

TP CONCEPTION DE BASE DE DONNEES :

Conception d'une Base de Données pour un Site de Paris Sportifs de Football

Objectif du TP

Dans ce TP, vous allez concevoir et créer une base de données pour un site de paris sportifs dédié au football. Ce site permettra aux utilisateurs de parier sur des matchs de football, de gérer leur portefeuille de paris, et de consulter des informations sur les équipes et les joueurs.

Vous devrez :

1. Concevoir le schéma de la base de données.
 2. Créer les tables et leurs relations en SQL.
 3. Ajouter des données de test dans la base de données.
 4. Exécuter une série de requêtes SQL pour manipuler et interroger les données.
-

Partie 1 : Conception du Modèle de Données

Avant de commencer à écrire du SQL, vous devez concevoir le schéma de votre base de données.

1. **Modèle Conceptuel de Données (MCD)** : Utilisez un outil de modélisation (par exemple, draw.io, Lucidchart) pour créer le MCD de votre base de données. Ce modèle doit inclure les entités principales, leurs attributs, et les relations entre elles, avec leurs cardinalités.

Entités et Attributs à Modéliser

Voici les entités et attributs de base à inclure dans votre modèle. Vous pouvez ajouter d'autres attributs si vous le jugez pertinent.

1. **Utilisateurs (users)**
2. **Équipes (teams)**
3. **Joueurs (players)**
4. **Matchs (matches)**
5. **Paris (bets)**
6. **Transactions (transactions)**

Partie 2 : Création des Tables en SQL

Une fois que vous avez conçu le MCD et le MLD, vous devez écrire le SQL pour créer les tables.

- **Contrainte de clés étrangères** : Veillez à définir les clés étrangères pour lier les tables entre elles.
- **Types de données** : Choisissez des types de données appropriés pour chaque colonne.
- **Contraintes et index** : Utilisez les contraintes de clé primaire et unique là où elles sont nécessaires.

Voici un exemple de création de table pour vous guider :

```
CREATE TABLE users (  
    user_id INT PRIMARY KEY AUTO_INCREMENT,  
    username VARCHAR(50) UNIQUE NOT NULL,  
    email VARCHAR(100) UNIQUE NOT NULL,  
    password VARCHAR(255) NOT NULL,  
);
```

Répétez cette étape pour chaque table.

Partie 3 : Insertion de Données de Test

Ajoutez des données de test dans chaque table pour simuler le fonctionnement de votre base de données.

Exemples de données à insérer :

- **Utilisateurs** : 5 utilisateurs, chacun avec un solde différent.
- **Équipes** : 10 équipes avec des informations fictives.
- **Joueurs** : 20 joueurs répartis entre les équipes.
- **Matches** : 5 matches programmés entre différentes équipes.
- **Paris** : 10 paris fictifs placés par différents utilisateurs.
- **Transactions** : 5 transactions de type dépôt ou retrait.

Partie 4 : Exécution des Requêtes SQL

Après avoir conçu et créé votre base de données, vous devez exécuter une série de 20 requêtes SQL pour manipuler et interroger les données. Ces requêtes vous permettront de tester les différentes fonctionnalités de votre base de données et d'extraire des informations pertinentes pour un site de paris sportifs de football.

Pour chaque requête, vous devez :

1. Rédiger le code SQL de la requête.
2. Expliquer brièvement ce que fait la requête.
3. Présenter le résultat obtenu (les données en sortie).

Liste des Requêtes SQL

1. **Afficher les 5 derniers paris placés par un utilisateur** : Récupérer les 5 derniers paris placés par l'utilisateur avec l'identifiant `user_id = 1`, en les triant par date de pari dans l'ordre décroissant.
2. **Lister tous les joueurs, leur position et l'équipe dans laquelle ils jouent** : Afficher le nom de chaque joueur, son poste, et le nom de son équipe.
3. **Calculer le nombre total de paris placés sur chaque match** : Récupérer le nombre total de paris pour chaque match, en groupant les résultats par `match_id`.
4. **Afficher les utilisateurs qui n'ont jamais placé de pari** : Afficher les informations de tous les utilisateurs qui n'ont encore jamais placé de pari.
5. **Lister les matchs qui se sont terminés par une victoire de l'équipe A** : Afficher tous les matchs terminés où l'équipe A a gagné.
6. **Afficher les utilisateurs ayant un solde supérieur à 100€** : Récupérer les informations des utilisateurs ayant un solde de plus de 100€.
7. **Afficher le total des paris gagnés et perdus par un utilisateur spécifique** : Calculer le nombre total de paris gagnés et perdus par l'utilisateur avec l'identifiant `user_id = 1`.
8. **Lister les équipes avec le nombre de victoires qu'elles ont eues** : Afficher le nombre de victoires pour chaque équipe, en comptant les matchs où l'équipe est listée comme gagnante.
9. **Lister les matchs sur lesquels un utilisateur a parié et le statut de chaque pari** : Afficher tous les matchs sur lesquels l'utilisateur avec l'identifiant `user_id = 1` a parié, ainsi que le montant du pari, la prédiction, et le statut de chaque pari.
10. **Afficher le montant total des transactions (dépôts et retraits) pour chaque utilisateur** : Calculer le montant total des dépôts et des retraits effectués par chaque utilisateur, en groupant les résultats par `user_id`.

11. **Ajouter un dépôt pour un utilisateur** : Ajouter une transaction de type "dépôt" de 150€ pour l'utilisateur avec l'identifiant `user_id = 1`.
 12. **Placer un pari sur un match pour une équipe** : Insérer un pari de 50€ sur l'équipe A pour le match ayant `match_id = 2` pour l'utilisateur avec l'identifiant `user_id = 1`.
 13. **Afficher tous les matchs programmés d'une équipe** : Afficher tous les matchs programmés pour une équipe spécifique (par exemple, l'équipe ayant `team_id = 1`).
 14. **Afficher les paris en attente pour un utilisateur** : Afficher tous les paris qui sont en statut "en attente" pour l'utilisateur avec l'identifiant `user_id = 1`.
 15. **Mettre à jour le statut d'un pari en "won" avec le montant gagné** : Mettre à jour le statut d'un pari pour indiquer qu'il a été gagné et calculer le montant gagné en multipliant le montant parié par une cote de 2,5.
 16. **Afficher les joueurs d'une équipe spécifique** : Afficher tous les joueurs appartenant à une équipe spécifique (par exemple, l'équipe ayant `team_id = 1`).
 17. **Lister les équipes avec leur nombre de joueurs** : Afficher le nombre de joueurs dans chaque équipe.
 18. **Calculer le total des paris placés par les utilisateurs pour chaque match** : Calculer la somme totale des montants pariés sur chaque match, en groupant les résultats par `match_id`.
 19. **Lister les utilisateurs ayant gagné un pari récemment** : Afficher les utilisateurs ayant gagné au moins un pari dans les 30 derniers jours.
 20. **Afficher le solde des utilisateurs et le total de leurs gains** : Récupérer le solde de chaque utilisateur ainsi que le montant total de tous les gains accumulés grâce aux paris gagnés.
-

Partie 5 : Présentation du Travail

1. **Diagrammes MCD et MLD** : Présentez vos diagrammes de Modèle Conceptuel et Logique.
2. **Code SQL de Création des Tables** : Fournissez le code SQL pour créer chaque table.
3. **Données de Test** : Présentez les données de test que vous avez insérées dans chaque table.
4. **Requêtes SQL** : Fournissez les requêtes SQL demandées et les résultats obtenus.
5. **Documentation** : Ajoutez une courte description expliquant chaque requête et son but.

OBJECTIFS BONUS :

Objectif Bonus 1 : Ajout d'une Table "Leagues" et d'une Table "Standings"

Description : Créez une table leagues pour stocker des informations sur les ligues de football, ainsi qu'une table standings pour enregistrer les positions des équipes dans chaque ligue. Cela permettra de suivre le classement des équipes dans des tournois.

Exemple de Requêtes SQL

1. **Afficher les équipes et leurs positions dans une ligue spécifique**
2. **Calculer le nombre total de points pour une équipe dans une ligue**

Objectif Bonus 2 : Suivi des Performances des Joueurs dans les Matches

Description : Créez une table `player_stats` pour enregistrer les statistiques des joueurs pour chaque match. Cela permettra de suivre les performances individuelles des joueurs.

Requêtes SQL pour Analyser les Performances des Joueurs

1. **Lister les joueurs ayant marqué au moins un but dans un match spécifique**
2. **Calculer le nombre total de buts marqués par un joueur au cours de la saison**
3. **Lister les joueurs ayant reçu un carton rouge dans un match spécifique**
4. **Calculer le nombre total de cartons jaunes reçus par un joueur dans la saison**
5. **Lister les joueurs ayant réalisé au moins une passe décisive dans un match spécifique**
6. **Afficher les 5 meilleurs buteurs de la saison**
7. **Calculer le total des buts, passes décisives, cartons jaunes et rouges pour chaque joueur**
8. **Lister les joueurs avec le plus de cartons rouges dans la saison**
9. **Afficher les performances d'un joueur spécifique dans un match donné**
10. **Calculer le nombre moyen de buts par match pour un joueur dans la saison**