

TD5: BDD.

Une plateforme propose plusieurs jeux payants. Le prix d'une partie est exprimé par un nombre entier, « nombre de crédits ». Par un moyen de paiement qui n'a pas d'importance ici un joueur peut recharger son nombre de crédits. Sa « cagnotte » désigne le nombre de crédits dont il dispose. Chaque partie jouée rapporte un certain nombre de points au joueur (ne pas confondre points et crédits). La base de données est organisée comme suit

1. une table des utilisateurs, **users** :
 - **pseudo** : pseudo (chaîne)
 - **nom** : nom complet (chaîne)
 - **inscrit** : date d'inscription (date)
 - **cagnotte** : crédits disponibles (entier)
2. une table **jeux** recense tous les jeux disponibles sur la plateforme :

- **id** : identifiant du jeu (chaîne)
- **nom** : nom du jeu (chaîne)
- **prix** : prix d'une partie (entier)

3. une table des **parties** où figurent toutes les parties jouées :

- **joueur** : pseudo d'un utilisateur
- **jeu** : identifiant d'un jeu
- **quand** : date de la partie
- **points** : points obtenus (entier)

Quelques fonctions SQL :

- extraction d'une partie d'une date :
date_part (*part*, *date*) où *part* est une chaîne telle que 'month', 'day', 'year'
- moyenne : **avg()** (fonction de regroupement)

Exercice 1 :

Pour cet exercice nous reprenons la base utilisée dans les td3 et td4 (« jeux »), en tenant compte des informations supplémentaires qui suivent.

Chaque joueur possède un pseudo qui lui est propre (2 joueurs différents ont des pseudo nécessairement différents). Sa date d'adhésion, son nom et sa cagnotte sont nécessairement définies. Toute donnée représentant un montant en euro sera représentée par un nombre décimal positif ou nul, à 2 chiffres après la virgule (et non pas un entier comme indiqué dans le TD3) et doit nécessairement être définie. Le nombre de points gagnés lors d'une partie est également nécessairement défini et positif ou nul.

Q 1 . Définition de la table **users**

- quel attribut pourrait-il servir de clé primaire ?
- quelles contraintes serait-il approprié d'associer à la définition de table
- écrivez la commande SQL de création de la table

Clé primaire: pseudo

contraintes not null unique sur pseudo

create table users (

 pseudo VARCHAR(20) PRIMARY KEY ,

 char [40] nom

 datetime inscrit

 int cagnotte

)

Q 4 . La base est supposée, à un moment donné, contenir exactement (tous les tuples sont représentés) :

users		
pseudo	nom	...
XX	xxx	...
YY	yyy	...

jeux		
id	nom	...
27B-6	jeu1	...
42Z-5	jeu2	...

parties		
joueur	jeu	...
YY	27B-6	...
ZZ	27B-6	...
YY	42Z-5	...
YY	42Z-5	...
ZZ	45T-9	...

1. ces données présentent des incohérences. Lesquelles ?
2. comment les éviter et assurer que la base ne se trouve dans cet état incohérent ?

1. Le jeu 45T-9 n'existe pas dans jeux, alors qu'il est listé dans parties.
2. Il faut que parties.jeu soit une clé étrangère.
FOREIGN KEY <nom> references jeux(id).

Exercice 2 :

Q 1 . Un parc informatique est réparti dans plusieurs salles. Chaque salle possède un identifiant que l'on supposera définitif et permanent, un nom , une capacité d'accueil (en nombre de personnes). Quels sont les attributs à définir pour une table **salles**? (indiquez la liste des attributs –nom et type–, il n'est pas demandé de le faire en SQL). Quelle est la clé primaire ?

VARCHAR(30) id
int capacité
char[60] nom

Q 2 . Dans une salle se trouvent plusieurs postes de travail (le nombre de postes de travail varie selon les salles et peut aussi varier dans le temps). Chaque poste est identifié par un numéro (permanent) au sein de la salle dans laquelle il se trouve.

Quels sont les attributs à définir pour une table **postes**? (indiquez la liste des attributs –nom et type–, il n'est pas demandé de le faire en SQL). Quelle est la clé primaire ?

type-, il n'est pas demandé de le faire en SQL). Quelle est la clé primaire ?

id poste int
varchar(30)