les informations concernant les avatars que les joueurs peuvent choisir.

Champs :

AvatarID : Clé primaire auto-incrémentée de l'avatar.

AvatarName : Nom de l'avatar (chaîne de caractères, jusqu'à 50 caractères).

ImageUrl : URL de l'image représentant l'avatar (chaîne de caractères, jusqu'à 255 caractères).

Clé primaire : AvatarID.

Moteur : InnoDB.

3. Table Player.

Description : La table Player contient les informations des utilisateurs inscrits au jeu.

Champs :

PlayerID : Clé primaire auto-incrémentée de chaque joueur.

Username : Nom d'utilisateur (chaîne de caractères, jusqu'à 50 caractères).

Email : Adresse e-mail (chaîne de caractères, jusqu'à 100 caractères, unique).

Password : Mot de passe du joueur (chaîne de caractères, cryptée).

LoginAttempts : Nombre de tentatives de connexion échouées (par défaut à 0).

TimeOut : Indique si le compte du joueur est temporairement bloqué (valeur booléenne, par défaut à FALSE).

AvatarID : Référence à l'avatar sélectionné par le joueur (clé étrangère référencée depuis la table Avatar).

CreatedAt : Date et heure de création du compte (par défaut à la date et heure actuelle).

LastLogin : Dernière connexion du joueur (par défaut NULL).

Relations :

La clé étrangère AvatarID référence la colonne AvatarID de la table Avatar. Si un avatar est supprimé, la colonne est mise à NULL (ON DELETE SET NULL).

Clé primaire : PlayerID.

Moteur : InnoDB.

4. Table Game.

Description : Cette table enregistre les informations des parties jouées par les joueurs.

Champs :

GameID : Clé primaire auto-incrémentée de chaque partie.

PlayerID : Référence au joueur qui a lancé la partie (clé étrangère référencée depuis la table Player).

StartTime : Date et heure du début de la partie (par défaut à la date et heure actuelle).

EndTime : Heure de fin de la partie.

Relations :

La clé étrangère PlayerID référence la colonne PlayerID de la table Player. Si un joueur est supprimé, toutes ses parties associées sont également supprimées (ON DELETE CASCADE).

Clé primaire : GameID.

Moteur : InnoDB.

5. Table Score.

Description : La table Score contient les informations relatives aux performances des joueurs lors des parties.

Champs :

ScoreID : Clé primaire auto-incrémentée de chaque score.

GameID : Référence à la partie concernée (clé étrangère référencée depuis la table Game).

LinesCleared : Nombre de lignes nettoyées par le joueur durant la partie (par défaut à 0).

PiecesDropped : Nombre de pièces déposées par le joueur durant la partie (par défaut à 0).

Points : Total des points gagnés (par défaut à 0).

Level : Niveau atteint par le joueur.

Creation : Date et heure de création du score (par défaut à la date et heure actuelle).

Relations :

La clé étrangère GameID référence la colonne GameID de la table Game. Si une partie est supprimée, tous les scores associés sont également supprimés (ON DELETE CASCADE).

Clé primaire : ScoreID.

Moteur : InnoDB.

6. Relations entre les tables.

Avatar et Player :

Un joueur peut avoir un avatar, représenté par la colonne AvatarID dans la table Player. Si un avatar est supprimé, la colonne AvatarID dans la table Player est mise à NULL.

Player et Game :

Un joueur peut avoir plusieurs parties associées. Si un joueur est supprimé, toutes les parties associées sont également supprimées grâce à la contrainte ON DELETE CASCADE.

Game et Score :

Une partie peut avoir plusieurs scores associés. Si une partie est supprimée, tous les scores associés sont supprimés via la contrainte ON DELETE CASCADE.

Conclusion

La base de données Gretris est conçue pour stocker les informations sur les avatars, les joueurs, leurs parties et les scores. Elle gère les relations entre les entités et assure l'intégrité des données grâce aux clés étrangères et aux contraintes de suppression en cascade.