
	BTS IG 1 <sup>ère</sup> année ALSI	Evaluation no 4 Base de données et SQL	
	Nom :		Page 1 / 7

## 1 Introduction

Cette évaluation est inspirée de l'étude de cas Anabio (métropole 2003)

### 1.1 *Documents autorisés :*

- Lexique SQL sans commentaire ni exemple d'utilisation des instructions.
- Calculatrices non autorisées

## 2 Contexte

La société ANABIO est spécialisée dans les analyses de produits et matériaux industriels qui ont pour but d'en garantir la qualité et la sécurité. ANABIO emploie environ 300 personnes, de formation scientifique, réparties dans six laboratoires situés à Strasbourg, Rennes, Marseille, Montpellier, Bordeaux et Toulouse. ANABIO possède des équipements très modernes (résonance magnétique nucléaire, spectrographes de masse, etc.).

## 3 Présentation du modèle

Chaque jour, le laboratoire réceptionne des échantillons pour lesquels différentes analyses doivent être effectuées. Les informations liées à cette activité sont gérées à partir du schéma relationnel suivant :

CLIENT (codeClient, nomClient, prénomClient, rueClient, CPClient, villeClient, telClient)

- codeClient : clé primaire.

TYPEANALYSE(refTypeAnalyse, designationTypeAnalyse, prixTypeAnalyse )

- refTypeAnalyse : clé primaire

ECHANTILLON(codeEchantillon, dateEntree, codeClient)

- codeEchantillon : clé primaire
- codeClient : clé étrangère en référence à codeClient de CLIENT

REALISER(codeEchantillon, refTypeAnalyse, dateRealisation)

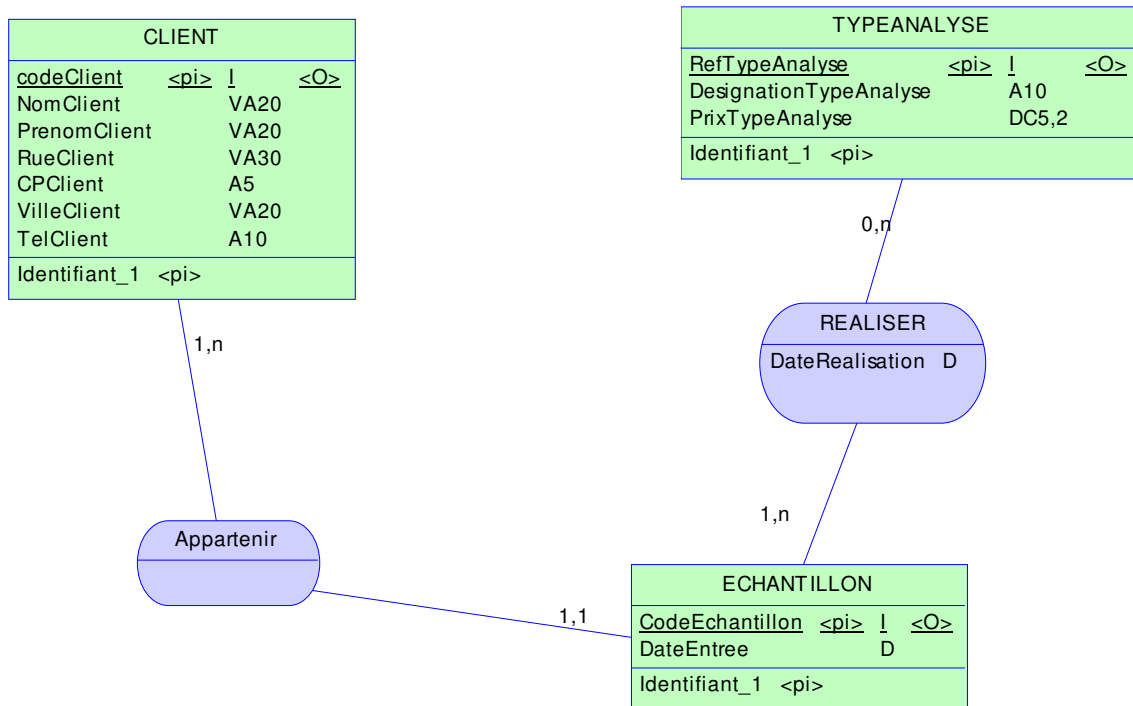
- codeEchantillon, refTypeAnalyse : clé primaire
- codeEchantillon : clé étrangère en référence à codeEchantillon de ECHANTILLON
- refTypeAnalyse : clé étrangère en référence à refTypeAnalyse de TYPEANALYSE

Les valeurs des champs *codeEchantillon* et *refTypeAnalyse* de la relation *REALISER* sont saisies à la réception de l'échantillon mais le champ *dateRealisation* n'est pas immédiatement renseigné : il le sera lorsque l'analyse aura été réalisée.



#### 4 Modèle conceptuel

Afin de vous aider dans la compréhension du problème, réaliser le modèle conceptuel à partir du modèle relationnel. Complétez, ce schéma, et barrez les symboles non utilisés.



#### 5 Réalisation de la base de donnée

- Tous les champs code et référence sont des entiers.
- Le code postal est de type chaîne de caractère sur une longueur de 5
- Les dates sont de types dates (jour, mois, année)
- Le prix est de type décimal avec 2 chiffres après la virgule
- La désignation type analyse est codée sur 10 caractères fixes.
- Les autres champs sont de types chaîne de caractères. La longueur sera déterminée par vos soins, la place dans la base devant être optimisée.

**5.1 Rédigez l'ensemble des requêtes permettant de créer la base de donnée (les contraintes seront nommées).**

##### Création de la base de donnée:

```
CREATE DATABASE "Echantillon"
WITH OWNER = postgres
ENCODING = 'SQL_ASCII'
TABLESPACE = pg_default;
```



**Création des tables :**

```
CREATE TABLE client
```

```
(  
    codeclient int4 NOT NULL,  
    nomclient varchar(20),  
    prenomclient varchar(20),  
    rueclient varchar(30),  
    cpclient char(5),  
    villeclient varchar(20),  
    telclient char(10),  
    CONSTRAINT pk_client PRIMARY KEY (codeclient)  
);
```

```
CREATE TABLE typeanalyse
```

```
(  
    reftypeanalyse int4 NOT NULL,  
    designationtypeanalyse char(10),  
    prixtypeanalyse numeric(5,2),  
    CONSTRAINT pk_typeanalyse PRIMARY KEY (reftypeanalyse)  
);
```

```
CREATE TABLE echantillon
```

```
(  
    codeechantillon int4 NOT NULL,  
    dateentree date,  
    codeclient int4 NOT NULL,  
    CONSTRAINT pk_echantillon PRIMARY KEY (codeechantillon),  
    CONSTRAINT fk_echantil_apparteni_client FOREIGN KEY (codeclient)  
        REFERENCES client (codeclient)  
);
```

```
CREATE TABLE realiser
```

```
(  
    codeechantillon int4 NOT NULL,  
    reftypeanalyse int4 NOT NULL,  
    daterealisation date,  
    CONSTRAINT pk_realiser PRIMARY KEY (reftypeanalyse,  
codeechantillon),  
    CONSTRAINT fk_realiser_realiser2_echantil FOREIGN KEY  
(codeechantillon)  
        REFERENCES echantillon (codeechantillon) MATCH SIMPLE  
        ON UPDATE RESTRICT ON DELETE RESTRICT,  
    CONSTRAINT fk_realiser_realiser_typeanal FOREIGN KEY  
(reftypeanalyse)  
        REFERENCES typeanalyse (reftypeanalyse) MATCH SIMPLE  
        ON UPDATE RESTRICT ON DELETE RESTRICT  
);
```

**5.2 Modifiez la structure de la table Client afin d'ajouter un champs email.**

```
ALTER TABLE CLIENT  
ADD Column Email VARCHAR(30);
```



## 6 Requêtes de sélection

### 6.1 *Donnez la liste des clients référencés*

```
SELECT * FROM CLIENT
```

### 6.2 *Indiquez les noms et prénoms des clients habitant Paris*

```
SELECT NomClient, PrenomClient FROM CLIENT  
WHERE VilleClient = 'Paris';
```

### 6.3 *Donnez le nombre de client référencé dans chaque ville*

```
SELECT VilleClient, Count(*) FROM CLIENT  
GROUP BY VilleClient;
```

### 6.4 *Donnez la même information par ordre décroissant du nombre de client*

```
SELECT VilleClient, Count(*) FROM CLIENT  
GROUP BY VilleClient ORDER BY 2 DESC;
```

### 6.5 *Donnez la liste des clients (nom, prénom) habitant dans le département de la Meuse.*

```
SELECT NomClient, PrenomClient FROM CLIENT  
WHERE CPClient LIKE '55%';
```

### 6.6 *Donnez les noms et prénoms des clients n'ayant pas d'adresse e-mail.*

```
SELECT NomClient, PrenomClient FROM CLIENT  
WHERE email IS NULL;
```

### 6.7 *Donnez les noms, prénoms et adresses complètes des clients ayant fourni des échantillons, ainsi que la date entre le 01-10-2005 et le 31-12-2005/*

```
SELECT NomClient, PrenomClient, RueClient, VilleClient, CPClient, Dateentree  
FROM CLIENT C, ECHANTILLON E  
WHERE C.CodeClient = E.CodeClient  
AND DateEntree BETWEEN '10-01-2005' AND '12-31-2005';
```



### 6.8 *Donnez le nombre d'analyse en cours (non réalisées)*

```
SELECT COUNT(*) FROM REALISER  
WHERE DateRealisation IS NULL;
```

### 6.9 *Donnez les noms, prénoms des clients attendant le résultat d'une analyse(analyse non réalisée).*

```
SELECT DISTINCT NomClient, PrenomClient FROM CLIENT C, ECHANTILLON E,  
REALISER R  
WHERE C.CodeClient = E.CodeClient  
AND E.CodeEchantillon = R.CodeEchantillon  
AND DateRealisation IS NULL;
```

### 6.10 *Donnez pour chaque client par ordre alphabétique (nom et prénom) le nombre d'échantillons déposés ainsi que les dates de dépôt du plus ancien et du plus*

	BTS IG 1 <sup>ère</sup> année ALSI	Evaluation no 4 Base de données et SQL	 Page 5 / 7
<b>Nom :</b>			

*récent.*

```
SELECT NomClient, PrenomClient, count(*), Min(Dateentree), Max(dateentree)
FROM CLIENT C, ECHANTILLON E
WHERE C.CodeClient = E. CodeClient
Group BY NomClient, PrenomClient;
```

**6.11** *Donnez la liste des analyses réalisées au mois de mars 2005.*

```
SELECT RefTypeAnalyse, Code Echantillon FROM REALISER
WHERE MONTH(dateRealisation) = 03 AND YEAR(DateRealisation) = 2005;
```

**6.12** *Donnez pour chaque échantillon, le nombre prévu d'analyse(réalisées ou non).*

```
SELECT CodeEchantillon, count(*) FROM REALISER
GROUP BY CodeEchantillon;
```

**6.13** *Donnez les codes et dates d'entrée des échantillons pour lesquels aucune analyse n'a été réalisée.*

```
SELECT CodeEchantillon, DateEntree FROM ECHANTILLON
WHERE CodeEchantillon NOT IN
(SELECT CodeEchantillon FROM REALISER WHERE DateRealisation IS not NULL);
```

**6.14** *Donnez les codes et dates d'entrée des échantillons pour lesquels plus de 5 analyses ont été réalisées ;*

```
SELECT CodeEchantillon, DateEntree FROM ECHANTILLON
WHERE Code Echantillon IN
(SELECT CodeEchantillon FROM REALISER
WHERE dateRealisation IS NOT NULL
GROUP BY Code Echantillon
HAVING COUNT(*) >5);
```

**6.15** *Donnez les désignations des analyses dont le prix est inférieur à celui de l'analyse possédant la référence 'A102'.*

```
SELECT DesignationTypeANalyse FROM TYPEANALYSE
WHERE PrixTypeAnalyse <
(SELECT PrixTypeAnalyse FROM TYPEANALYSE WHERE DesignationTypeAnalyse =
'A102')
```

Autre solution :

```
SELECT TR.DesignationTypeAnalyse FROM TYPEANALYSE TM, TYPEANALYSE TR
WHERE TM.designationTypeAnalyse = 'A102'
AND TR.PrixTypeAnalyse < TM.PrixTypeAnalyse
```

**7** Requêtes de mise à jour

**7.1** *La base est vierge. Réalisez l'insertion d'un jeu de données dans les différentes*



*tables. Les données seront définies par vous-même à votre convenance.*

```
INSERT INTO CLIENT VALUES (1,'Dupont','Jean','Rue de Jean',  
'75000','Paris','01020304',Null);  
INSERT INTO CLIENT VALUES (2,'Durand','Pierre','Rue de Pierre', '75000', 'Paris',  
'01020304', 'toto@lui.fr');  
INSERT INTO CLIENT VALUES (3,'Autremont','Paul','Rue de Paul', '55000','Bar le Duc',  
'01020304', Null);  
INSERT INTO CLIENT VALUES (4,'Reste','Jacques','Rue de Jacques', '57000', 'Metz',  
'01020304', 'uneadresse@unsite');
```

```
INSERT INTO TYPEANALYSE VALUES (1,'A200',8.10);  
INSERT INTO TYPEANALYSE VALUES (2,'A102',10.00);  
INSERT INTO TYPEANALYSE VALUES (3,'A300',12.20);  
INSERT INTO TYPEANALYSE VALUES (4,'A400',5);  
INSERT INTO TYPEANALYSE VALUES (5,'A500',4.20);
```

```
INSERT INTO ECHANTILLON VALUES (1,'05-10-2005',1);  
INSERT INTO ECHANTILLON VALUES (2,'12-10-2005',2);  
INSERT INTO ECHANTILLON VALUES (3,'10-20-2005',3);  
INSERT INTO ECHANTILLON VALUES (4,'02-02-2006',4);  
INSERT INTO ECHANTILLON VALUES (5,'12-30-2005',2);  
INSERT INTO ECHANTILLON VALUES (6,'10-10-2005',1);  
INSERT INTO ECHANTILLON VALUES (7,'05-10-2005',4);  
INSERT INTO ECHANTILLON VALUES (8,'05-07-1999',3);  
INSERT INTO ECHANTILLON VALUES (9,'02-28-2005',1);
```

```
INSERT INTO REALISER VALUES (1,1,'10-12-2005');  
INSERT INTO REALISER VALUES (1,2,Null);  
INSERT INTO REALISER VALUES (2,1,'08-10-2006');  
INSERT INTO REALISER VALUES (2,2,Null);  
INSERT INTO REALISER VALUES (3,1,'06-15-2005');  
INSERT INTO REALISER VALUES (3,2,Null);  
INSERT INTO REALISER VALUES (2,4,'05-20-1999');  
INSERT INTO REALISER VALUES (1,4,Null);  
INSERT INTO REALISER VALUES (4,1,'02-28-2005');  
INSERT INTO REALISER VALUES (1,3,Null);  
INSERT INTO REALISER VALUES (3,3,'12-30-2005');  
INSERT INTO REALISER VALUES (4,4,Null);  
INSERT INTO REALISER VALUES (2,3,'01-29-2006');  
INSERT INTO REALISER VALUES (3,4,Null);
```

### **7.2 Augmentez de 10% tous les prix des analyses.**

```
UPDATE TYPEANALYSE  
SET PrixTypeAnalyse = PrixTypeAnalyse*1.1;
```

### **7.3 Il a été défini un prix plancher de 8 Euros pour toutes les analyses. Mettez à jour la table ANALYSE.**

```
UPDATE TYPEANALYSE  
SET PrixTypeAnalyse = 8  
WHERE PrixTypeAnalyse <8;
```



**7.4** *Aujourd'hui, toutes les analyses en cours ont été réalisées. Mettez à jour la base.*

```
UPDATE REALISER
SET DateRealisation = '02-06-2006'
WHERE DateRealisation IS NULL;
```

**7.5** *Le client Dupont vient de fournir son adresse e-mail (Dupont@btsig.fr). Mettez à jour la table correspondante*

```
UPDATE CLIENT
SET email = 'Dupont@btsig.fr'
WHERE NomClient = 'Dupont';
```

**7.6** *Suite à un bug informatique, des entrées ont été réalisées le 01 février 2007 au lieu du 1er février 2006. Mettez à jour la base.*

```
UPDATE ECHANTILLON
SET DateEntree = '02-01-2006'
WHERE DateEntree = '02-01-2007';
```

```
UPDATE REALISER
SET Date Realisation = '02-01-2006'
WHERE DateRealisation = '02-01-2007';
```

**7.7** *Afin de préparer la nouvelle campagne, de nouvelles analyses ont été définies. Ces nouvelles analyses sont disponibles dans une table ANALYSE2006 dont la structure (champs, types de donnée) est identique à TYPEANALYSE. Mettez à jour la table TYPEANALYSE à partir de la table ANALYSE2006.*

```
INSERT INTO TYPEANALYSE
SELECT * FROM ANALYSE2006;
```