

Allan, Halima, Michelle, Oussama

TP : Conception et manipulation d'une BDD à partir d'un jeu de données

---

# BASE DE DONNÉES

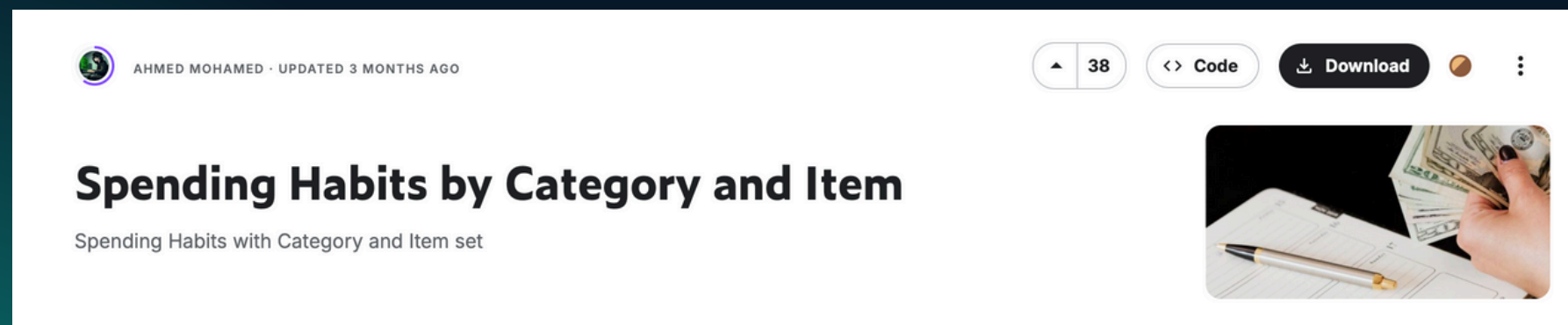
data set → modélisation → conception → requêtes SQL



# Notre sujet

Nous sommes une startup de fintech qui aide les particuliers à mieux comprendre et anticiper leurs dépenses mensuelles.

Comment comprendre le comportement d'achat des clients grâce aux données ?



A screenshot of a GitHub repository page. At the top left, it shows the user profile 'AHMED MOHAMED · UPDATED 3 MONTHS AGO'. To the right are navigation buttons: '38' (forks), '<> Code', 'Download', and a three-dot menu. The main title is 'Spending Habits by Category and Item' with a subtitle 'Spending Habits with Category and Item set'. Below the text is a thumbnail image showing a hand holding a pen over a document with a calculator and some cash in the background.



# Présentation du jeu de données

- attributs
- types de données

## Caractéristiques du jeu de données:

- 10,000 transactions enregistrées
- Clients uniques
- Détails par transaction : catégorie, article, quantité, prix
- Informations de paiement et date de transaction

spending\_patterns\_detailed.csv (793.84 kB)

Detail Compact Column

▲ Customer ID	▲ Category	▲ Item	# Quantity	# Price Per ...	# Total Spent	▲ Payment ...	▲ Location	📅 Transactio...
CUST_0159	Groceries	Milk	1	1.28	1.28	Debit Card	Mobile App	2024-11-13
CUST_0017	Friend Activities	Dinner with Friends	2	74.69	149.39	Debit Card	In-store	2023-05-04
CUST_0094	Housing and Utilities	Water Bill	1	76.06	76.06	Digital Wallet	Mobile App	2023-12-16

# Analyse du jeu de données

- attributs
- types de données

Caractéristiques du jeu de données:

- 10,000 transactions enregistrées
- Clients uniques
- Détails par transaction : catégorie, article, quantité, prix
- Informations de paiement et date de transaction

spending\_patterns\_detailed.csv (793.84 kB)

Detail Compact Column

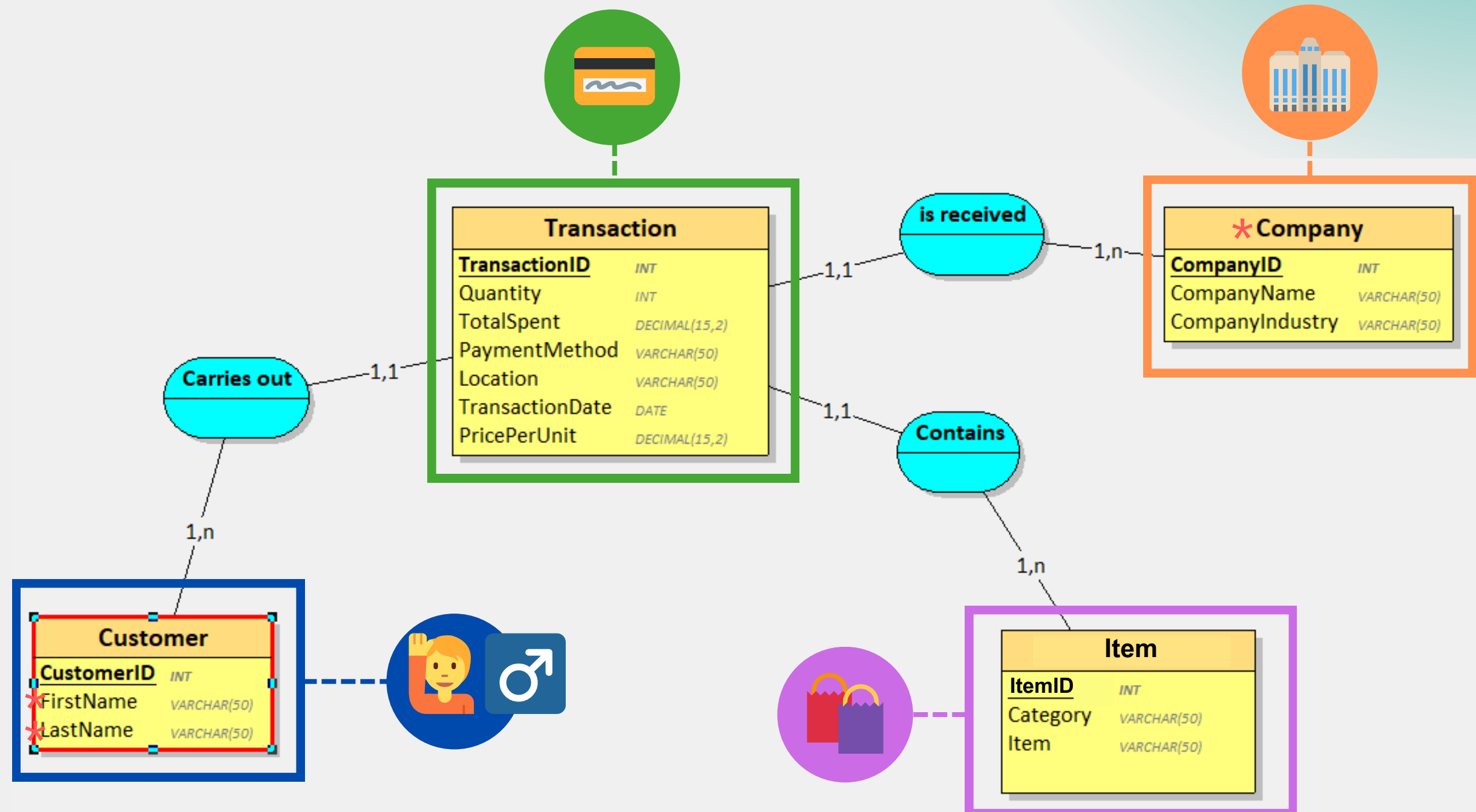
Customer ID	Category	Item	Price Per ...	Quantity	Total Spent	Payment ...	Location	Transactio...
CUST_0159	Groceries	Milk	1.28	1	1.28	Debit Card	Mobile App	2024-11-13
CUST_0017	Friend Activities	Dinner with Friends	74.69	2	149.39	Debit Card	In-store	2023-05-04
CUST_0094	Housing and Utilities	Water Bill	76.06	1	76.06	Digital Wallet	Mobile App	2023-12-16



# Modélisation de la BDD

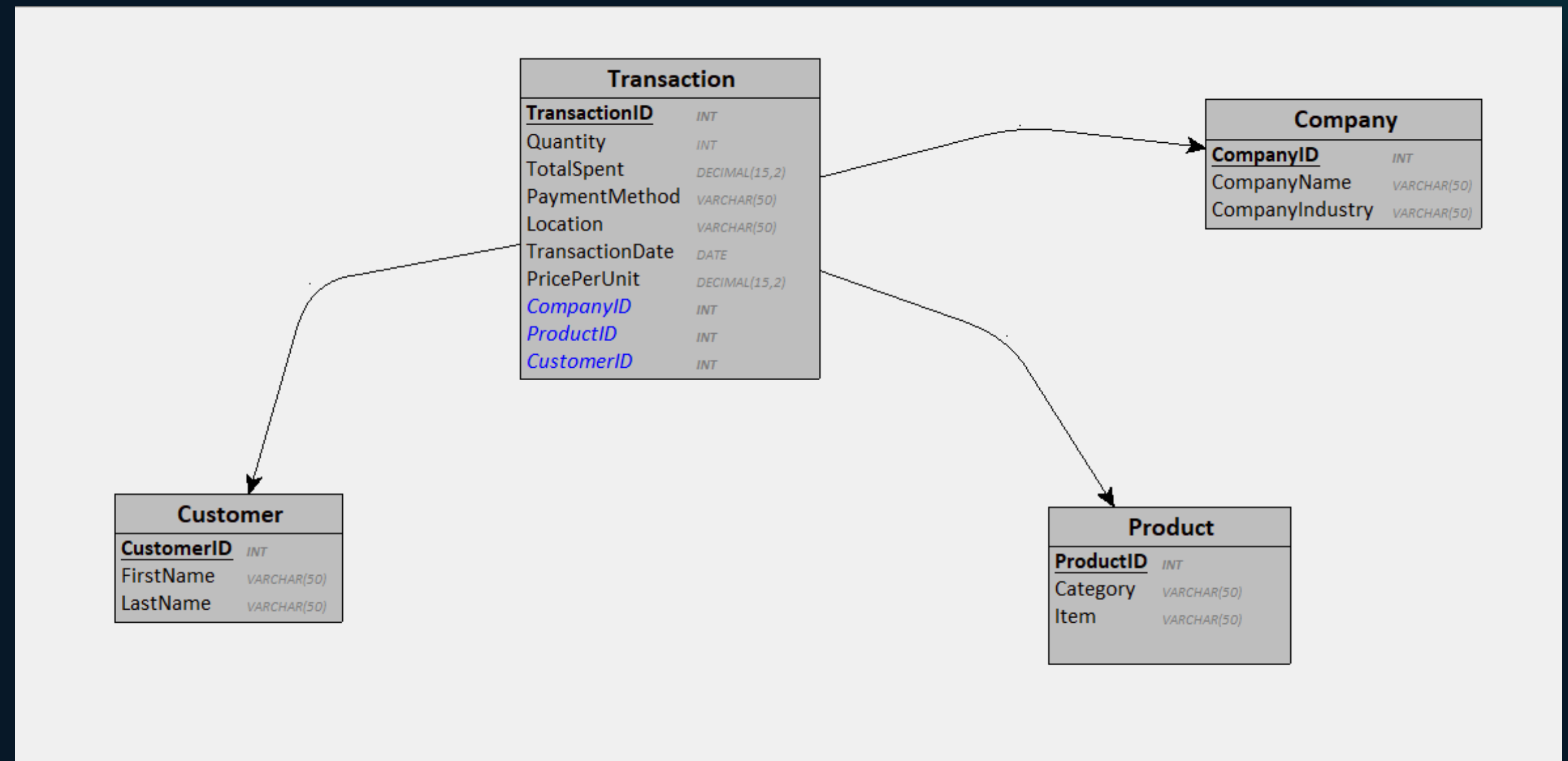
## Looping

- diagramme entité-association (DEA)
- MCD
- CamelCase
- entités
- les relations entre elles, cardinalités et liens
- attributs pour chaque entité
- types de données pour chaque attribut
- clés primaires



# Modélisation de la BDD

- MLD
- entités
- les relations entre elles
- attributs pour chaque entité
- types de données pour chaque attribut
- clés étrangères



# Contraintes d'intégrité

- **Intégrité des entités** : Chaque table a une clé primaire unique exemple `CompanyName` est unique pour éviter les doublons  
`AUTO_INCREMENT` pour `ProductID` (évite les erreurs manuelles), ..

## INTÉGRITÉ DES ENTITÉS

## INTÉGRITÉ RÉFÉRENTIELLE

- **Intégrité des données** : Contraintes de non-nullité, valeurs positives, et contraintes d'unicité exemple
- ✓ **NOT NULL** sur toutes les colonnes obligatoires.
- ✓ **CHECK (Quantity > 0)** → Pas de quantités négatives ou nulles.
- ✓ **CHECK (TotalSpent > 0)** → Assure une valeur positive.

## INTÉGRITÉ DES DONNÉES

- **Intégrité référentielle** : Les clés étrangères référencent les clés primaires existantes exemple **ON DELETE CASCADE** Supprime les transactions si le client, l'entreprise ou le produit est supprimé.

# Création de la base de données



```
1 -- Création de la base de données
2 CREATE DATABASE ProjetBDD;
3
4 -- Sélection de la base de données
5 USE ProjetBDD;
6
```



# Création des tables



```
1 -- Table: Customer
2 CREATE TABLE Customer (
3     CustomerID INT PRIMARY KEY,
4     FirstName VARCHAR(50),
5     LastName VARCHAR(50)
6 );
```

```
1 -- Table: Product
2 CREATE TABLE Product (
3     ProductID INT PRIMARY KEY AUTO_INCREMENT,
4     Category VARCHAR(50) NOT NULL,
5     Product VARCHAR(50) NOT NULL
6 );
```

```
1 -- Table: Company
2 CREATE TABLE Company (
3     CompanyID INT PRIMARY KEY,
4     CompanyName VARCHAR(50) NOT NULL UNIQUE,
5     CompanyIndustry VARCHAR(50) NOT NULL
6 );
```

```
1 CREATE TABLE Transaction (
2     TransactionID INT PRIMARY KEY,
3     Quantity INT NOT NULL CHECK(Quantity > 0),
4     PaymentMethod VARCHAR(50) NOT NULL,
5     Location VARCHAR(50) NOT NULL,
6     TransactionDate DATE NOT NULL,
7     PricePerUnit DECIMAL(15,2) NOT NULL CHECK(PricePerUnit > 0),
8     CustomerID INT NOT NULL,
9     CompanyID BIGINT NOT NULL,
10    ProductID INT NOT NULL,
11    TotalSpent DECIMAL(15,2) NOT NULL CHECK(TotalSpent = PricePerUnit * Quantity),
12    FOREIGN KEY (CustomerID) REFERENCES Customer(CustomerID) ON DELETE CASCADE,
13    FOREIGN KEY (CompanyID) REFERENCES Company(CompanyID) ON DELETE CASCADE,
14    FOREIGN KEY (ProductID) REFERENCES Product(ProductID) ON DELETE CASCADE
15 );
```

# Importation des données



```
1 INSERT INTO Transaction VALUES (1, 1, 1.28, 'Debit Card', 'Mobile App', '2024-11-13', 1.28, 1, 10360655946119, 1);
2 INSERT INTO Transaction VALUES (2, 2, 149.39, 'Debit Card', 'In-store', '2023-05-04', 74.69, 2, 96705349659832, 2);
3 INSERT INTO Transaction VALUES (3, 1, 76.06, 'Digital Wallet', 'Mobile App', '2023-12-16', 76.06, 3, 72724278245498, 3);
4 INSERT INTO Transaction VALUES (4, 5, 56.18, 'Cash', 'In-store', '2024-01-28', 11.24, 4, 78645062816044, 4);
5 INSERT INTO Transaction VALUES (5, 3, 131.12, 'Debit Card', 'In-store', '2023-07-28', 43.71, 5, 2771984988393, 5);
6 INSERT INTO Transaction VALUES (6, 2, 31.74, 'Digital Wallet', 'Mobile App', '2024-07-14', 15.87, 4, 2771984988393, 5);
7 INSERT INTO Transaction VALUES (7, 1, 371.29, 'Debit Card', 'Mobile App', '2024-03-08', 371.29, 6, 18900639258735, 6);
8 INSERT INTO Transaction VALUES (8, 1, 140.59, 'Debit Card', 'Mobile App', '2023-01-14',
```

Création de la BDD → Création des tables → **Importation des données**

	TransactionID	Quantity	TotalSpent	PaymentMethod	Location	TransactionDate	PricePerUnit	CustomerID
<input type="checkbox"/> Éditer Copier Supprimer	1	1	1.28	Debit Card	Mobile App	2024-11-13	1.28	
<input type="checkbox"/> Éditer Copier Supprimer	2	2	149.39	Debit Card	In-store	2023-05-04	74.69	
<input type="checkbox"/> Éditer Copier Supprimer	3	1	76.06	Digital Wallet	Mobile App	2023-12-16	76.06	
<input type="checkbox"/> Éditer Copier Supprimer	4	5	56.18	Cash	In-store	2024-01-28	11.24	
<input type="checkbox"/> Éditer Copier Supprimer	5	3	131.12	Debit Card	In-store	2023-07-28	43.71	
<input type="checkbox"/> Éditer Copier Supprimer	6	2	31.74	Digital Wallet	Mobile App	2024-07-14	15.87	
<input type="checkbox"/> Éditer Copier Supprimer	7	1	371.29	Debit Card	Mobile App	2024-03-08	371.29	
<input type="checkbox"/> Éditer Copier Supprimer	8	1	140.59	Debit Card	Mobile App	2023-01-14	140.59	
<input type="checkbox"/> Éditer Copier Supprimer	9	1	1011.41	Credit Card	In-store	2023-08-15	1011.41	
<input type="checkbox"/> Éditer Copier Supprimer	10	5	234.20	Digital Wallet	Online	2024-10-15	46.84	

# Sélection de données (SELECT)

```

1 -- Requête pour afficher toutes les transactions avec les informations du client, du produit et
  de l'entreprise associée
2 SELECT c.CustomerID, c.FirstName, c.LastName, t.TransactionID, t.TransactionDate, t.TotalSpent,
  t.Quantity, t.PaymentMethod, t.Location, t.PricePerUnit, p.ProductID, p.Product AS ProductName,
  p.Category, co.CompanyID, co.CompanyName, co.CompanyIndustry
3
4 FROM transaction t JOIN customer c ON t.CustomerID = c.CustomerID
5 JOIN product p ON t.ProductID = p.ProductID
6 JOIN company co ON t.CompanyID = co.CompanyID;
7
  
```



CustomerID	FirstName	LastName	TransactionID	TransactionDate	TotalSpent	Quantity	PaymentMethod	Location	PricePerUnit	ProductID
5	Andres	Mercier	5	2023-07-28	131.12	3	Debit Card	In-store	43.71	5
4	Pierre	Rodriguez	6	2024-07-14	31.74	2	Digital Wallet	Mobile App	15.87	5
9	Louis	Lemoine	10	2024-10-15	234.20	5	Digital Wallet	Online	46.84	9
14	Michael	Torres	16	2023-12-29	564.54	2	Credit Card	Mobile App	282.27	14
20	Camille	Fernandez	34	2023-01-07	193.03	5	Debit Card	Online	38.61	9
14	Michael	Torres	35	2024-05-25	250.74	3	Credit Card	Mobile App	83.58	14
4	Pierre	Rodriguez	43	2024-03-25	137.95	3	Cash	Mobile App	45.98	9
4	Pierre	Rodriguez	49	2024-11-29	584.24	2	Cash	Online	292.12	14

# Agrégation (GROUP BY)

```

1 -- Requête pour trouver les produits les plus vendus en termes de quantité totale achetée
2
3 SELECT
4     p.ProductID,
5     p.Product AS ProductName,
6     SUM(t.Quantity) AS TotalQuantitySold
7 FROM transaction t
8 JOIN product p ON t.ProductID = p.ProductID
9 GROUP BY p.ProductID, p.Product
10 ORDER BY TotalQuantitySold DESC;
11

```

ProductID	ProductName	TotalQuantitySold
2	Books	823
4	Car	819
33	Public Transit	815
37	Shoes	813
7	Clothes	795
21	Gift Cards	787
36	Shampoo	787
22	Gym Membership	753
40	Soap	686
5	Car Repair	651
43	Toothpaste	625
12	Dinner with Friends	613
11	Dentist Visit	593
45	Video Games	592
9	Concert Tickets	575

# Jointures (JOIN)

```

1 -- Requête pour afficher le nombre de transactions et la dépense moyenne par client
2
3 SELECT
4     c.CustomerID,
5     c.FirstName,
6     c.LastName,
7
8     COUNT(t.TransactionID) AS TotalTransactions,
9     AVG(t.TotalSpent) AS AverageSpending
10
11 FROM transaction t
12 JOIN customer c ON t.CustomerID = c.CustomerID
13 GROUP BY c.CustomerID, c.FirstName, c.LastName
14 ORDER BY TotalTransactions DESC;

```




CustomerID	FirstName	LastName	TotalTransactions	AverageSpending
17	Manon	Garnier	696	2967.934138
11	Jean	Lopez	520	2775.964385
14	Michael	Torres	509	3114.436365
2	Benjamin	Diaz	490	2297.348612
19	Lucia	Davis	452	2711.367434
25	Valentina	Noel	444	2244.560000
12	Oliver	Dupont	419	1003.373508
9	Louis	Lemoine	403	1581.714715
7	Charlotte	Johnson	402	3718.539552
4	Pierre	Rodriguez	401	4850.367007
6	Alejandro	Anderson	389	3739.934910
21	Guillaume	Lopez	356	534.627528
3	Sebastien	Chevalier	348	2836.600144
27	Evelyn	Perez	328	2160.302774
30	James	Moreau	321	2939.105171

# Filtres et conditions (WHERE, HAVING, ORDER BY)

```

1 -- Requête pour afficher toutes les transactions dont le total dépensé est supérieur à 100
2
3
4 SELECT t.TransactionID, t.TransactionDate, t.TotalSpent, t.Quantity, t.PaymentMethod, t.Location, t.PricePerUnit,
5 c.CustomerID, c.FirstName, c.LastName
6 FROM transaction t
7 JOIN customer c ON t.CustomerID = c.CustomerID
8 WHERE t.TotalSpent > 100
9 ORDER BY t.TotalSpent DESC;

```



TransactionID	TransactionDate	TotalSpent	Quantity	PaymentMethod	Location	PricePerUnit	CustomerID	FirstName	LastName
3390	2023-12-13	352230.76	5	Credit Card	Online	70446.15	2	Benjamin	Diaz
8479	2023-10-07	337617.55	5	Digital Wallet	Online	67523.51	14	Michael	Torres
3597	2024-02-06	330956.88	5	Cash	In-store	66191.38	20	Camille	Fernandez
7838	2023-06-30	325127.03	5	Cash	Online	65025.41	6	Alejandro	Anderson
6468	2023-12-22	319220.40	5	Credit Card	In-store	63844.08	22	Amelia	Miller
7257	2024-03-18	317197.82	5	Cash	Online	63439.56	30	James	Moreau
1123	2024-05-06	309905.60	5	Credit Card	In-store	61981.12	5	Andres	Mercier
6090	2023-11-24	297403.40	5	Digital Wallet	Mobile App	59480.68	4	Pierre	Rodriguez
5080	2023-01-24	295387.25	5	Cash	In-store	59077.45	26	Juan	Torres
6907	2024-09-20	291770.06	4	Cash	In-store	72942.51	19	Lucia	Davis
5867	2024-11-22	278057.06	4	Digital Wallet	Online	69514.26	5	Andres	Mercier

# Analyse des résultats

- Analyse des résultats
- Les conclusions



Requête	Objectif
1. Transactions détaillées	Voir toutes les transactions avec clients, produits et entreprises
2. Produits les plus vendus	Identifier les articles les plus populaires
3.. Transactions et moyenne des dépenses par client	Analyser le comportement des clients
4. Transactions > 100€	Repérer les achats importants



# Merçi !

Michelle, Oussama, Halima, Allan