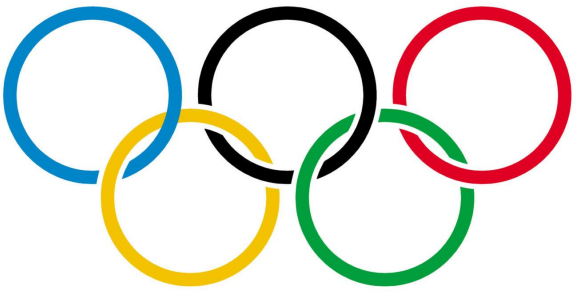
RAPPORT SAE 2.04

Exploitation de base de données



DELESPIERRE Jules - DHORME Victor | Groupe C

Partie 1 : Compréhension du sujet :

Tout d'abord dans notre SAE, nous avions eu à comprendre les fichiers CSV ; grâce à la commande *wc* suivie de l'option *-l* :

Nous avons pu savoir que le fichier athletes_events.csv comporte 271 117 lignes et que le fichier noc_regions en comporte 231.

La première ligne du fichier CSV athletes_events est : "ID","Name","Sex","Age","Height","Weight","Team","NOC","Games","Year","Season","City","Sport","Event","Medal" ; la commande que nous avons utilisé est *head -n 1* athlete_events.csv ; étant donné que c'est un fichier CSV, cette première ligne montre les différentes colonnes du fichier.

D'après le résultat de la commande d'au-dessus, on peut affirmer que le séparateur sont les virgules.

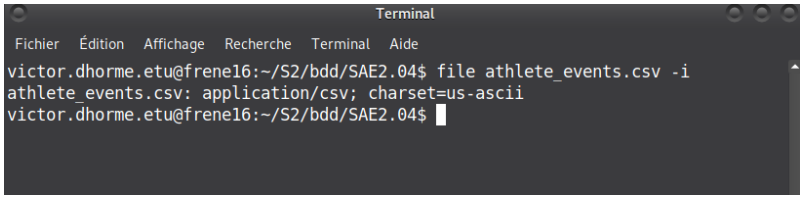
Si nous regardons les 5 premières lignes, grâce à la commande *head -n 5 athlete_events.csv*, on s'aperçoit que chaque ligne représente une donnée.+

Dans le fichier, il y a 14 colonnes, nous avons utilisé la commande "*head -n 1 athlete_events.csv | grep ,*"

La colonne "Season", différencie les jeux d'été des jeux d'hiver

Il y a 6 lignes qui font référence à Jean-Claude Killy dans le fichier athlete_events.csv , nous avons trouvé ce chiffre grace a la commande : *cat athlete_events.csv | grep "Jean-Claude Killy" -c.*

Pour trouver le type d'encodage du fichier , il faut exécuter la commande "*file athlete_events.csv -i".*



Le type est donc "us-ASCII".

Pour importer ces données, nous envisageons de créer une table SQL import et ensuite de charger le fichier dedans.

Partie 2 : Importer les données :

Pour créer une table import, nous avons utilisé la commande CREATE TABLE en SQL, avec comme colonnes toutes les colonnes présentes dans notre fichier CSV avec les types de données associées. Et ensuite, pour importer les données nous avons dû utiliser la commande copy qui permet d'entrer tout le fichier CSV dans une table.

Cette table import nous permet de stocker les données en brut avant de les ventiler.

```
CREATE TABLE import(ID int, Name text, Sex char(1), Age int, Height int, Weight float, Team text, NOC text, Games text, Year int, Season text, City text, Sport text, Event text, Medal text);
\copy import from athlete_events.csv delimiter ',' CSV HEADER NULL 'NA'
```

Ensuite, pour le besoin du projet, nous allons retirer quelques données, tels que les épreuves artistiques, et les épreuves d'avant 1920 ; pour cela nous allons utiliser la requete DELETE, qui permet de supprimer des données avec des conditions.

```
DELETE FROM import WHERE Year<1920 OR Sport='Art Competitions';
```

Cela nous permet d'avoir après cette suppression, 255 080 lignes dans notre table import.

Ensuite, nous allons importer les données représentatives des pays, c'est-à-dire le NOC du Pays ainsi que le nom de celui-ci. Le NOC sont les initiales de National Olympic Committee, c'est une combinaison de 3 lettres pour chaque état qui est enregistré auprès du Comité International Olympique.
Pour importer ces données, il faut tout d'abord créer une table qui pourra accueillir nos nouvelles données :
Nous avons utilisé une colonne NOC qui peut comporter 3 caractères, ainsi que les noms des pays dans la colonne region, puis enfin la dernière colonne du CSV comporte des notes textuelles sur le pays.

```
CREATE TABLE regions(NOC char(3), region text, notes text);
\copy regions from noc_regions.csv delimiter ',' CSV HEADER;
```

Pour que le script SQL puisse être utilisé autant de fois que possible, il faut faire en sorte que les tables que l'on va créer n'existent pas encore dans la base de données, si nous essayons de créer des tables avec des noms déjà attribués, le script ne les créera pas.
Donc pour éviter ce problème, nous allons rajouter les requêtes qui suppriment les tables, avant que l'on demande de les créer !
Pour cela nous allons ajouter au début du fichier :

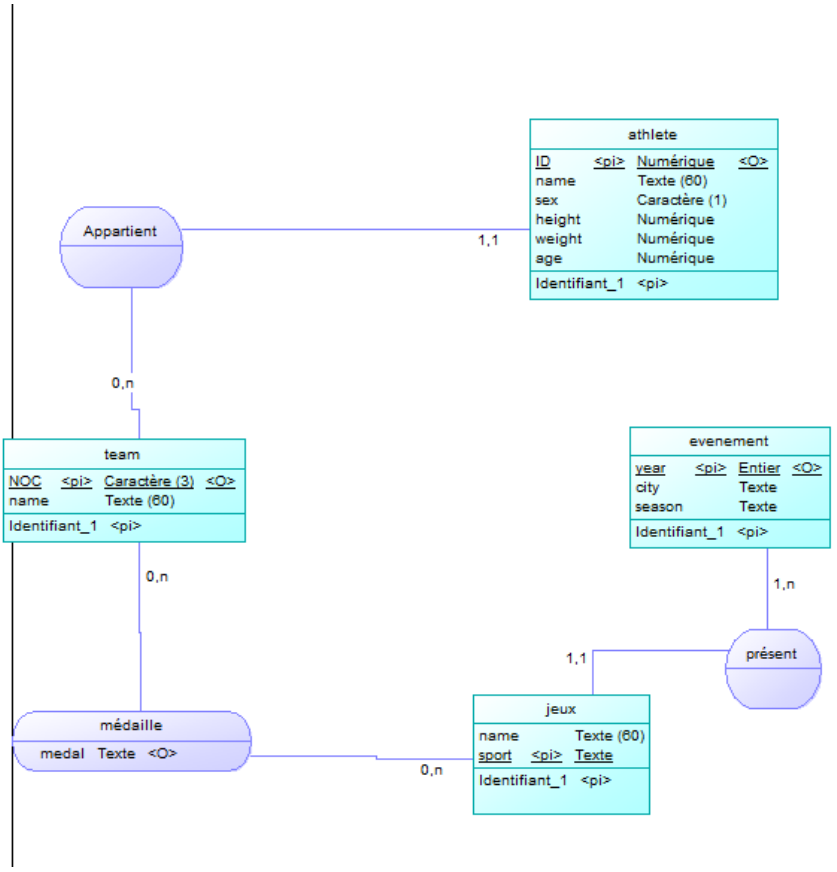
```
DROP TABLE import;
DROP TABLE regions;
```

Partie 4 : Ventiler les données :

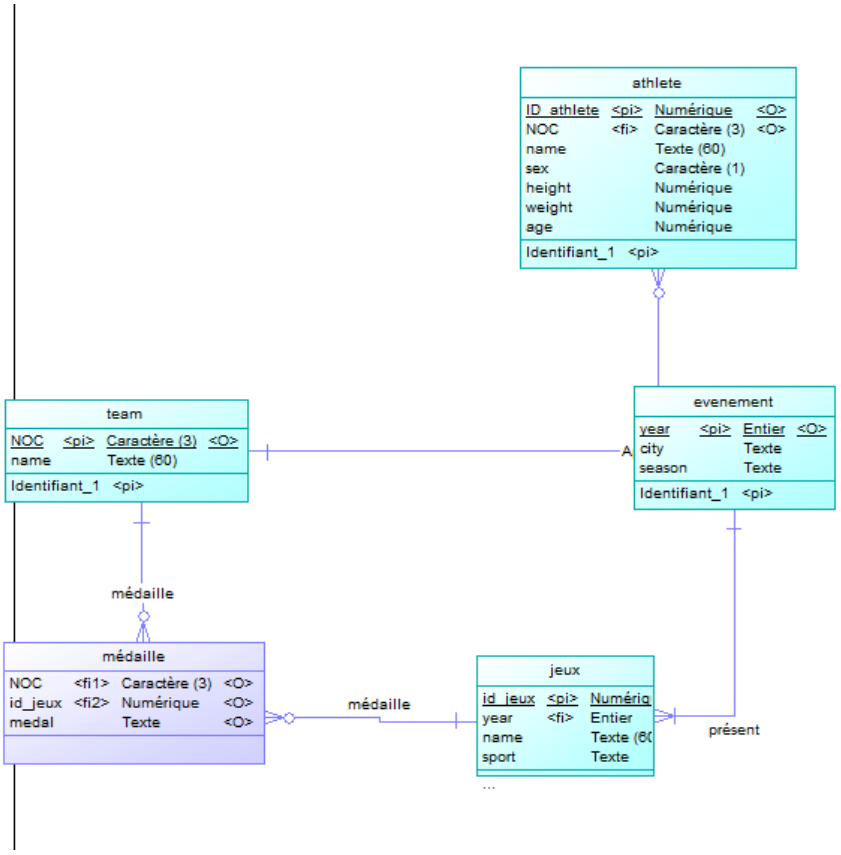
Pour ventiler nos données nous avons d'abord réfléchi sur les différentes données qui étaient présentes dans la table import, au début nous avions voulu ventiler les données de la sorte qu'on arrive avec une table athlète avec son identifiant, nom, age, taille, poids, nationalité, puis ensuite avoir une table pays avec les noms des pays en lien avec le NOC, et une table jeux qui contient toutes les années où il y a eu des jeux avec leurs villes d'organisation, la table épreuves qui contient toutes les épreuves avec le sport et le nom de l'épreuve, ainsi qu'un identifiant pour que cela soit plus simple dans les associations. Toutes ces tables sont reliées par l'association participe qui contient le numéro d'athlète ainsi que le numéro de l'épreuve et l'année des jeux.

Les problèmes de ce schéma là sont qu'il y a eu de 1924, lors de la création des jeux olympiques d'hiver, jusqu'en 1992, l'organisation des jeux olympiques d'hiver la même année que les jeux d'été donc dans un premier temps, la clé primaire sur l'année n'était pas compatible car certaines valeurs différentes avaient la même clé primaire. Ensuite, nous nous sommes rendus compte que la table athlète ne pouvait pas comporter le NOC, ni l'âge, car en effet, l'âge change en fonction des participations et que certains athlètes ont participé à différents jeux sous différents drapeaux. Pour régler ces problèmes, nous avons décidé d'inclure l'âge de l'athlète ainsi que le NOC dans la table participation, et de créer un "id_jeux" qui est une valeur numérique qui s'incrémente pour la table jeux. Nous avons donc conclus que le MCD ainsi que le MLD le plus efficace était le suivant :

MCD correspondant :



MLD correspondant :



Une fois que nos tables ont été créées et lorsqu'on les a rempli avec les données, nous avons regardé le poids des tables. Tout d'abord le fichier athletes_events.csv environ 40 Mo et le fichier noc_regions.csv fait lui environ 4Ko.

Alors que la table import fait elle : 47259648 octets, et la somme de toutes les tables creer font : 22224896 octets .

La somme des poids des tables exportés : 10429034 octets

On peut donc constater qu'une fois les données ordonnées dans une BDD , les données prennent énormément moins de place car les redondances sont évitées au maximum et les données sont plus faciles d'accès.

Partie 6 : Personnalisation du rapport.

nom	nom	year	name
Venezuela	Table Tennis Women's Singles	2016	Gremlis Andrena Arvelo Garca
Venezuela	Table Tennis Men's Singles	1988	Francisco Lpez
Venezuela	Table Tennis Women's Doubles	2000	Luisana Prez
Venezuela	Table Tennis Women's Doubles	2004	Luisana Prez
Venezuela	Table Tennis Women's Singles	1988	Elizabeth Popper
Venezuela	Table Tennis Women's Singles	1996	Fabiola Isabel Ramos Portillo
Venezuela	Table Tennis Women's Doubles	2000	Fabiola Isabel Ramos Portillo
Venezuela	Table Tennis Women's Singles	2000	Fabiola Isabel Ramos Portillo
Venezuela	Table Tennis Women's Doubles	2004	Fabiola Isabel Ramos Portillo
Venezuela	Table Tennis Women's Singles	2004	Fabiola Isabel Ramos Portillo
Venezuela	Table Tennis Women's Singles	2008	Fabiola Isabel Ramos Portillo
Venezuela	Table Tennis Women's Singles	2012	Fabiola Isabel Ramos Portillo

Pour essayer de notre base de données, nous avons choisi de prendre comme pays le Venezuela et comme sport le Tennis de Table. Tout d'abord, nous avons voulu voir toutes les participations de ce pays dans ce sport.

name
Elizabeth Popper
Fabiola Isabel Ramos Portillo
Francisco Lpez
Gremlis Andrena Arvelo Garca
Luisana Prez

Cela nous montre toutes les participations. Il y a 12 participations au total
Puis ensuite, nous avons voulu avoir la liste des athlètes qui ont participé :

Seulement 5 athlètes ont représenté le Venezuela au tennis de table sur les 12 participations. De plus, nous nous sommes demandé où se situait les jeux olympiques lors de ces participations :

name	year	city
Elizabeth Popper	1988	Seoul
Francisco Lpez	1988	Seoul
Fabiola Isabel Ramos Portillo	1996	Atlanta
Fabiola Isabel Ramos Portillo	2000	Sydney
Luisana Prez	2000	Sydney
Fabiola Isabel Ramos Portillo	2004	Athina
Luisana Prez	2004	Athina
Fabiola Isabel Ramos Portillo	2008	Beijing
Fabiola Isabel Ramos Portillo	2012	London
Gremlis Andrena Arvelo Garca	2016	Rio de Janeiro

Et enfin, nous avons regardé si leurs participations avaient ramené une médaille :

name	year	city	medal

Malheureusement dans ses 12 participations les athlètes ne sont jamais parvenus à remporter une médaille dans ce sport.