

BASE DE DONNEE

TRVEAUX PRATIQUES N°3

SNOUSSI ANIS
BARGHOUDA MOHAMED LAMINE

REALISE PAR :

SNOUSSI Anis
BARGHOUDA Mohamed
Lamine

COMPTE RENDU

BUTS :

- La création d'une base de données contenant plusieurs tables.
- Recherche de données dans la base donnée.
- L'exécution des requêtes avancés sur la base de données.



1. Création de la base donnée

a. Création des tables

i. Création du table client :

```
CREATE TABLE client (  
  `NumCl` varchar(10) NOT NULL,  
  `Nom` varchar(30) NOT NULL,  
  `Adresse` varchar(15) NOT NULL,  
  `Nationalité` varchar(15) NOT NULL,  
  PRIMARY KEY (`NumCl`));
```

ii. Création du table detailOperation :

```
CREATE TABLE detailoperation (`CodeOp` int(11) NOT NULL, `CodeDevise` varchar(10),  
  `Montant` int(11) NOT NULL, `Contreval` float NOT NULL, PRIMARY KEY (`CodeOp`,  
  `CodeDevise`);  
ALTER TABLE detailoperation ADD CONSTRAINT liaison2 FOREIGN KEY (CodeOp) REFERENCES  
  operation.CodeOp  
ALTER TABLE detailoperation ADD CONSTRAINT liaison3 FOREIGN KEY (CodeDevise) REFERE  
  NCES devise.CodeD
```

iii. Création du table devise :

```
CREATE TABLE devise (`CodeD` varchar(10) NOT NULL, `Libelle` varchar(20) NOT NULL,  
  `Unite` int(11) NOT NULL, `CoursAchat` float NOT NULL, `CoursVente` float NOT NULL,  
  PRIMARY KEY (`CodeD`));  
ALTER TABLE devise ADD CONSTRAINT Condition1 CHECK (libelle = UPPER(libelle));  
ALTER TABLE devise ADD CONSTRAINT Condition2 CHECK (Unite > 0);
```

iv. Création du table opération :

```
CREATE TABLE operation (`CodeOp` int(11) NOT NULL, `TypeOp` varchar(5) NOT NULL,  
  `DateOp` date, `NumeroCl` varchar(10) DEFAULT NULL, PRIMARY KEY (`CodeOp`));  
ALTER TABLE operation ADD CONSTRAINT liaison1 FOREIGN KEY (NumeroCl)  
  REFERENCES client.NumCl;  
ALTER TABLE operation ADD CONSTRAINT Condition3 CHECK  
  ( TypeOp = "Achat" OR TypeOp = "Vente");
```

b. Ajout des données dans les tables

i. Insertion dans client :

```
INSERT INTO client (`NumCl`, `Nom`, `Adresse`, `Nationalité`) VALUES
('K029534', 'Asma Tlili', 'Tunis', 'Tunisienne'),
('K054679', 'Rim Fakhfakh', 'Tunis', 'Tunisienne'),
('K145309', 'Ines Lakhoua', 'Tunis', 'Tunisienne'),
('K549027', 'Ahmed Ben Marzouk', 'Sfax', 'Tunisienne'),
('M123497', 'Zied Soltani', 'Tunis', 'Tunisienne'),
('M125302', 'Mohamed Ben Salem', 'Tunis', 'Tunisienne'),
('M132089', 'Zahra Sellami', 'Sfax', 'Tunisienne'),
('M134236', 'Amna Riahi', 'Sousse', 'Tunisienne'),
('M202745', 'Samir Elouni', 'Bizerte', 'Tunisienne'),
('M243678', 'Sami Meddeb', 'Tunis', 'Tunisienne');
```

ii. Insertion dans detailoperation :

```
INSERT INTO detailoperation (`CodeOp`, `CodeDevis`, `Montant`, `Contreval`) VALUES
(1, 'BEF056', 7000, 3.221),
(1, 'FRF250', 2500, 1.981),
(2, 'CAD124', 600, 0.914),
(2, 'USD840', 500, 1.462),
(3, 'ESP724', 1250, 7.603),
(4, 'DEM280', 600, 6.644),
(4, 'GBP826', 200, 2.121),
(5, 'SAR682', 200, 3.802),
(5, 'USD840', 100, 1.423),
(6, 'EUR978', 100, 1.423),
(7, 'DEM280', 400, 6.644),
(7, 'EUR978', 150, 1.3),
(7, 'FRF250', 150, 1.981);
```

iii. Insertion dans devise :

```
INSERT INTO devise (`CodeD`, `Libelle`, `Unite`, `CoursAchat`, `CoursVente`) VALUES
('BEF056', 'FRANC BELG', 100, 3.137, 3.221),
('CAD124', 'DOLLAR CANADIEN', 1, 0.9405, 0.014),
('CHF756', 'FRANC SUISSE', 10, 9.5018, 8.865),
('DEM280', 'DEUTSCHE M', 10, 6.474, 6.644),
('ESP724', 'PESTA ESPAGNOL', 1000, 7.603, 7.818),
('EUR978', 'EURO', 1, 1.264, 1.3),
('FRF250', 'FRANC FRANCAIS', 10, 1.932, 1.981),
('GBP826', 'LIVRE STERLING', 1, 2.067, 2.121),
('ITL380', 'LIRE ITALIEN', 1000, 11.436, 12.156),
('SAR682', 'RIYAL SAOUDIEN', 10, 3.802, 3.926),
('USD840', 'DOLLAR US', 1, 1.423, 1.462);
```

iv. Insertion dans opération :

```
INSERT INTO operation (`CodeOp`, `TypeOp`, `DateOp`, `NumeroCl`) VALUES  
(1, 'Vente', '2001-01-01', 'M123497'),  
(2, 'Vente', '2001-01-01', 'K145309'),  
(3, 'Achat', '2001-01-01', 'K054679'),  
(4, 'Vente', '2001-02-05', 'M125302'),  
(5, 'Achat', '2001-02-05', 'M132089'),  
(6, 'Achat', '2001-02-05', 'M123497'),  
(7, 'Vente', '2001-03-22', 'M134236');
```

2. Requêtes simples de données

a. Sélection des codes devise commençant par la lettre 'e'

```
SELECT CodeD FROM devise WHERE CodeD LIKE 'e%';
```

b. Sélection des codes devise qui ont un cours d'achat strictement > 2.00

```
SELECT CodeD FROM devise WHERE coursachat > 2.00;
```

c. Sélection des codes devise dont l'unité entre 5 et 50

```
SELECT CodeD from devise WHERE (unite > 5 AND unite < 50);
```

d. Sélection des clients dont le nom contient 'Eloumi'

```
SELECT * FROM client WHERE nom like '%Eloumi%';
```

e. Sélection des opérations effectuées entre 5 et 10 février

```
SELECT * FROM operation WHERE month(DateOp) = 2 AND day(DateOp) >= 5 AND day(DateOp) <= 10;
```

f. Sélection des codes et libellés des devises dont le cours d'achat est < 4

```
SELECT CodeD , libelle FROM devise WHERE coursachat < 4;
```

g. Sélection des opérations effectuées à des dates antérieures d'au moins 30 jours à la date actuelle

```
SELECT * FROM operation WHERE DateOp < adddate(date(now()), -30);
```

h. Sélection des clients dont la nationalité est tunisienne et l'adresse est Sfax

```
SELECT * FROM client WHERE nationalité LIKE 'Tunisienne' AND adresse LIKE 'Sfax';
```

i. Sélection des clients habitant à Bizerte ou Sousse

```
SELECT * FROM client WHERE adresse LIKE 'Bizerte' OR adresse LIKE 'Sousse';
```

3. Requêtes plus avancées

a. Augmentation le cours de vente de 10% pour les devises commençantes par la lettre 'e'

```
UPDATE devise SET CoursAchat = 1.1*CoursAchat WHERE CodeD LIKE 'C%';
```

b. Affichage le nombre de clients dans la table client

```
SELECT COUNT(NumCl) FROM Client;
```

c. Affichage du nom des clients qui ont effectué aucune transaction

```
SELECT * FROM client WHERE ( NumCl NOT IN (SELECT client FROM operation));
```

d. Calcul du cours d'achat le plus bas, le max et la moyenne

```
SELECT MIN(CoursAchat) FROM devise;  
SELECT MAX(CoursAchat) FROM devise;  
SELECT ( (SELECT SUM(CoursAchat) FROM devise) / (SELECT COUNT(CoursAchat) FROM devise));
```

e. Affichage du nombre du clients dans la ville Sfax

```
SELECT COUNT(NumCl) FROM client WHERE adresse LIKE 'Sfax';
```

f. Affichage du nombre de clients dans chaque ville

```
Select adresse,COUNT(*)FROM client GROUP BY adresse;
```

- g. Affichage des clients qui habitent à Sousse ou qui ont effectué des opérations à une date <= 10/02/2001

```
SELECT * FROM client WHERE Adresse LIKE 'Sousse' AND ( NumCl IN (SELECT client FROM operation WHERE DateOp <= '2001-02-10' ))
```

- h. Affichage du nom des clients qui ont effectué des opérations

```
SELECT Nom FROM client WHERE (NumCl IN (SELECT client FROM operation))
```

- i. Affichage de la valeur de chaque opération effectuée par un client

```
SELECT montant FROM detailoperation WHERE codeOp IN (SELECT codeOp FROM operation WHERE client IN (SELECT NumCl FROM client))
```

- j. Affichage de l'ID des clients qui ont effectué une vente supérieure à 100

```
SELECT NumCl FROM client WHERE (NumCl IN (SELECT Client FROM operation WHERE (TypeOp LIKE 'Vente' AND (CodeOp IN (SELECT CodeOp FROM detailoperation WHERE Montant >100)))))
```

- k. Affichage des transactions des clients qui ont le nom Zied

```
SELECT * FROM operation WHERE client IN (SELECT NumCl FROM client WHERE nom LIKE 'Zied');
```

- l. Affichage du montant et nom des clients qui ont effectué une opération de vente

```
SELECT detailoperation.montant, client.nom FROM detailoperation,client,operation WHERE client.numCl = operation.Client AND operation.codeOp=detailoperation.codeOp AND operation.typeOp="vente";
```