

1 Requete SQL

1.1 Echauffement

Pour s'échauffer, voici quelques exemples de requêtes (ici interrogations).
A l'aide de la base de donnée "Tour de France". Répondre aux requêtes ci-dessous :

1. Quel est le nom de l'équipe de code "GFC" ?
2. Combien y a-t-il d'étapes ?
3. Quelles sont les étapes de plus de 190 km ?
4. Quelles sont les équipes dont le nom comporte la chaîne "Team" ?
5. Donner la liste des coureurs de l'équipe Deceuninck.
6. Quels temps a mis Alaphilippe lors de l'étape Sisteron-Orcière ?
7. Donner le dossard des coureurs et le temps mis lors de l'étape partant de Gap.

1.2 Requête interrogative

- Le résultat d'une requête interrogative sera toujours une table/une relation.
- Une requête SQL se termine toujours par ; (un point virgule)
- Une requête interrogative (simple) est de la forme :

```
SELECT <liste d'attribut(s)> #colonnes que l'on veut garder
FROM <liste de table(s)> ; #à partir de quelles tables
```

1. Quels sont les noms et prénoms des coureurs ?

```
SELECT Coureurs.nomCoureur,Coureurs.prénomCoureur
FROM Coureurs;
```

Lorsqu'il n'y a pas d'ambiguïté pour la désignation d'un attribut il est possible d'omettre le nom de la table dans sa désignation :

```
SELECT nomCoureur,prénomCoureur
FROM Coureurs;
```

Pour tout garder tous les attributs :

```
SELECT * FROM Coureurs;
```

2. *C'est à vous* : Donner la liste des noms des équipes ?

...

3. *C'est à vous* : Quelles sont les villes de départ ?

...

On évite les doublons :

```
SELECT DISTINCT VilleDépart FROM Etapes;
```

4. Donner la liste des étapes dans l'ordre croissant des noms de ville de départ..

```
SELECT * FROM Etapes
ORDER BY VilleDépart ASC;
```

5. *C'est à vous* : Donner la liste des coureurs triés dans l'ordre croissant de leur Nom.

```
...
...
```

ASC (dans l'ordre), DESC (ordre inverse)

6. Quel est l'étape la plus petite ?

```
SELECT *, MIN(km) FROM Etapes;
```

7. Quel est le nombre de participants ?

```
SELECT COUNT(*) FROM Coureurs;
```

8. Combien y a t-il de ville de départ ? (Réponse 3)

```
SELECT COUNT(DISTINCT VilleDépart) FROM Etapes;
```

On pourra utiliser les fonctions suivantes :

- **AVG()** : moyenne des valeurs
- **SUM()** : somme des valeurs
- **MIN()** : valeur minimum, **MAX()** : valeur maximum
- **COUNT(DISTINCT()**

9. *C'est à vous* : Quelles est la distance moyenne des étapes du tour ?

```
...
```

10. *C'est à vous* : Quelles est la distance totale des étapes du tour en m ?

```
...
```

Possibilité de ne garder que certaines lignes en ajoutant de la clause optionnelle **WHERE**.

```
SELECT ... FROM ...
WHERE <condition> ;
```

<condition> est une condition portant sur des attributs et/ou valeurs et peut utiliser différents opérateurs :

- opérateurs de comparaison <, >, <=, >=, =, <> équivaut à !=
- opérateurs booléens AND, OR, NOT
- prédicat d'appartenance à un intervalle : BETWEEN ... AND ...
- prédicat de recherche : LIKE

Si un tuple donné satisfait la condition alors il est gardé sinon il est écarté.

11. Donner le nom et prénom des coureurs de codeEquipe "COF".

```
SELECT nomCoureur, prénomCoureur FROM Coureurs
WHERE codeEquipe="COF";
```

12. Donner les étapes dont le kilométrage est compris en 160 et 190km.

```
SELECT * FROM Etapes
WHERE km>=160 AND km<=190;
```

On peut aussi utiliser :

```
SELECT * FROM Etapes
WHERE km BETWEEN 160 AND 190;
```

13. Donner les coureurs dont le nom commence par un C.

```
SELECT * FROM Coureurs
WHERE nomCoureur LIKE 'c%';
```

14. Donner les coureurs dont le nom fini par un E

```
SELECT * FROM Coureurs
WHERE nomCoureur LIKE '%E';
```

15. Donner les équipes dont le nom contient "team".

```
SELECT * FROM Equipes
WHERE nomEquipe LIKE '%team%';
```

16. *C'est à vous* : Donner les coureurs dont le nom ET le prénom contiennent un a. (On ordonnera les résultats par ordre croissant des noms)

```
...
...
...

```

17. Observer le résultat de :

```
SELECT * FROM Coureurs,Equipes;
```

Vocabulaire : On dit que l'on fait le **produit cartésien** de deux relations.

18. *C'est à vous* :

- Combien y a-t-il d'enregistrements dans la requête précédente ?
 - Combien y a-t-il d'enregistrements dans la relation Coureurs ?
 - Combien y a-t-il d'enregistrements dans la relation Équipes ?
- On va supprimer certaines lignes :

```
SELECT * FROM Coureurs,Equipes
WHERE Coureurs.codeEquipe=Equipes.codeEquipe;
```

On va supprimer certaines colonnes :

```
SELECT nomCoureur, prénomCoureur, nomEquipe      #Attributs à afficher
FROM Coureurs, Equipes                          #table à utiliser
WHERE Coureurs.codeEquipe=Equipes.codeEquipe;    #Jointure
```

Remarque : la jointure fait le lien entre la clef primaire et la clé étrangère.

On va supprimer certaines colonnes :

```
SELECT nomCoureur, prénomCoureur, nomEquipe      #Attributs à afficher
FROM Coureurs, Equipes                          #table à utiliser
WHERE Coureurs.codeEquipe=Equipes.codeEquipe;    #Jointure
AND Equipes.codeEquipe='COF'                    #contrainte
```

On peut aussi écrire :

```
SELECT nomCoureur, prénomCoureur, nomEquipe      #Attributs à afficher
FROM Equipes                                    #table à utiliser
JOIN Coureurs ON Coureurs.codeEquipe=Equipes.codeEquipe #Jointure
WHERE Equipes.codeEquipe='COF';                 #contrainte
```

19. *C'est à vous* : Qu'est ce que la requête précédente a permis d'afficher ?

...

20. Donner le dossard des coureurs et le temps mis lors de l'étape 4.

```
SELECT Temps.dossard, nomCoureur, tempsRéalisé, numéroEtape
FROM Temps, Coureurs
WHERE Temps.dossard=Coureurs.dossard AND numéroEtape=4;
```

```
SELECT * FROM Coureurs
JOIN Temps ON Temps.dossard=Coureurs.dossard
WHERE Temps.numéroEtape=4;
```

21. Donner le dossard des coureurs et le temps mis lors de l'étape partant de Gap.

```
SELECT Temps.dossard,nomCoureur, tempsRéalisé, Temps.numéroEtape
FROM Temps,Coueurs,Etapes
WHERE Temps.dossard=Coueurs.dossard
AND Temps.numéroEtape=Etapes.numéroEtape
AND Etapes.villeDépart='Gap'
```

```
SELECT Temps.dossard,nomCoureur, tempsRéalisé, Temps.numéroEtape
FROM Temps
JOIN Coueurs ON Temps.dossard=Coueurs.dossard
JOIN Etapes ON Temps.numéroEtape=Etapes.numéroEtape
WHERE Etapes.villeDépart='Gap';
```

22. *C'est à vous* : Donner la liste des coueurs de l'équipe Deceuninck.

...

...

23. *C'est à vous* : Quels temps à mis Alaphilippe lors de l'étape Sisteron-Orcière ?

...

...

Exercice n° 1

Téléchargez une base de données relative aux communes de France (communes.sqlite) et ouvrez-la dans DB Browser (la population est en milliers d'habitants et la superficie en hectares). Écrivez et testez les commandes SQL permettant d'afficher :

1. toutes les communes par ordre alphabétique (36613 réponses) ;
2. toutes les communes du PAS-DE-CALAIS (895 réponses) ;
3. toutes les communes du PAS-DE-CALAIS de superficie supérieure à 1500 ha (62 réponses) ;
4. la superficie de la ville de WINGLES (1 réponse) ;
5. tous les noms des communes du PAS-DE-CALAIS et leur superficie, de la plus peuplée à la moins peuplée (895 réponses) ;
6. la plus grande commune de France en superficie (1 réponse) ;
7. Le département et la région où se trouve la ville de ARLES (1 réponse) ;
8. la population moyenne des communes du Pas-de-Calais (1 réponse) ;
9. la densité moyenne des communes de France, en habitants par km^2 (1 réponse) ;
10. le nombre de sous-préfectures de Lorraine de moins de 3000 ha (8 réponses) ;
11. la superficie totale de toutes les sous-préfectures et les chef-lieu de cantons de population entre 100 et 1000 habitants. (Attention l'attribut population est en milliers d'hbts) (réponse :1396559 ha) ;

(source de la base de données des communes : Insee)