

Bases de données  
TP 1



# Sommaire

0. Introduction.....	3
1. Découverte de SQLite.....	3
2. Requêtes SQL.....	4
3. Query plan.....	5
4. Conclusion.....	6

# 0. Introduction

Dans ce TP nous allons nous familiariser avec l'outil SQLite qui nous permettra de créer et de gérer des bases de données.

## 1. Découverte de SQLite

Création des trois tables étudiant, professeur et enseignement :

```
CREATE TABLE etudiant(etudId INTEGER, nom TEXT, prenom TEXT, adress
TEXT);
CREATE TABLE professeur(profId INTEGER, nom TEXT, prenom TEXT);
CREATE TABLE enseignement(ensId INTEGER, sujet TEXT, etudId INTEGER,
profId INTEGER);
```

Insertion des enregistrements :

```
INSERT INTO etudiant VALUES (0,'Baron','Nicolas','Rennes');
INSERT INTO etudiant VALUES (1,'Chauvel','Tom','Rennes');

INSERT INTO professeur VALUES (0,'Lamarche','Fabrice');
INSERT INTO professeur VALUES (1,'Feuillatre','Hélène');

INSERT INTO enseignement VALUES (0,'IHM',1,1);
INSERT INTO enseignement VALUES (1,'CPP',0,0);
INSERT INTO enseignement VALUES (1,'CPP',1,0);
```

Formulation et exécution des requêtes SQL :

```
SELECT * from etudiant JOIN enseignement USING(etudId) WHERE
sujet='IHM';
SELECT nom,prenom from etudiant JOIN enseignement USING(etudId) WHERE
sujet='CPP';
```

## 2.Requêtes SQL

Dans le TD1, nous devons réaliser les requêtes suivantes sur la base de données épiceries :

1. Trouver les noms des produits originaires de Dijon.

```
SELECT nom_p FROM Produit WHERE origine='Dijon';
```

```
cassis  
moutarde  
cornichon
```

2. Trouver les noms et numéros des fournisseurs de salade.

```
SELECT nom_f,FE.f from Fourniture AS FE , Produit AS P, Fournisseur AS  
FR where FE.f = FR.f AND FE.p = P.p AND nom_p = 'salade';
```

```
Bornibus|f1  
Bossuet|f4
```

3. Trouver les numéros des produits qui sont fournis par un fournisseur Parisien.

```
SELECT p from Fourniture join Fournisseur using(f) where ville =  
'Paris';
```

```
p1  
p4  
p5  
p6  
p2  
p4
```

4. Trouver les noms et les remises consenties par des fournisseurs qui fournissent au moins un des produits de Dijon.

```
SELECT DISTINCT nom_f, remise from Produit JOIN Fourniture using(p)  
JOIN Fournisseur using(f) where origine='Dijon';
```

```
Bornibus|5  
Mercier|7  
Colbert|3  
Bossuet|6
```

5. Trouver les noms des produits que fournit Bornibus en quantité inférieure à 5.

```
SELECT nom_p from Fourniture join Produit using(p) join Fournisseur  
using(f) where nom_f='Bornibus' AND qte < 5;
```

```
cassis  
moutarde  
cornichon
```

6. Trouver les noms et villes des fournisseurs qui consentent une remise supérieure à celle accordée par Bornibus.

```
SELECT F2.nom_f , F2.ville from Fournisseur F1, Fournisseur as F2 WHERE  
F1.remise < F2.remise and F1.nom_f = 'Bornibus';
```

```
Mercier|Paris  
Bossuet|Dijon  
Tanguy|Riec
```

### 3. Query plan

Affiche le nom des fournisseurs et le nom des produits qu'ils vendent.

Scanne la table fs, recherche dans la table ft pour matcher tous les fournisseurs puis recherche dans la table p pour matcher les produits.

```
|--SCAN fs  
|--SEARCH ft USING AUTOMATIC COVERING INDEX (f=?)  
`--SEARCH p USING AUTOMATIC COVERING INDEX (p=?)
```

## 4. Conclusion

Grâce à ce TP, nous avons pu utiliser l'outil SQL et nous pouvons désormais créer et interroger des bases de données.