

REPUBLIQUE DU SENEGAL



Un Peuple – Un But – Une Foi

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR DE LA RECHERCHE ET DE L'INNOVATION

UNIVERSITE IBA DER THIAM DE THIES



**U.F.R DES SCIENCES ET TECHNOLOGIES- U.F.R DES SCIENCES
ECONOMIQUES ET SOCIALES**

Projet : création d'une application qui permet de faire l'interfaçage
entre **PYTHON** et **MSQL**

**Master 1 en Sciences des Données et Application option (MSDA/IIA, SEM,
CAA)**

Matière : Base De Données relationnelle

Présenté par :

Kalidou Alassane Diao

Professeur: Dr Youssoufa Ly

Gora

Diallo

Khona

Fall

Jean Marcel Waly Ndiogoye

Année Académique 2023 /2024


```

create table Client(
    idClient INT NOT NULL,
    nom VARCHAR(30),
    prenom VARCHAR(40),
    ville VARCHAR(25),
    telephone INT NOT NULL,
    email VARCHAR(30),
    PRIMARY KEY(idClient));

```

Insertion des valeurs dans la table « Client »

```

INSERT INTO Client(idClient, nom, prenom, ville, telephone, email)
VALUES
    (1, 'NDIOGOYE', 'Jean Marcel Waly', 'Mboro', '772452809', 'marcel01@gmail.com'),
    (2, 'DIAO', 'Kalidou Alassane', 'Tamba', '783051784', 'diaokalidou@gmail.com'),
    (3, 'DIALLO', 'Gora', 'Matam', '760186543', 'gora123@gmail.com'),
    (4, 'FALL', 'Khona', 'Dakar', '709865432', 'fallkhona89@gmail.com'),
    (5, 'NDIAYE', 'Sokhna Diarra', 'Bignona', '770274680', 'boussodiarra@gmail.com');

```

Création de la table « Commande »

```

create table Commande(
    idCommande INT NOT NULL,
    idClient INT NOT NULL,
    idproduit INT NOT NULL,
    dateCommande DATE,
    PRIMARY KEY(idCommande),
    FOREIGN KEY (idClient) REFERENCES Client(idClient),
    FOREIGN KEY (idproduit) REFERENCES Produit(idproduit));

```

Insertion des valeurs dans la table « Commande »

```

INSERT INTO Commande(idCommande, idClient, idproduit, dateCommande)
VALUES
    (1, 5, 40, '2024-07-15'),
    (2, 2, 20, '2024-07-20'),
    (3, 1, 20, '2024-07-25'),
    (4, 4, 10, '2024-07-30'),
    (5, 1, 50, '2024-08-01'),
    (6, 1, 30, '2024-08-05'),
    (7, 3, 30, '2024-08-10'),
    (8, 3, 10, '2024-09-15'),
    (9, 5, 10, '2024-09-20'),
    (10, 4, 40, '2024-09-30');

```

Création de la table « CommandeProduit »

```
create table CommandeProduit(  
    idCommande INT NOT NULL,  
    idproduit INT NOT NULL,  
    quantite INT NOT NULL,  
    PRIMARY KEY (idCommande, idproduit),  
    FOREIGN KEY (idCommande) REFERENCES Commande(idCommande),  
    FOREIGN KEY (idproduit) REFERENCES Produit(idproduit));
```

Insertion des valeurs dans la table « CommandeProduit »

```
Insert into CommandeProduit(idCommande, idproduit, quantite)  
VALUES  
    (1, 40, 4),  
    (2, 20, 5),  
    (3, 20, 8),  
    (4, 10, 3),  
    (5, 50, 4),  
    (6, 30, 2),  
    (7, 30, 5),  
    (8, 10, 3),  
    (9, 10, 3),  
    (10, 40, 2);
```

EXPLICATION DES LIGNES DE Codes python :

```
import mysql.connector  
from mysql.connector import Error, errorcode
```

Dans ces deux lignes de codes, nous avons import le module « mysql.connector » qui nous permettra de nous connecter à notre base de données et « Error » et « errorcode » pour gérer les erreurs liés à cette connexion

```
config = {  
    'user': 'root',  
    'password': 'Kalash@2023',  
    'host': '127.0.0.1',  
    'database': 'Electromenager',  
    'raise_on_warnings': True  
}
```

Ici, nous enregistrons les informations liées à notre base de données créée dans MySQL dans une variable appelée config

```
try:  
    cnx = mysql.connector.connect(**config)
```

```
except mysql.connector.Error as err:
    if err.errno == errorcode.ER_ACCESS_DENIED_ERROR:
        print("le nom d'utilisateur ou le mot de passe n'est pas correct")
    elif err.errno == errorcode.ER_BAD_DB_ERROR:
        print("la base de données n'existe pas")
    else :
        print(" erreur non géré",err)
    exit()
```

Ce bout de code nous permet de gérer les erreurs potentielles rencontrées lors de la connexion en retournant certaines phrases nous permettant de savoir exactement les erreurs commises

Nous passons maintenant aux requêtes effectuées et les résultats retournés sont mis sous forme de liste et sont affichés après chaque requête.

```
paragraphe = 1
cursorSel = cnx.cursor()
cnx.commit()
Aff_query = "SELECT nomproduit FROM Produit WHERE prixUnitaire > 25000"
cursorSel.execute(Aff_query)
resultats = cursorSel.fetchall()
print(resultats)
```

nomproduit
Iphone Xr
Refrigerateur
Micro-ondes
Cuisiniere
Machine a laver

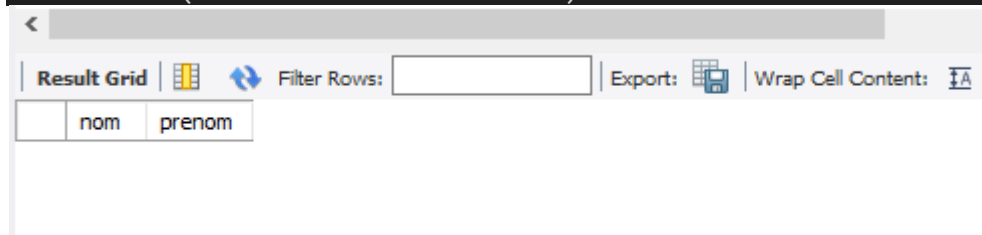
Dans cette capture d'écran se trouve notre première requete après que la connexion à la base de données ait réussi. Cette requête nous renvoie les produits dont le prix unitaire est supérieur à 25000 FCFA.

```
kali = "SELECT nom, prenom FROM Client WHERE idClient >= 3"
cursorSel.execute(kali)
result = cursorSel.fetchall()
print(result)
```

nom		prenom
DIALLO	Gora	
FALL	Khona	
NDIAYE	Sokhna Diarra	

Celle-ci retourne les prénoms et les noms des clients dont l'idClient est supérieur ou égal à 3

```
kalis = "SELECT C.nom, C. prenom FROM CLIENT C LEFT JOIN COMMANDE COM ON C.
idclient = COM.idclient WHERE COM.idcommande IS NULL"
cursorSel.execute(kalis)
result = cursorSel.fetchall()
if result:
    print("Résultats de la requête :")
    for row in result:
        print(f"nom: {row[0]}, prenom: {row[1]}")
else:
    Print ("Aucun résultat trouvé.")
```



The screenshot shows a database application window. At the top, there's a toolbar with icons for 'Result Grid', 'Filter Rows', 'Export', and 'Wrap Cell Content'. Below the toolbar, there's a table with two columns: 'nom' and 'prenom'. The table is currently empty.

La capture ci-dessus renvoie, s'il y'en a, les clients qui n'ont pas encore effectué de commande, sinon la phrase « aucun résultat trouvé ». Et pour cela, nous avons fait une jointure entre les tables Client et Commande.

```
pro_com = "select nomproduit,nom,dateCommande from Produit, Client, Commande
where Produit.idproduit = Commande.idproduit AND Client.idClient =
commande.idClient"
cursorSel.execute(pro_com)
answer = cursorSel.fetchall()
if answer:
    print("les resultats sont:")
    for row in answer:
        print(f"nomproduit: {row[0]}, nom: {row[1]}, dateCommande: {row[2]}")
else:
    print("Pas de resultats")
```

	nomproduit	nom	dateCommande
▶	Micro-ondes	NDIOGOYE	2024-07-25
	Machine a laver	NDIOGOYE	2024-08-01
	Cuisiniere	NDIOGOYE	2024-08-05
	Micro-ondes	DIAO	2024-07-20
	Cuisiniere	DIALLO	2024-08-10
	Refrigerateur	DIALLO	2024-09-15
	Refrigerateur	FALL	2024-07-30
	Aspirateurs	FALL	2024-09-30
	Aspirateurs	NDIAYE	2024-07-15
	Refrigerateur	NDIAYE	2024-09-20

Cette requête retourne les produits, les noms des clients ainsi que les dates des commandes en utilisant la jointure entre nos trois tables que sont : Produit, Client et Commande

```
waly = "select * from Produit where idproduit NOT IN (Select Distinct
idproduit from Commande)"
cursorSel.execute(waly)
resultat = cursorSel.fetchall()
print(resultat)
```

	idproduit	nomproduit	prixUnitaire	quantite
▶	6	Iphone Xr	140000	50
*	NULL	NULL	NULL	NULL

Dans cette requête, nous voulons avoir la liste des produits qui n'ont pas été commandés

```
gora = "UPDATE Produit SET prixUnitaire = 190000 WHERE idproduit=50"
cursorSel.execute(gora)
diallo = cursorSel.fetchall()
if diallo:
    print("echec")
else:
    print("maj effectue")
```

Ici, nous effectuons la mise à jour du prix unitaire de l'un de nos produits qu'est la machine à laver.

Pour vérifier que la mise à jour a été effectué, on a affiché le prix unitaire d'une machine à laver après la requête précédente

	prixUnitaire
▶	190000

```
lopez = "SELECT P. nomproduit FROM Produit P JOIN Commande C ON P.idproduit = C.idproduit JOIN Client CL ON C.idclient = CL.idclient WHERE CL.nom = 'NDIOGOYE' AND CL.prenom = 'Jean Marcel Waly'"
cursorSel.execute(lopez)
resultat = cursorSel.fetchall()
print(resultat)
```

	nomproduit
▶	Micro-ondes
	Machine a laver
	Cuisiniere

Là, nous cherchons les produits commandés par le client « NDIOGOYE Jean Marcel Waly »

```
ref="SELECT C.nom, C.prenom FROM Client C JOIN Commande CO ON C.idClient=CO.idClient WHERE CO.idproduit=10"
cursorSel.execute(ref)
ref_com = cursorSel.fetchall()
print(ref_com)
```

	nom	prenom
▶	FALL	Khona
	DIALLO	Gora
	NDIAYE	Sokhna Diarra

Nous cherchons à trouver les clients qui ont commandé au moins un Réfrigérateur

```
bar="SELECT idproduit, idclient FROM Commande WHERE dateCommande BETWEEN '2024-08-01' AND '2024-08-31'"
cursorSel.execute(bar)
rab= cursorSel.fetchall()
print(rab)
```


<div> <div><</div> <div>Result Grid</div> <div> <div>Filter Rows:</div> <div></div> </div> <div>Export:</div> <div>Wrap Cell Content:</div> </div>		
	idproduit	idclient
▶	50	1
	30	1
	30	3

Affichage des commandes effectuées en Aout 2024