

Cours 1 SQL

Requêtes de base

1. Contexte :

Soit la base de données de données composée de la seule table suivante :
Produit (idProduit, designation, prix, qte, categorie).

2. Administration de la base de données

a. Création de la base de données

- Requête de création de la base de données :
create database stock_266 ;
- Requête d'affichage des bases de données :
show databases ;
- Utilisation de la base de données
use stock_266 ;
- Lister les tables de la base données
show tables ;
- Détruire une base de données
drop database stock_266 ;

```
[mysql> create database stock_266;  
Query OK, 1 row affected (0,00 sec)  
  
[mysql> use stock_266;  
Database changed  
[mysql> show tables;  
Empty set (0,01 sec)
```

b. Requête de création de la table

La table produit :

Produit (idProduit, designation, prix, qte, categorie).
create table produit (
idProduit int (3) not null auto_increment,
designation varchar(50),
prix float ,
qte int ,
categorie varchar(50),
primary key (idProduit)
);

```
mysql> create table produit (
  -> idProduit int (3) not null auto_increment,
  -> designation varchar(50),
  -> prix float ,
  -> qte int ,
  -> categorie varchar(50),
  -> primary key (idProduit)
  -> );
Query OK, 0 rows affected (0,32 sec)

mysql> show tables ;
+-----+
| Tables_in_stock_266 |
+-----+
| produit              |
+-----+
1 row in set (0,00 sec)
```

c. Stockage des données : insertion dans la table

Syntaxe : insert into nomTable values (liste des valeurs);

insert into produit values (null, "Café", 1.20, 23, "Alimentaire");

insert into produit values (null, "Sucre", 0.80, 10, "Alimentaire");

insert into produit values (null, "Aspirine", 3.40, 20, "Médicament");

d. Affichage des données

Syntaxe : select * from nomTable ;

select * from produit ;

```
mysql> select * from produit ;
+-----+-----+-----+-----+-----+
| idProduit | designation | prix | qte | categorie |
+-----+-----+-----+-----+-----+
| 1         | Café       | 1.2 | 23  | Alimentaire |
| 2         | Sucre      | 0.8 | 10  | Alimentaire |
| 3         | Aspirine   | 3.4 | 20  | Médicament  |
+-----+-----+-----+-----+-----+
3 rows in set (0,02 sec)
```

Sélection de champs parmi ceux de la table

Syntaxe : select champs1, champs2, ... from nomTable ;

Afficher seulement la désignation et le prix des produits :

select designation, prix from produit ;

```
mysql> select designation, prix from produit ;
+-----+-----+
| designation | prix |
+-----+-----+
| Café       | 1.2  |
| Sucre      | 0.8  |
| Aspirine   | 3.4  |
+-----+-----+
3 rows in set (0,02 sec)
```

- Sélection avec des clauses :

Les opérateurs de test : >, >=, <, <=, !=, =

Les opérateurs logiques : and, or, not

Syntaxe : `select * from nomTable where clause ;`

clause peut être simple ou composée à l'aide des opérateurs logiques.

- Afficher les produits ayant un prix supérieur à 1 euros.

`select * from produit where prix > 1 ;`

```
mysql> select * from produit where prix > 1 ;
+-----+-----+-----+-----+-----+
| idProduit | designation | prix | qte | categorie |
+-----+-----+-----+-----+-----+
|          1 | Café       | 1.2  | 23  | Alimentaire |
|          3 | Aspirine   | 3.4  | 20  | Médicament  |
+-----+-----+-----+-----+-----+
2 rows in set (0,04 sec)
```

- Afficher les produits à commander dans la quantité est critique (inférieure ou égale à 20).
`Select * from produit where qte <= 20 ;`
- Afficher les produits alimentaires qui ont un prix supérieur à 1 euros
`select * from produit where categorie = "Alimentaire" and prix > 1 ;`
- Afficher les prix totaux des produits.
`select (prix * qte) from produit ;`
- On peut créer des alias pour les champs calculés ou sélectionnés
`select (prix*qte) as total from produit ;`

```
mysql> select (prix*qte) as total from produit ;
+-----+
| total |
+-----+
| 27.600001096725464 |
| 8.00000011920929  |
| 68.00000190734863  |
+-----+
3 rows in set (0,00 sec)
```

e. Fonctions mathématiques

Min, Max, SUM, AVG, Count

Min : donne la plus petite valeur de la colonne sélectionnée

Max : donne la plus grande valeur de la colonne sélectionnée

SUM : additionne toutes les valeurs de la colonne

AVG : donne la valeur moyenne des valeurs de la colonne

Count : compte le nombre d'enregistrements dans la table.

- Quel est le prix moyen des produits ?

`select avg(prix) as prixMoyen from produit ;`

```
mysql> select avg(prix) as prixMoyen from produit ;
```

prixMoyen
1.8000000516573589

```
1 row in set (0,01 sec)
```

- Quelle est la quantité la plus petite de tous les produits ?
select min(qte) as qteMin from produit ;

```
mysql> select min(qte) as qteMin from produit ;
```

qteMin
10

```
1 row in set (0,01 sec)
```

- Quel est le chiffre d'affaires global du stock ?
select sum(prix*qte) as chiffreAffaires from produit ;

```
mysql> select sum(prix*qte) as chiffreAffaires from produit ;
```

chiffreAffaires
103.60000312328339

```
1 row in set (0,01 sec)
```

- Quel est le prix le plus élevé des produits ?
select max(prix) as maxPrix from produit ;

```
mysql> select max(prix) as maxPrix from produit ;
```

maxPrix
3.4000000953674316

```
1 row in set (0,01 sec)
```

- Combien il y a de produits dans la table ?
select count(*) as nbProduits from produit ;

```
mysql> select count(*) as nbProduits from produit ;
```

nbProduits
3

```
1 row in set (0,01 sec)
```

- Combien y a-t-il de produits alimentaires ?
select count(*) as nbAlimentaires from produit where categorie = "Alimentaire" ;

```
mysql> select count(*) as nbAlimentaires from produit where categorie = "Alimentaire" ;
+-----+
| nbAlimentaires |
+-----+
|                2 |
+-----+
1 row in set (0,04 sec)
```

- Quel est le chiffre d'affaires des produits non alimentaires ?
 select sum(qte * prix) as chiffreNonAlimentaire from produit where
 categorie != "Alimentaire" ;

f. Modification de la structure d'une table

Après la création d'une table, on peut apporter des modifications dans structure :

- Ajouter une colonne (un champs)
- Supprimer une colonne (un champs)
- Modifier le type d'une colonne (un champs)

Syntaxe :

Alter table nomTable [ADD, DROP, MODIFY] colonne type_colonne ;

Pour afficher la structure de la table : desc nomTable ;

- Ajouter une colonne dans la table produit qui s'appelle date_achat de type date.

Alter table Produit add date_achat date ;

```
mysql> desc produit ;
+-----+-----+-----+-----+-----+-----+
| Field      | Type      | Null | Key | Default | Extra      |
+-----+-----+-----+-----+-----+-----+
| idProduit  | int(3)    | NO   | PRI | NULL    | auto_increment |
| designation | varchar(50) | YES  |     | NULL    |              |
| prix       | float     | YES  |     | NULL    |              |
| qte        | int(11)   | YES  |     | NULL    |              |
| categorie  | varchar(50) | YES  |     | NULL    |              |
| date_achat | date      | YES  |     | NULL    |              |
+-----+-----+-----+-----+-----+-----+
6 rows in set (0,01 sec)
```

- Ajouter une colonne dans la table produit qui s'appelle disponible de type booléen.

Alter table Produit add disponible boolean ;

```
[mysql> desc produit ;
```

Field	Type	Null	Key	Default	Extra
idProduit	int(3)	NO	PRI	NULL	auto_increment
designation	varchar(50)	YES		NULL	
prix	float	YES		NULL	
qte	int(11)	YES		NULL	
categorie	varchar(50)	YES		NULL	
date_achat	date	YES		NULL	
disponible	tinyint(1)	YES		NULL	

```
7 rows in set (0,02 sec)
```

Si on veut supprimer la colonne disponible :

Alter table Produit Drop disponible ;

g. Modification des enregistrements de la table

Pour modifier les données des colonnes d'une table, on utilise la requête update avec la syntaxe suivante :

update nomTable set colonne1 = valeur, colonne2 = valeur ... where clause ;

- La quantité du produit café passe de 23 à 50. Réaliser la modification.
Update Produit set qte = 50 where designation = "Café" ;

```
[mysql> Update Produit set qte = 50 where designation = "Café" ;
Query OK, 1 row affected (0,09 sec)
Rows matched: 1 Changed: 1 Warnings: 0

[mysql> select * from produit;
```

	idProduit	designation	prix	qte	categorie	date_achat	disponible
	1	Café	1.2	50	Alimentaire	NULL	NULL
	2	Sucre	0.8	10	Alimentaire	NULL	NULL
	3	Aspirine	3.4	20	Médicament	NULL	NULL

```
3 rows in set (0,01 sec)
```

- La date d'achat du café est : 2022-12-20. Réaliser la modification.
Update Produit set date_achat = "2022-12-20 " where designation = "Café" ;

```
[mysql> Update Produit set date_achat = "2022-12-20 " where designation = "Café" ;
Query OK, 1 row affected (0,01 sec)
Rows matched: 1 Changed: 1 Warnings: 0

[mysql> select * from produit;
```

	idProduit	designation	prix	qte	categorie	date_achat	disponible
	1	Café	1.2	50	Alimentaire	2022-12-20	NULL
	2	Sucre	0.8	10	Alimentaire	NULL	NULL
	3	Aspirine	3.4	20	Médicament	NULL	NULL

```
3 rows in set (0,00 sec)
```

- Le médicament Aspirine n'est plus disponible. Réaliser la modification.

Update Produit set disponible = 0 where designation = "Aspirine";

```
mysql> Update Produit set disponible = 0 where designation = "Aspirine";
Query OK, 1 row affected (0,03 sec)
Rows matched: 1  Changed: 1  Warnings: 0
```

```
[mysql>
```

```
[mysql> select * from produit;
```

idProduit	designation	prix	qte	categorie	date_achat	disponible
1	Café	1.2	50	Alimentaire	2022-12-20	NULL
2	Sucre	0.8	10	Alimentaire	NULL	NULL
3	Aspirine	3.4	20	Médicament	NULL	0

```
3 rows in set (0,00 sec)
```

h. La suppression des enregistrements d'une table

Pour supprimer un enregistrement dans une table, on utilise la requête delete avec la syntaxe suivante :

Delete from nomTable where clause ;

Delete from nomTable ; // Suppression de tous les enregistrements.

- Supprimer le produit Médicament Aspirine

Delete from Produit where designation = "Aspirine";

```
mysql> Delete from Produit where designation = "Aspirine";
Query OK, 1 row affected (0,03 sec)
```

```
[mysql> select * from produit;
```

idProduit	designation	prix	qte	categorie	date_achat	disponible
1	Café	1.2	50	Alimentaire	2022-12-20	NULL
2	Sucre	0.8	10	Alimentaire	NULL	NULL

```
2 rows in set (0,01 sec)
```

i. Les fonctions sur les dates

On peut manipuler les dates avec des fonctions réservées :

- obtenir l'année : year(uneDate)
- obtenir le mois : month(uneDate)
- obtenir le jour : day (uneDate)
- tester une date comprise entre deux dates : between

Lister les produits achetés en 2022.

select * from Produit where year(date_achat) = "2022";

j. La clause Like

Cette clause permet de rechercher des enregistrements avec une sélection sur les champs : ? remplace un seul caractère et % remplace plusieurs caractères.

Syntaxe : Select * from Table where colonne like "%dec" and ... ;

La colonne commence par n'importe quels caractères et se finie par "dec".

- Afficher les produits tels que la désignation contient la lettre "C"
Select * from produit where designation like "%C%";

```
[mysql> Select * from produit where designation like "%C%";
```

idProduit	designation	prix	qte	categorie	date_achat	disponible
1	Café	1.2	50	Alimentaire	2022-12-20	NULL
2	Sucre	0.8	10	Alimentaire	NULL	NULL

```
2 rows in set (0,00 sec)
```