

## TP : SQL, fonctions d'agrégation

### 1 Données météo

On utilisera la base de donnée `meteo.sqlite` constituée d'une seule table. Cette base de donnée ne contient qu'une seule table `Relevés` sur laquelle on peut se poser plusieurs questions statistiques.

1. Afficher les relevés météo effectués entre le 1er avril 2010 et le 25 avril 2010. Attention les dates sont au format "AAAA-MM-JJ" (avec les guillemets). On remarquera que SQL comprend comment comparer ses valeurs même si elles sont enregistrées au format texte<sup>1</sup>.
2. Calculer la température moyenne sur l'ensemble des relevés.
3. Calculer la température moyenne du mois d'octobre 2014.
4. Afficher la température moyenne selon le type de temps (brumeux, soleil, etc). On présentera les résultats sous la forme d'un tableau à deux colonnes "Temps" et "Température moyenne".
5. De même afficher la température moyenne et la pression moyenne selon le type de temps.
6. Afficher les types de temps pour lesquels la pression moyenne est inférieure à 1010 hPa.
7. Afficher la pression moyenne pour chaque type de temps, calculée seulement sur les jours où la pression est inférieure à 1010 hPa.
8. Afficher le jour, la date, l'heure et la valeur du record de température maximale.
9. Afficher de même le record de température minimale pour l'année 2013.
10. Afficher la vitesse moyenne du vent selon les directions du vent. On présentera ces résultats de façon ordonnée à l'aide de la commande `ORDER BY colonne`
11. À l'aide d'une commande SQL indiquer le nombre de le relevés effectués.
12. Compter le nombre de types de temps différents.
13. Pour chacun de ces types de temps, donner le nombre de relevés effectués.
14. Classer les types de temps du plus fréquent au moins fréquent. On pourra utiliser `ORDER BY colonne DESC` pour classer de façon décroissante.
15. Classer les jours de la semaine selon leur température moyenne, du plus chaud au plus froid, pour l'année 2010.

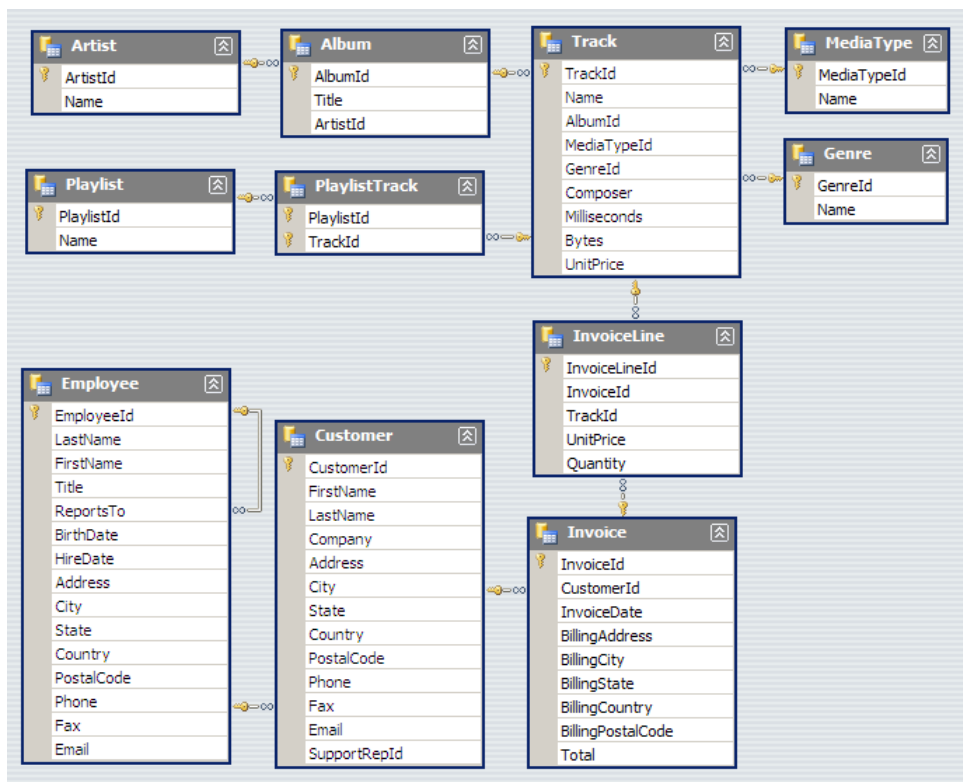
---

1. car l'ordre lexicographique pour le format AAAA-MM-JJ est compatible avec l'ordre chronologique

## 2 Plate-forme musicale Chinook

On utilisera la base de donnée fictive `chinook.sqlite` qui représente les données d'un site internet de vente de musique en ligne. On y gère les morceaux musicaux, les albums, les playlists, mais également les clients (*customer*) et les factures (*invoice*).

Voici un schéma de cette base de donnée.



Dans les questions qui suivent on prendra garde aux jointures naturelles : par exemple les attributs "Name" existent dans les tables "Genre", "Artist" et "Track" (et d'autres encore) mais ne signifient pas la même chose.

1. Afficher les titres, les compositeurs et les tailles de fichier de l'album "Mezmerize"
2. Afficher un tableau comptant le nombre de morceaux disponibles dans chaque genre.  
On pourra trier les résultats.
3. Afficher les titres concernés par la facture numéro 40.
4. Afficher les titres achetés par Marc Dubois et leur prix unitaire
5. En déduire le total dépensé par Marc Dubois sur ce site
6. Afficher une liste des prénoms et noms des clients du site ainsi que le total dépensé par chacun d'eux. Qui est le plus gros client ?
7. Afficher les titres de la *playlist* "Classical"