



---

DECEMBRE 2021

# RAPPORT FINAL DU PROJET EN DATAWAREHOUSE

Conception, réalisation et analyse  
financière d'une banque en Czech.

PRÉSENTÉ A

Mme. Meryem Rhanoui

PRÉSENTÉ PAR

Hajar Lachheb

# Introduction

## Description du projet

### - Mise en Contexte

Le montant des nouveaux crédits contractés par les familles tchèques s'élève à 1 milliard 700 millions d'euros. Le citoyen tchèque s'est vraiment habitué à changer son mode de vie. Au lieu d'économiser et d'acheter après, il préfère acheter, utiliser et payer après, donc vivre à crédit. Phénomène normal dans la société de consommation, diraient les opposants de tous bords à la mondialisation.

“ In the Czech Republic lives more than 10% people in poverty, which is about 1 million inhabitants”

### - Présentation du projet (Intitulé et résumé)

On se met dans le contexte d'une banque en Tchéquie qui a un grand nombre de clients qui s'inscrivent de manière journalière. La banque a alors une base de données de clients. Mais vu que quand on parle de clientèle, la banque devrait fournir un grand effort pour la fidéliser. Mis à part la fidélisation, la banque devrait aussi être en mesure de connaître les clients VIP et les clients qu'elle devrait plus ou moins éviter pour ne pas entraîner des pertes internes. Les principaux rôles d'une banque sont :

- Garder l'argent
- Offrir des crédits

Lorsque la banque offre des crédits à ses clients, elle prend un grand risque vu qu'elle ne peut pas savoir à 100% si le client va lui rendre son argent ou pas. D'où la nécessité d'avoir une garantie mais ceci ne marche toujours pas, vu que la plupart des banques ont perdu des millions à cause des clients qui n'arrivent pas à payer et fuient le pays.

Dans notre cas, notre projet va nous permettre de prédire si un client est bon ou pas en prenant en considération un grand nombre d'éléments.

## Objectifs (description des principaux objectifs du projet)

On considère que la banque veut améliorer ses services. Par exemple, les directeurs de banque n'ont qu'une vague idée de qui est un bon client (à qui offrir des services supplémentaires) et qui est un mauvais client (à surveiller attentivement pour minimiser les pertes de la banque).

Heureusement, la banque stocke des données sur leurs clients, les comptes (transactions sur plusieurs mois), les prêts déjà accordés, les cartes de crédit émises.

En vain, le but de mon analyse serait de permettre à la banque :

- De prédire un bon client et un mauvais client.

- De prédire quand est qu'un client pourrait devenir mauvais (en prenant en compte des indicateurs)
- De connaître les raisons qui pourraient pousser un client à être mauvais et ne pas payer son crédit.
- D'adapter ses offres aux besoins de ses clients.
- D'adapter ses offres aux conditions économique et sociale de chaque district.

### Analyse et proposition d'indicateurs

| Indicateurs choisis   | Explication de l'indicateur  |
|---|--|
| Le nombre de transaction moyen par ville au niveau de la République Tchèque   | Ceci va nous permettre de connaître les grandes villes et petites villes ainsi que les grandes banques et les petites banques.   |
| Le nombre de "client fidèle"/"mauvais clients" et leur villes respectifs  | On va définir un bon et un mauvais client d'abord en utilisant la capacité du client a retourné le montant du crédit en respectant le délai. Après avoir défini le bon et le mauvais client, il faut faire un tri et voir le nombre de villes ayant un nombre abondant de mauvais clients et celui ayant des clients fidèles et comparer ceci avec le nombre de transaction par ville. |
| Les villes/districts qui ont plus le plus de clients preneurs de crédits (Loans takers)                                   | Les villes qui ont tendance à prendre des crédits sont des petites villes ou de grandes villes ?   |
| Le nombre moyen d'immigrants au niveau des villes/ districts  | Le nombre d'immigrants par ville, ceci pourrait nous donner une indication de plus sur le type de ville et nous prédire si c'est les villes ayant le plus d'immigrants qui prennent le plus de crédits?  |
| Le taux moyen de clients fidèle / mauvais clients preneurs de crédits en prenant en considération le genre et le district | Le genre pourrait aussi définir si le client va réussir à payer le crédit qu'il a pris ou pas. En prenant en considération le district ou la ville, on pourrait savoir d'avantage la ville, le genre et le pourcentage moyen des bons et mauvais clients.  |
| L'âge moyen des clients fidèles/mauvais clients par district  | L'âge est aussi important pour définir la catégorie d'âge qui finit par prendre des crédits et encore une fois définir d'autres indicateurs permettant de préciser si le client est bon ou pas.<br>On prend aussi en considération le district, car on pourrait savoir si la ville est estudiantine d'où le nbr important de mauvais créditeurs ou pas.                                |
| Le nombre moyen des transactions par clients (Clients fidèle vs mauvais clients)  | Est ce que les mauvais clients ont tendance à faire qu'un nombre moindre de transactions en prenant  |

|  |  |
|--|--|
|  | en considération le montant moyen des transactions? What about the good clients ?  |
| La date moyenne d'ouverture du compte qui permet aux clients de prendre un crédit ( Date moyenne d'ouverture du compte des mauvais clients et celui des clients fidèles) | La date d'ouverture des comptes nous permet-elle de définir un bon et un mauvais client? Si le compte est ancien, est ce que la banque considérerait que le client est fidèle ?  |
| L'utilisation de la carte et le type de carte le plus abondant au niveau des clients fidèles et au niveau des mauvais clients  | * Les clients ayant une carte de crédit arrivent-t-il toujours à payer leurs dettes et leur crédit. Encore un autre indicateur me permettant de prédire un bon et un mauvais client. Surtout le type de carte (Golden etc)   |
| Le salaire moyen des clients par district (en prenant en considération les indicateurs économiques tel que l'activité mere au niveau de chaque foyer etc)                | Le salaire moyen et la situation familiale des familles au niveau du district pourrait nous indiquer si le non paiement du crédit est aussi lies au salaire moyen et aux indicateurs économiques du district tel que l'activité mere ( Agriculture, usinage etc) - Je pourrais aussi diviser les familles en trois classes (Pauvre, Moyenne, Riches) |
| Les clients utilisant une carte en fonction de son âge, sa ville et la somme d'argent totale qu'il a déjà au niveau de son compte  | Est ce que les personnes ayant une carte de crédit sont des personnes appartenant à une ville X, à un âge Y et ont une somme d'argent Z. Ces indicateurs vont me permettre davantage de connaître le type du client surtout après le résultat de *   |
| Le nombre moyen des mauvais clients ayant un foyer $\geq 2$ et leur district respectif   | Les foyers dont le nombre est supérieur ou égale à 2 sont les foyers n'arrivent toujours pas à payer leur dette et est ce que ceci dépend aussi du district ou de la ville (un foyer de 4 qui vit dans la capitale du pays arrive toujours à payer le crédit alors qu'au niveau d'une autre ville, ce n'est pas le même cas).                        |
| Le pourcentage du nombre de clients (Mauvais et fidèles) des banques par rapport au nombre total des habitants de la district/ville                                      | L'effectif des mauvais/fidèles clients par rapport au nombre total des habitants du district.  |
| Le nombre de personnes non employées par région en prenant en la comparant avec la ville/région/district qui contient le plus de mauvais clients.                        | Ce résultat pourrait nous donner une autre motivation qui va pousser un mauvais client à prendre un crédit, un autre indicateur que la banque devrait prendre en compte.   |
| Le classement des dix premiers aliments consommables les plus chers au niveau du pays en prenant en compte le taux d'inflation.  | Ces indicateurs pourraient nous donner davantage motivation poussant le client à prendre des crédits. Cet indicateur est à prendre en considération par la banque.   |

|   |   |
|---|---|
| Les clients (Mauvais ou fidele) sont plus des créditeurs ou débiteurs en prenant en considération le taux moyen du montant de transaction par mois ou semestre. | Les mauvais clients ont tendance à retirer plus d'argent que de débiter de l'argent et s'il débitent, qu'elle est le nombre moyen du montant de transaction par mois ou semestre. |
|---|---|

### Dataset Utilisée

Dans mon projet, j'utilise deux datasets. La première dataset est une dataset qui contient 8 tables. Les tables sont diversifiées et contiennent des informations "réelles" à propos d'une banque.

**La première dataset contient 5 tables :** Comptes, Clients, Disposition, Ordre de paiement, Transaction, Crédit, Carte de crédit, District. L'ensemble de données au niveau de ce dataset est une collection d'informations financières d'une banque tchèque.

- L'ensemble de données concerne plus de 5 300 clients bancaires avec environ 1 000 000 de transactions. De plus, la banque représentée dans l'ensemble de données a accordé près de 700 prêts et émis près de 900 cartes de crédit, qui sont toutes représentées dans les données.

<https://data.world/lpetrocelli/czech-financial-dataset-real-anonymized-transactions>

**- relation compte (4500 objets dans le fichier ACCOUNT.ASC)** - chaque enregistrement décrit les caractéristiques statiques d'un compte,

**-relation client (5369 objets dans le fichier CLIENT.ASC)** - chaque enregistrement décrit les caractéristiques d'un client,

**-disposition de la relation (5369 objets dans le fichier DISP.ASC)** - chaque enregistrement met en relation un client avec un compte c'est-à-dire que cette relation décrit les droits des clients à exploiter des comptes,

**-relation ordre permanent (6471 objets dans le fichier ORDER.ASC)** - chaque enregistrement décrit les caractéristiques d'un ordre de paiement,

**-transaction de relation (1056320 objets dans le fichier TRANS.ASC)** - chaque enregistrement décrit une transaction sur un compte,

**- prêt relationnel (682 objets dans le fichier LOAN.ASC)** - chaque enregistrement décrit un prêt accordé pour un compte donné,

**-relation carte de crédit (892 objets dans le fichier CARD.ASC)** - chaque enregistrement décrit une carte de crédit émise sur un compte,

**-relation données démographiques (77 objets dans le fichier DISTRICT.ASC)** - chaque fiche décrit les caractéristiques démographiques d'un quartier. Cette dimension va être divisée en deux sous dimensions : REGIONS et REGIONSANALYSIS. Ces tables font partie de la dimension DISTRICTS et ceci pour faciliter l'analyse.

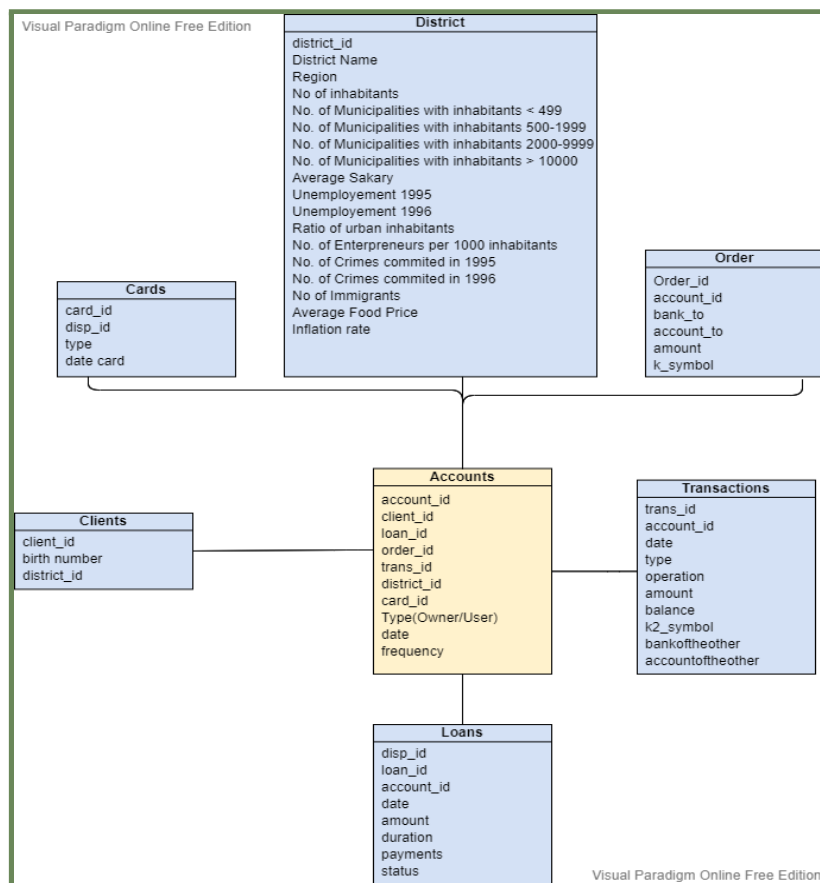
**La deuxième dataset** / table contient des indicateurs utilisés pour détailler plus notre dataset d'origine. Cette dataset contient des moyennes et des sommes et constituerait ainsi notre table de faits.

### Outils utilisés pour notre projet

- Python
- Microsoft SQL Server Management Studio
- Microsoft SQL Server Integration Services
- Jupyter Notebook

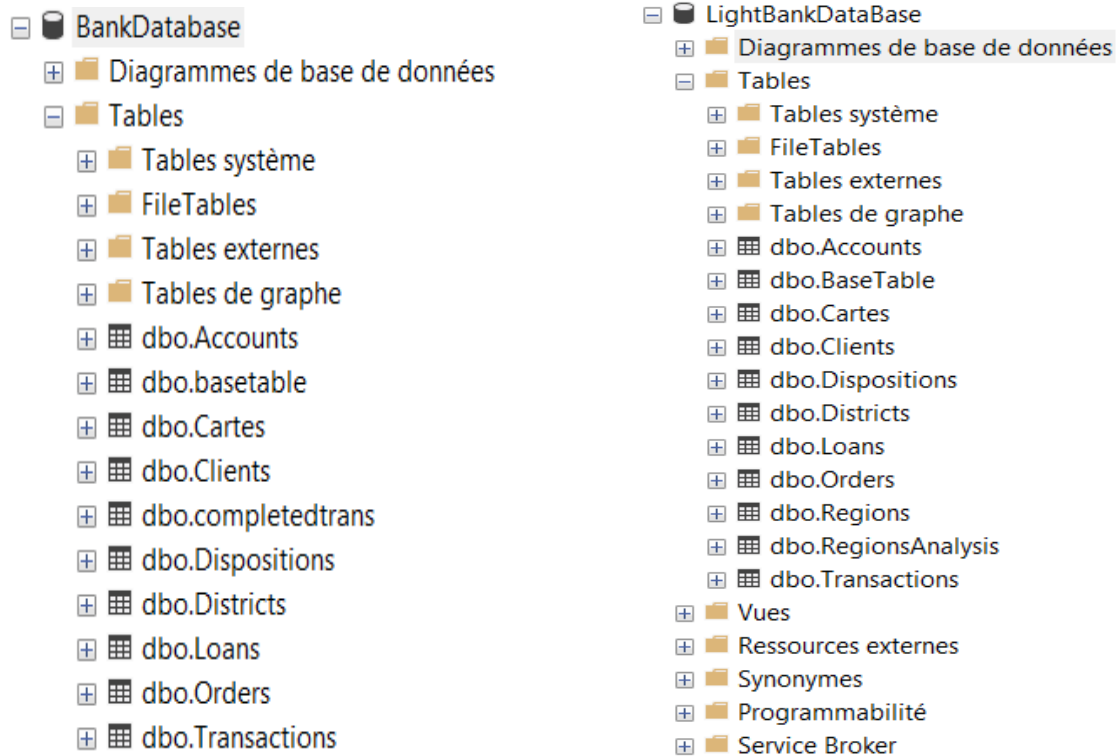
## Conception

### - Première conception du DataWarehouse



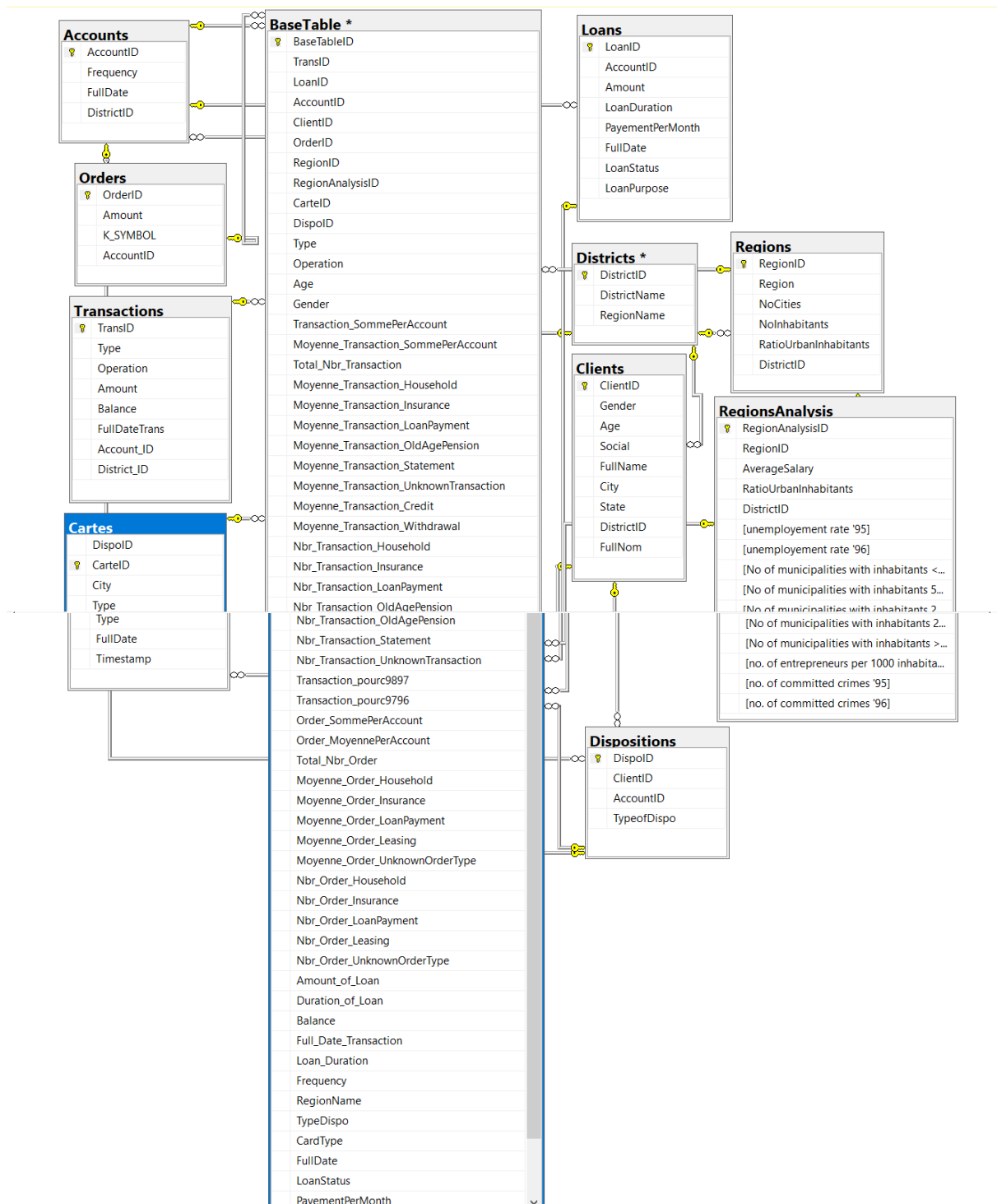
## - Création du DataWarehouse

Pour créer mon DataWarehouse, nous avons commencé par créer un Database dont le nom est : LightBankDataBase. Et nous avons créé par la suite les tables correspondantes tout en précisant les bons types de données et les colonnes convenables. (On a choisi que les colonnes dont on aura besoin au niveau de notre analyse).



Nous allons alors mettre en œuvre l'entrepôt de données sur SSMS sous le nom LightBankDataBase, nous utiliserons un modèle en flocon principalement mais nous tâchons à ce que les autres dimensions aient des clés secondaires afin d'assurer une bonne liaison entre les différentes dimensions.

Les attributs de chaque tableau sont représentés sur le diagramme ci-dessous.



## Collecte et préparation des données

Pour la collecte et la préparation des données, nous avons tachées d'effectuer des recherches pendant des semaines afin de trouver l'ensemble des données les plus adéquates pour effectuer l'analyse financière tant voulue.

Le résultat était alors un dataset qui était déjà nettoyé et préparé par la banque en question mais encore le dataset contient des données réelles des clients d'où la pertinence de ce dernier.



La pipeline ETL va nous aider à homogénéiser les données avec lesquelles nous allons travailler tout en effectuant des changements tel que la conversion de données, l'ajout des clés secondaire pour lier plus les tables etc

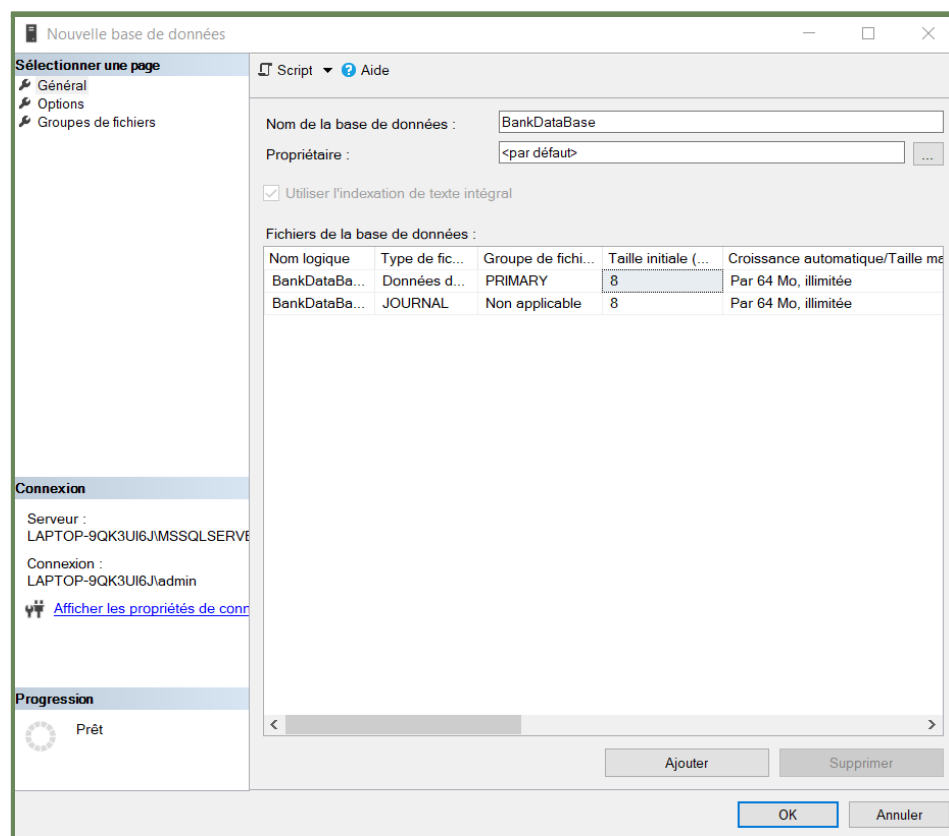
## Préparation de l'environnement de travail

Avant de commencer à utiliser Visual Studio afin de créer notre pipeline ETL. Il fallait préparer l'environnement de travail dont nous aurons besoin. Qui dit environnement dit la source principale de nos données devrait être bel et bien existante et la destination devrait être préparée.

Dans notre cas, nous allons commencer par créer un DataBase initial qui va contenir les datasets dont on aura besoin.

### 1- Création de la DataBase BankDataBase qui va contenir les données sources

La création de la DataBase "BankDataBase" se fait en utilisant le SQL Server et en cliquant sur la rubrique "Nouvelle base de données", il faut s'assurer que nous avons entreteenu une bonne connexion avec notre serveur et que le test fonctionne bien.



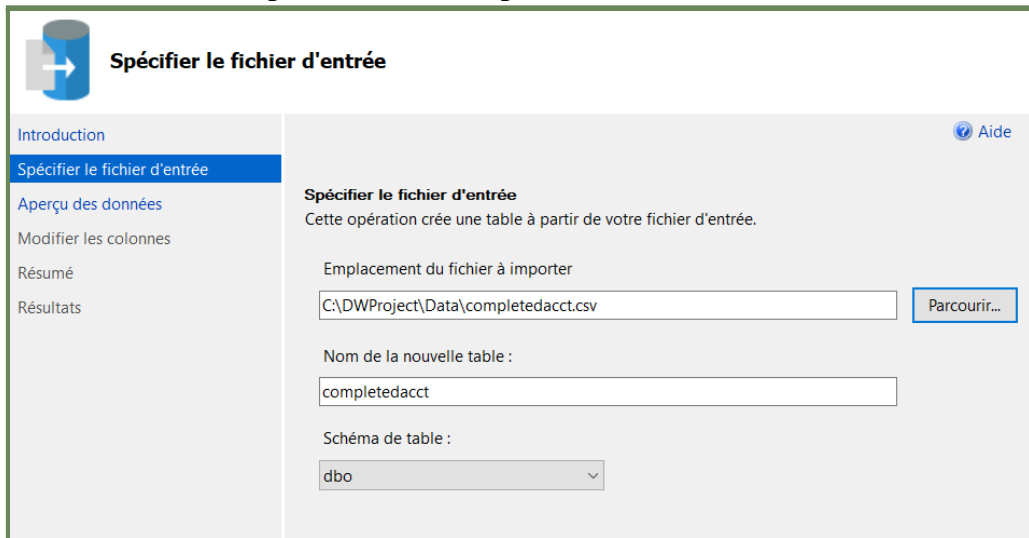
Le résultat de la création de la base de données et le suivant, ça nous affiche les dossiers tels que Tables et diagrammes de base de données.

- [-] BankDatabase
  - [+] Diagrammes de base de données
  - [+] Tables
  - [+] Tables système
  - [+] FileTables

## 2-Chargement des données au niveau de SQL Server

La deuxième étape serait de charger les données qu'on a au niveau de notre base de données BankDataBase. Le chargement se fait en utilisant la section "importer un fichier plat". Dans notre cas, nos fichiers sont sous format .CSV donc en incluant chaque table, on précise le type de données pour chaque attributs aussi même s'ils sont affichés automatiquement.

- 
- **dbo.Accounts : Importation de completedacct**

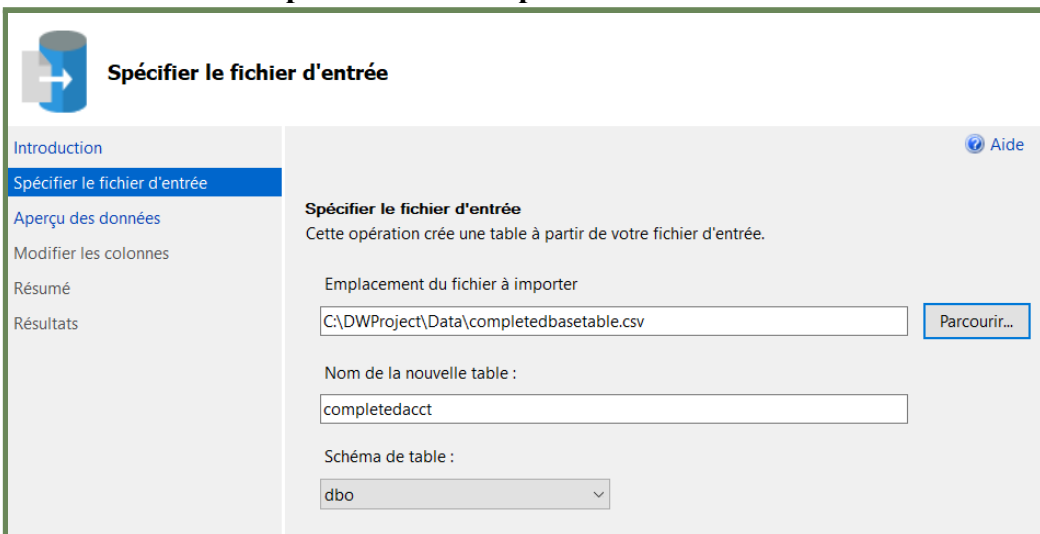


The screenshot shows the 'Spécifier le fichier d'entrée' (Specify input file) dialog box in SQL Server Data Tools. The dialog has a left sidebar with navigation links: 'Introduction', 'Spécifier le fichier d'entrée' (selected), 'Aperçu des données', 'Modifier les colonnes', 'Résumé', and 'Résultats'. The main area is titled 'Spécifier le fichier d'entrée' and contains the following fields:

- Emplacement du fichier à importer:** A text box containing 'C:\DWProject\Data\completedacct.csv' and a 'Parcourir...' (Browse...) button.
- Nom de la nouvelle table :** A text box containing 'completedacct'.
- Schéma de table :** A dropdown menu with 'dbo' selected.

An 'Aide' (Help) link is visible in the top right corner.

- **dbo.BaseTable : Importation de completedbasetable**




The screenshot shows the 'Spécifier le fichier d'entrée' (Specify input file) dialog box in SQL Server Data Tools. The dialog has a left sidebar with navigation links: 'Introduction', 'Spécifier le fichier d'entrée' (selected), 'Aperçu des données', 'Modifier les colonnes', 'Résumé', and 'Résultats'. The main area is titled 'Spécifier le fichier d'entrée' and contains the following fields:

- Emplacement du fichier à importer:** A text box containing 'C:\DWProject\Data\completedbasetable.csv' and a 'Parcourir...' (Browse...) button.
- Nom de la nouvelle table :** A text box containing 'completedacct'.
- Schéma de table :** A dropdown menu with 'dbo' selected.

An 'Aide' (Help) link is visible in the top right corner.

- **dbo.Cartes : Importation de completedcard**



## Spécifier le fichier d'entrée

Introduction

**Spécifier le fichier d'entrée**

Aperçu des données

Modifier les colonnes

Résumé

Résultats

[Aide](#)


**Spécifier le fichier d'entrée**  
Cette opération crée une table à partir de votre fichier d'entrée.

Emplacement du fichier à importer

Nom de la nouvelle table :

Schéma de table :

- **dbo.Clients : Importation de completedclient**



## Spécifier le fichier d'entrée

Introduction

**Spécifier le fichier d'entrée**

Aperçu des données

Modifier les colonnes

Résumé

Résultats

[Aide](#)


**Spécifier le fichier d'entrée**  
Cette opération crée une table à partir de votre fichier d'entrée.

Emplacement du fichier à importer

Nom de la nouvelle table :

Schéma de table :

- **dbo.Dispositions : Importation de completeddisposition**



## Spécifier le fichier d'entrée

Introduction

**Spécifier le fichier d'entrée**

Aperçu des données

Modifier les colonnes

Résumé

Résultats

[Aide](#)


**Spécifier le fichier d'entrée**  
Cette opération crée une table à partir de votre fichier d'entrée.

Emplacement du fichier à importer

Nom de la nouvelle table :

Schéma de table :

- **dbo.Loans : Importation de completedloan**



## Spécifier le fichier d'entrée

- Introduction
- Spécifier le fichier d'entrée**
- Aperçu des données
- Modifier les colonnes
- Résumé
- Résultats

[Aide](#)

### Spécifier le fichier d'entrée

Cette opération crée une table à partir de votre fichier d'entrée.


Emplacement du fichier à importer



Nom de la nouvelle table :

Schéma de table :

- **dbo.Orders : Importation de completedorder**



## Spécifier le fichier d'entrée

- Introduction
- Spécifier le fichier d'entrée**
- Aperçu des données
- Modifier les colonnes
- Résumé
- Résultats

[Aide](#)

### Spécifier le fichier d'entrée

Cette opération crée une table à partir de votre fichier d'entrée.


Emplacement du fichier à importer



Nom de la nouvelle table :

Schéma de table :

- **dbo.Districts : Importation de completeddistrict**



## Spécifier le fichier d'entrée

- Introduction
- Spécifier le fichier d'entrée**
- Aperçu des données
- Modifier les colonnes
- Résumé
- Résultats

[Aide](#)

### Spécifier le fichier d'entrée

Cette opération crée une table à partir de votre fichier d'entrée.

Emplacement du fichier à importer



Nom de la nouvelle table :

Schéma de table :

- **dbo.Transactions : Importation de completedtrans**

**Spécifier le fichier d'entrée**

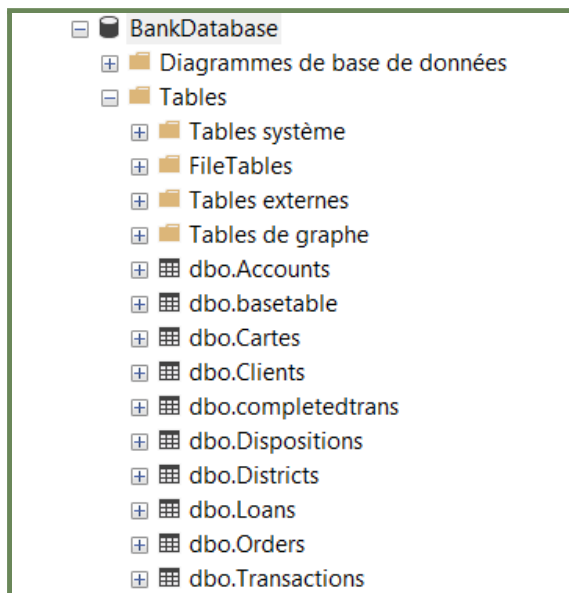
Introduction | **Spécifier le fichier d'entrée** | Aperçu des données | Modifier les colonnes | Résumé | Résultats

**Spécifier le fichier d'entrée**  
Cette opération crée une table à partir de votre fichier d'entrée.

Emplacement du fichier à importer  
 [Parcourir...](#)

Nom de la nouvelle table :

Schéma de table :



→ Après avoir réussi toutes les étapes de l'importation pour chaque table, on reçoit un message que l'importation est réussie. Dans certains cas où le type est incompatible, on reçoit un message qui nous montre que l'importation a échoué vu que le type est incompatible.

→ Nous aurons comme résultat final le résultat suivant. Notre DataBase est rempli de tables importées ci-dessus. On pourrait vérifier s'ils sont importés correctement en utilisant une requête SQL qui permet d'afficher les 200 premières lignes.

### 3- Création de la DataBase LightBankDataBase qui va contenir les données de la destination

Notre DataBase source est maintenant créé, il est temps de créer notre DataBase Destination qui va contenir les données résultantes de notre Pipeline ETL. Après avoir créé un DataBase dont le nom est "LightBankDataBase", il faut créer les tables suivantes en utilisant des requêtes SQL "UPDATE".

#### Table "District" :

```
CREATE TABLE Districts (
  DistrictID tinyint NOT NULL,
  DistrictName NVARCHAR(50),
  RegionName NVARCHAR(50),
  PRIMARY KEY (DistrictID),)
```

### Table “Regions”:

```
CREATE TABLE Regions (  
  RegionID NVARCHAR(50),  
  Region NVARCHAR(50),  
  NoCities tinyint,  
  NoInhabitants int,  
  RatioUrbanInhabitants FLOAT,  
  DistrictID tinyint,  
  PRIMARY KEY (RegionID),  
  FOREIGN KEY (DistrictID) REFERENCES Districts(DistrictID))
```

### Table “RegionsAnalysis” :

```
CREATE TABLE RegionsAnalysis (  
  RegionAnalysisID INTEGER NOT NULL,  
  RegionID NVARCHAR(50),  
  AverageSalary smallint,  
  RatioUrbanInhabitants FLOAT,  
  DistrictID tinyint,  
  [unemployment rate '95] float,  
  [unemployment rate '96] float,  
  [No of municipalities with inhabitants <499] tinyint,  
  [No of municipalities with inhabitants 500 - 1999] tinyint,  
  [No of municipalities with inhabitants 2000- 9999] tinyint,  
  [No of municipalities with inhabitants > 10000] tinyint,  
  [no. of entrepreneurs per 1000 inhabitants] tinyint,  
  [no. of committed crimes '95] int,  
  [no. of committed crimes '96] int,  
  PRIMARY KEY (RegionAnalysisID),  
  FOREIGN KEY (RegionID) REFERENCES Regions(RegionID))
```

### Table “Accounts” :

```
CREATE TABLE Accounts (  
  AccountID NVARCHAR(50),  
  Frequency NVARCHAR(50),  
  FullDate DATE,  
  DistrictID tinyint,  
  PRIMARY KEY (AccountID),  
  FOREIGN KEY (DistrictID) REFERENCES Districts(DistrictID))
```

### Table “Clients” :

```
CREATE TABLE Clients (  
  ClientID NVARCHAR(50) NOT NULL,  
  Gender NVARCHAR(50),  
  Age tinyint,  
  Social NVARCHAR(50),  
  FullName NVARCHAR(50),  
  City NVARCHAR(50),  
  State NVARCHAR(50),  
  DistrictID tinyint,  
  PRIMARY KEY (ClientID),  
  FOREIGN KEY (DistrictID) REFERENCES Districts(DistrictID))
```

### Table “Dispositions” :

```
CREATE TABLE Dispositions (  
  DispoID MONEY NOT NULL,  
  ClientID NVARCHAR(50),  
  AccountID NVARCHAR(50),  
  PRIMARY KEY (DispoID),  
  FOREIGN KEY (ClientID) REFERENCES Clients(ClientID),  
  FOREIGN KEY (AccountID) REFERENCES Accounts(AccountID))
```

#### Table “Cartes” :

```
CREATE TABLE Cartes (  
  DispoID MONEY NOT NULL,  
  CarteID NVARCHAR(50),  
  City NVARCHAR(50),  
  Type NVARCHAR(50),  
  FullDate DATE,  
  NoDays int,  
  PRIMARY KEY (CarteID),  
  FOREIGN KEY (DispoID) REFERENCES Dispositions(DispoID))
```

#### Table “Loans” :

```
CREATE TABLE Loans (  
  LoanID MONEY NOT NULL,  
  AccountID NVARCHAR(50),  
  Amount int,  
  LoanDuration tinyint,  
  PayementPerMonth smallint,  
  FullDate DATE,  
  LoanStatus NVARCHAR(50),  
  LoanPurpose NVARCHAR(50),  
  PRIMARY KEY (LoanID),  
  FOREIGN KEY (AccountID) REFERENCES Accounts(AccountID))
```

#### Table “Orders” :

```
CREATE TABLE Orders (  
  OrderID INTEGER NOT NULL,  
  Amount FLOAT,  
  K_SYMBOL NVARCHAR(50),  
  PRIMARY KEY (OrderID),  
  AccountID NVARCHAR(50),  
  FOREIGN KEY (AccountID) REFERENCES Accounts(AccountID))
```

#### Table “Transactions” :

```
CREATE TABLE Transactions (  
  TransID NVARCHAR(50) NOT NULL,  
  AccountID NVARCHAR(50),  
  DistrictID tinyint,  
  Type NVARCHAR(50),  
  Operation NVARCHAR(50),  
  Amount FLOAT,  
  Balance FLOAT,  
  FullDateTrans DATE,  
  PRIMARY KEY (TransID))
```

Table “BaseTable” :

```
CREATE TABLE BaseTable (  
BaseTableID smallint NOT NULL,  
TransID NVARCHAR(50),  
LoanID MONEY,  
AccountID NVARCHAR(50),  
DistrictID tinyint,  
ClientID NVARCHAR(50),  
OrderID INT,  
RegionID NVARCHAR(50),  
RegionAnalysisID int,  
CarteID NVARCHAR(50),  
DispoID MONEY,  
Type NVARCHAR(50),  
Operation NVARCHAR(50),  
Age FLOAT,  
Gender NVARCHAR(50),  
Transaction_SommePerAccount FLOAT,  
Moyenne_Transaction_SommePerAccount FLOAT,  
Total_Nbr_Transaction FLOAT,  
Moyenne_Transaction_Household FLOAT,  
Moyenne_Transaction_Insurance FLOAT,  
Moyenne_Transaction_LoanPayment FLOAT,  
Moyenne_Transaction_OldAgePension FLOAT,  
Moyenne_Transaction_Statement FLOAT,  
Moyenne_Transaction_UnknownTransaction FLOAT,  
Moyenne_Transaction_Credit FLOAT,  
Moyenne_Transaction_Withdrawal FLOAT,  
Nbr_Transaction_Household FLOAT,  
Nbr_Transaction_Insurance FLOAT,  
Nbr_Transaction_LoanPayment FLOAT,  
Nbr_Transaction_OldAgePension FLOAT,  
Nbr_Transaction_Statement FLOAT,  
Nbr_Transaction_UnknownTransaction FLOAT,  
Transaction_pourc9897 FLOAT,  
Transaction_pourc9796 FLOAT,  
Order_SommePerAccount FLOAT,  
Order_MoyennePerAccount FLOAT,  
Total_Nbr_Order FLOAT,  
Moyenne_Order_Household FLOAT,  
Moyenne_Order_Insurance FLOAT,  
Moyenne_Order_LoanPayment FLOAT,  
Moyenne_Order_Leasing FLOAT,  
Moyenne_Order_UnknownOrderType FLOAT,  
Nbr_Order_Household FLOAT,  
Nbr_Order_Insurance FLOAT,  
Nbr_Order_LoanPayment FLOAT,  
Nbr_Order_Leasing FLOAT,  
Nbr_Order_UnknownOrderType FLOAT,  
Amount_of_Loan FLOAT,  
Duration_of_Loan FLOAT,
```



```

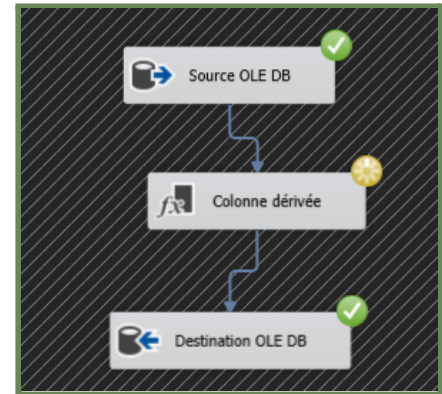
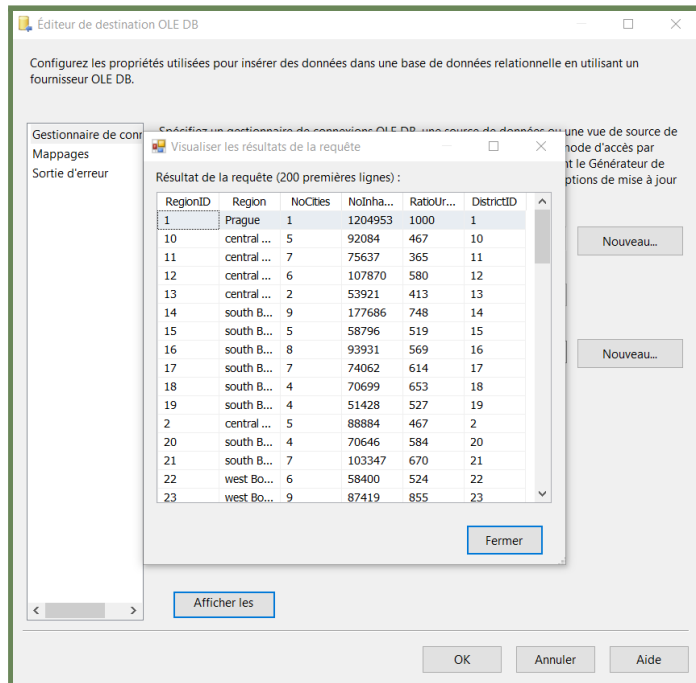
Balance FLOAT,
Full_Date_Transaction DATE,
Loan_Duration tinyint,
Frequency NVARCHAR(50),
RegionName NVARCHAR(50),
TypeDispo NVARCHAR(50),
CardType NVARCHAR(50),
Payement_Per_Month smallint,
FullDate DATE,
LoanStatus NVARCHAR(50),
PRIMARY KEY (BaseTableID),
FOREIGN KEY (TransID) REFERENCES Transactions(TransID),
FOREIGN KEY (AccountID) REFERENCES Accounts(AccountID),
FOREIGN KEY (DispoID) REFERENCES Dispositions(DispoID),
FOREIGN KEY (LoanID) REFERENCES Loans(LoanID),
FOREIGN KEY (DistrictID) REFERENCES Districts(DistrictID),
FOREIGN KEY (RegionID) REFERENCES Regions(RegionID),
FOREIGN KEY (RegionAnalysisID) REFERENCES RegionsAnalysis(RegionAnalysisID),
FOREIGN KEY (CarteID) REFERENCES Cartes(CarteID),
FOREIGN KEY (ClientID) REFERENCES Clients(ClientID),
FOREIGN KEY (OrderID) REFERENCES Orders(OrderID),
)

```

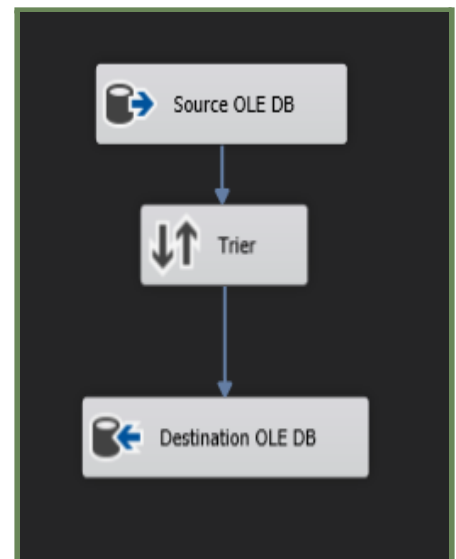
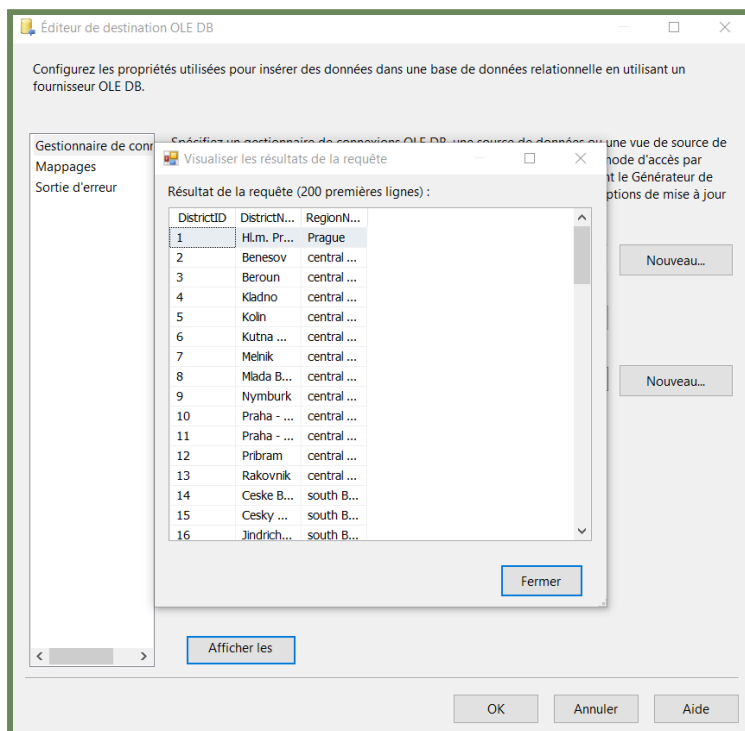
## ETL Pipeline

L'objectif de notre ETL Pipeline, c'est être capable de produire un ensemble de tables qui vont nous aider à faire les bonnes analyses et à prendre des décisions. Notre Dataset a été originalement nettoyé par la banque en question donc notre objectif serait de la transformer en des données pertinentes et liées entre elles nous permettant ainsi d'atteindre nos objectifs en termes d'analyse.

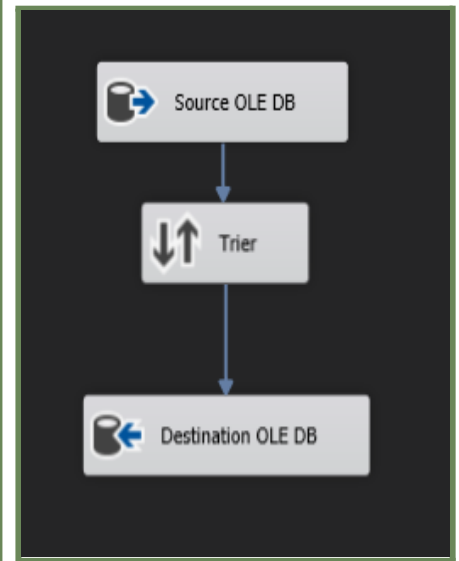
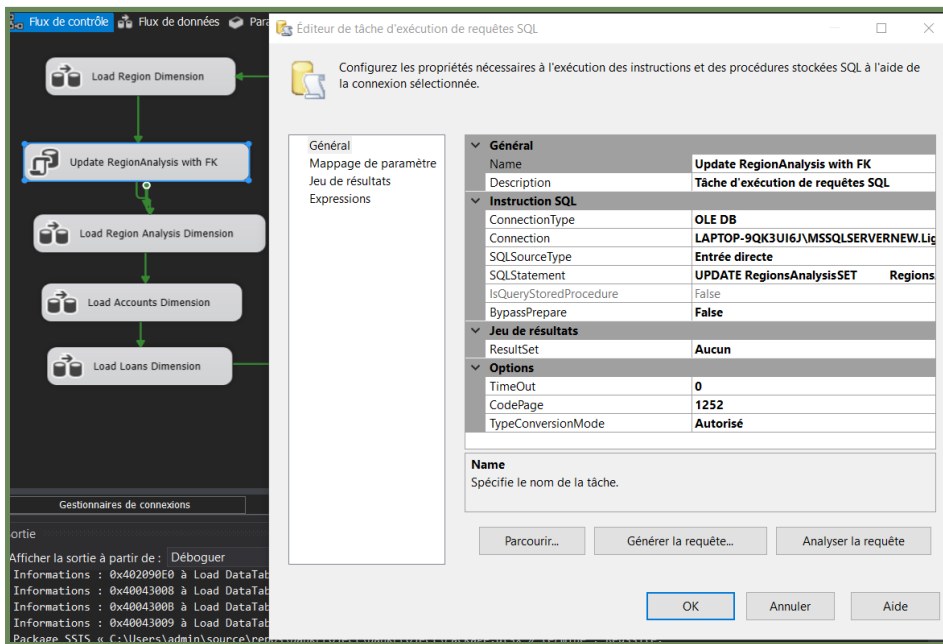
## Pour la Dimension “Region” :



## Pour la Dimension “District” :



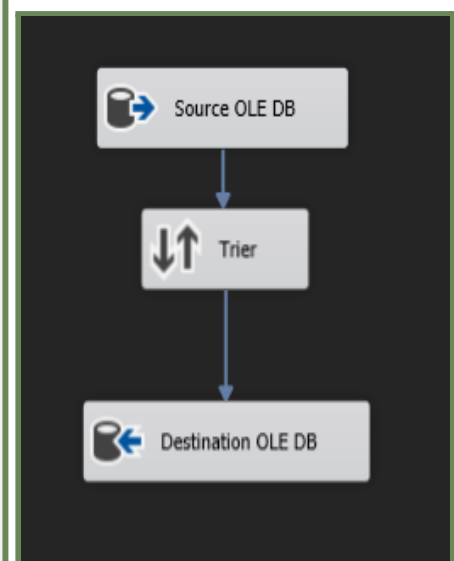
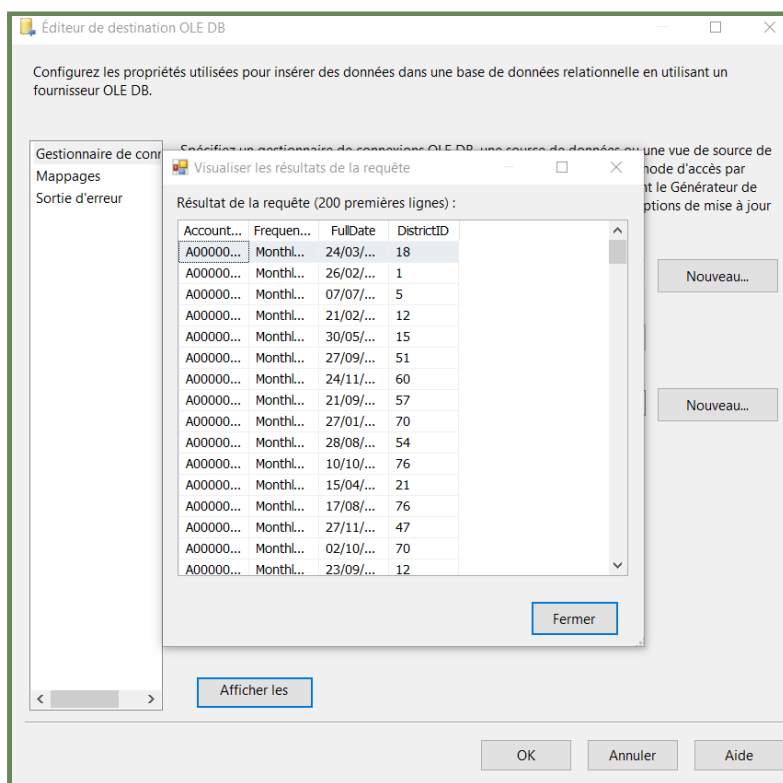
## Pour la Dimension “RegionsAnalysis” :



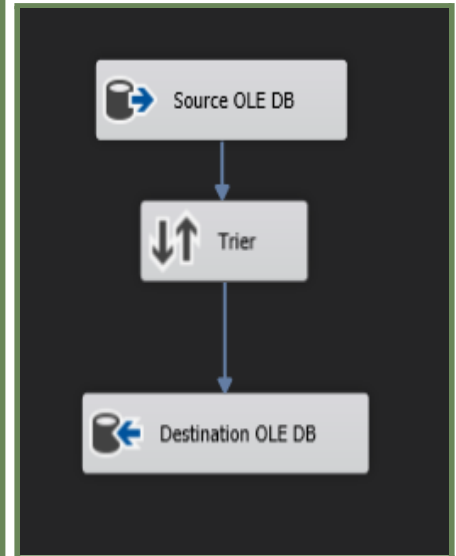
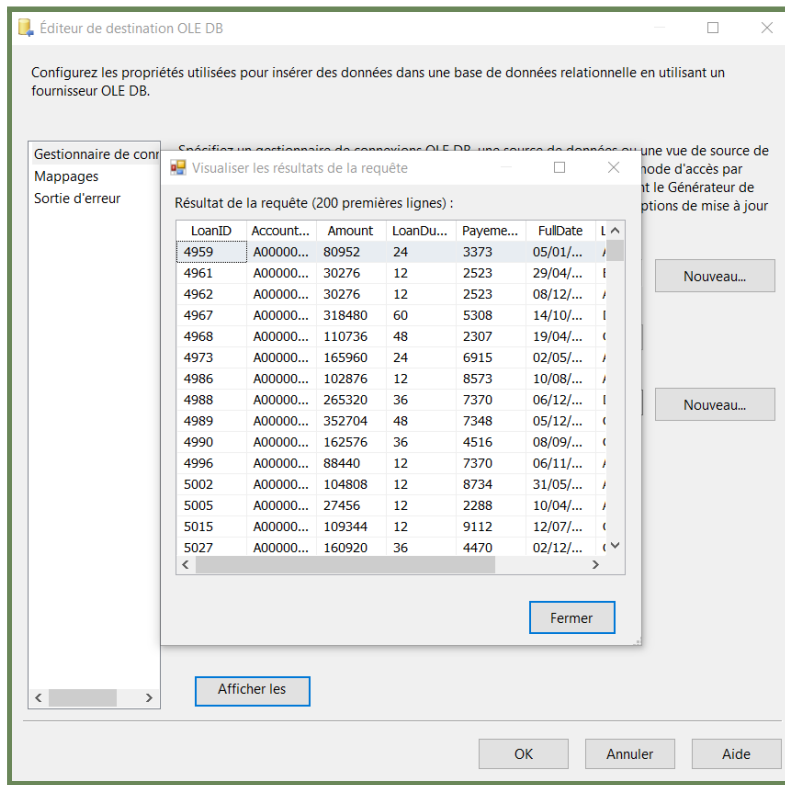
```

UPDATE RegionsAnalysis
SET      RegionsAnalysis.RegionID = Regions.RegionID
FROM      RegionsAnalysis
INNER JOIN Regions ON RegionsAnalysis.DistrictID = Regions.DistrictID
  
```

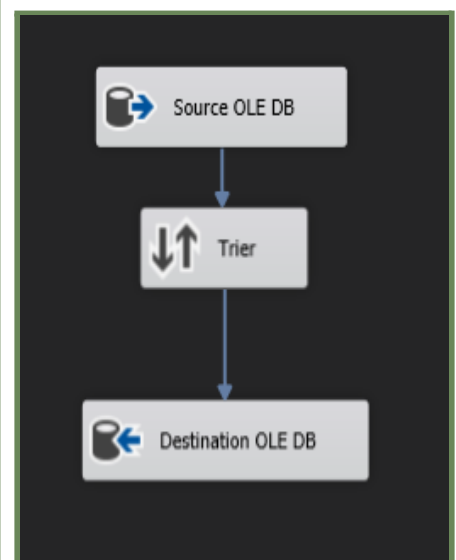
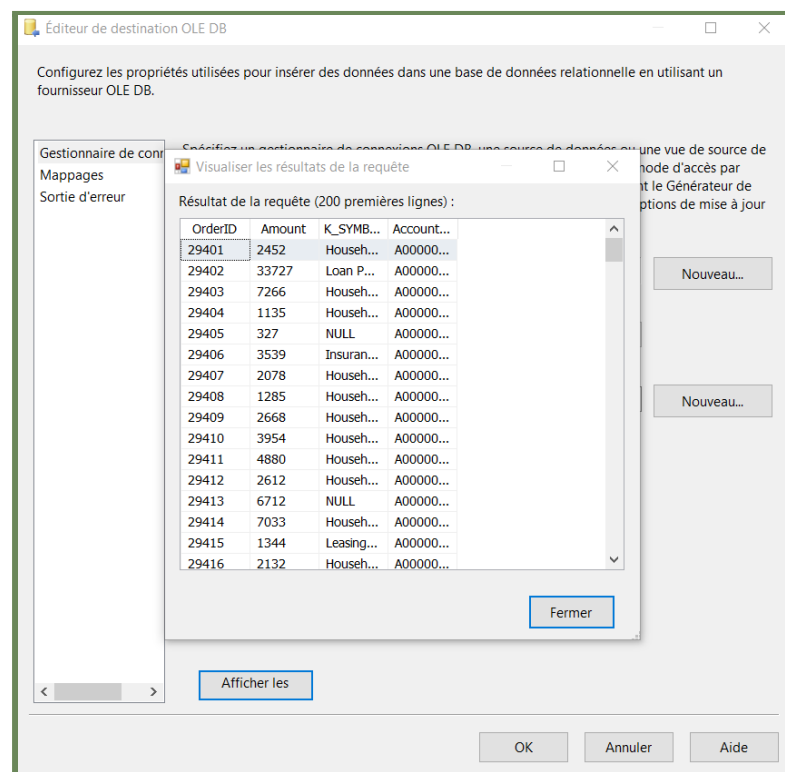
## Pour la Dimension “Accounts” :



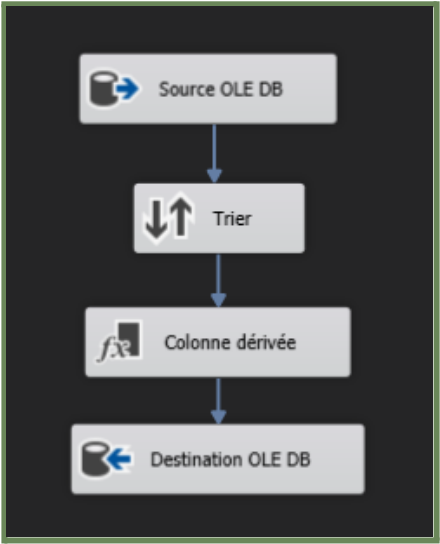
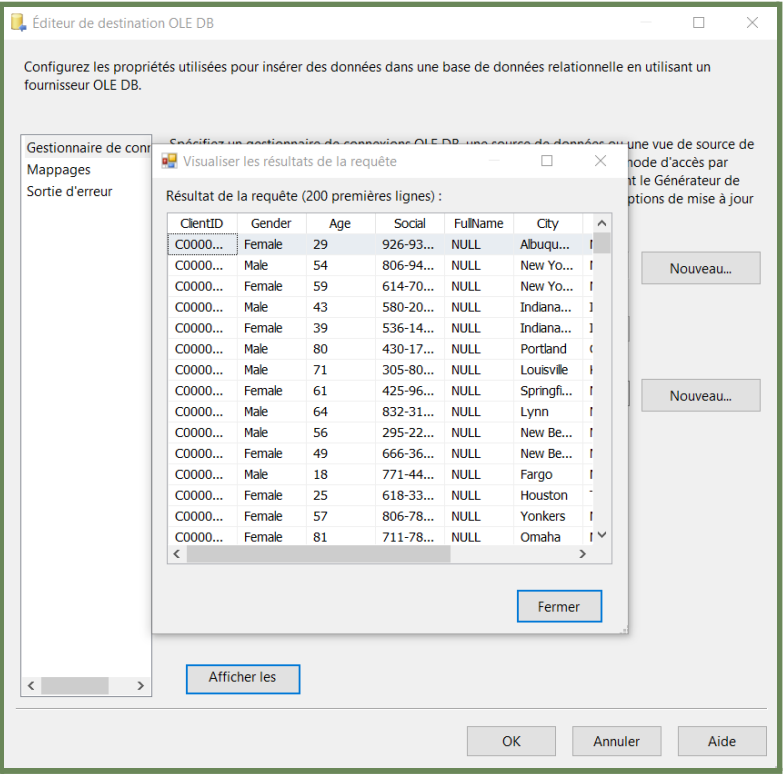
## Pour la Dimension “Loans” :



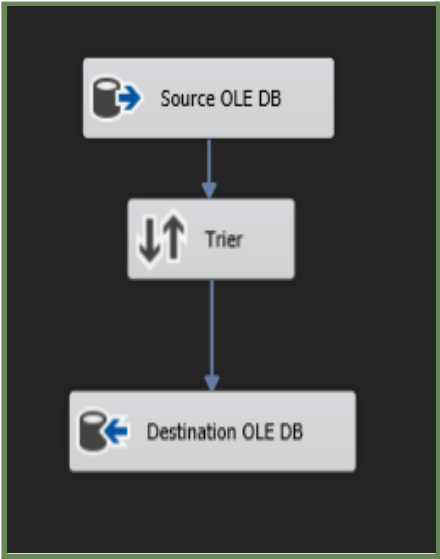
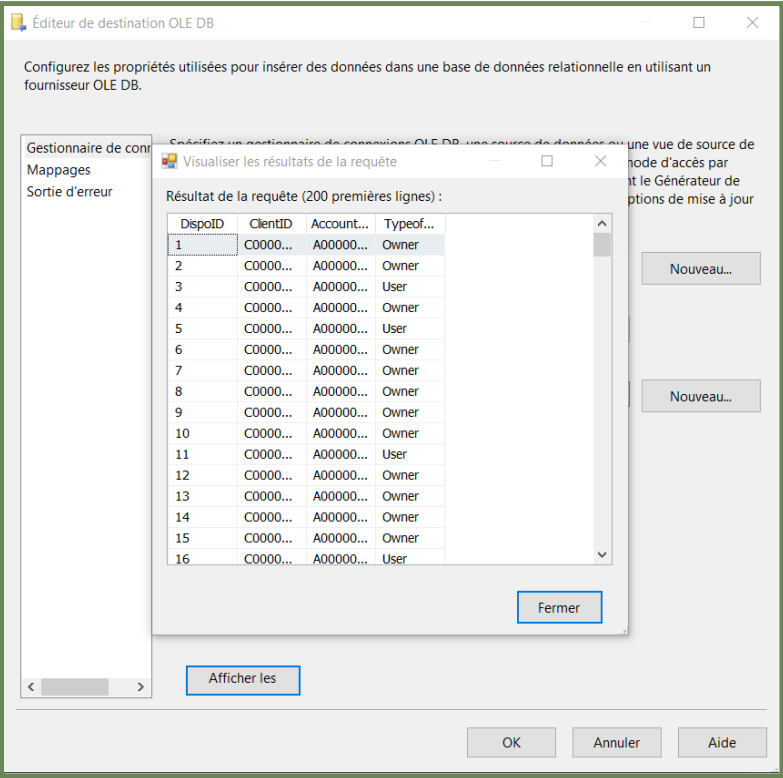
## Pour la Dimension “Orders” :



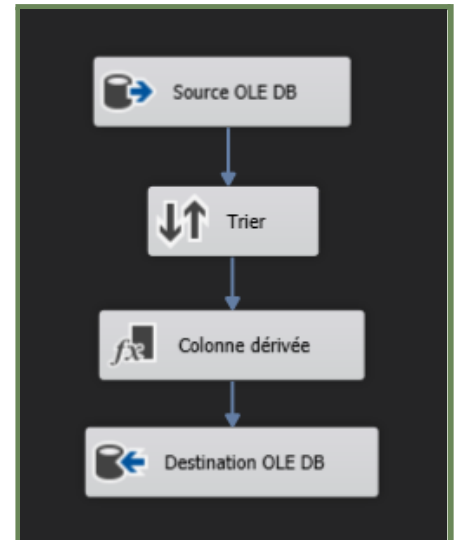
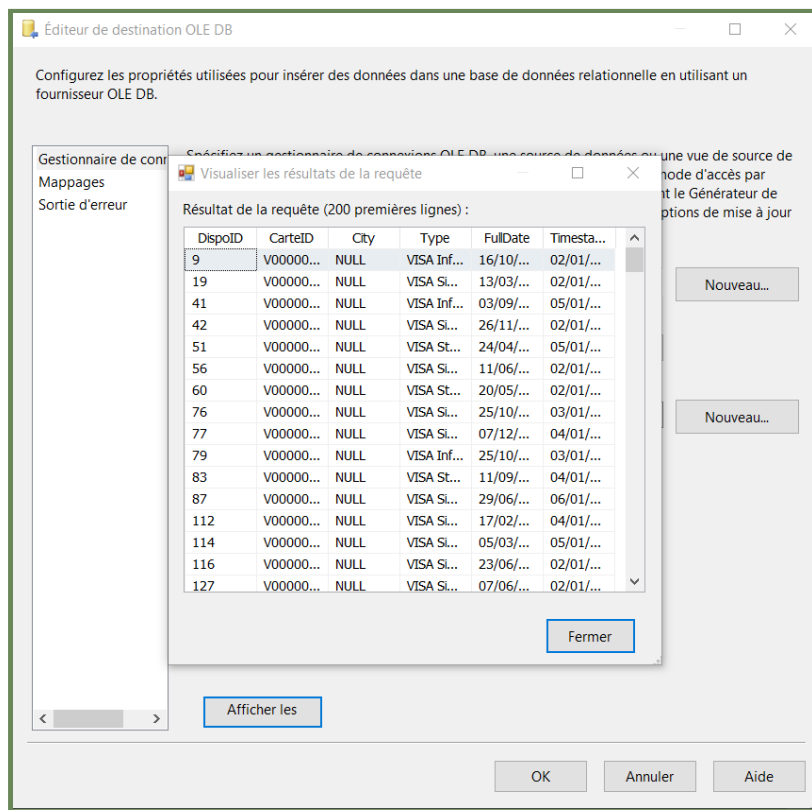
Pour la Dimension “Clients” :



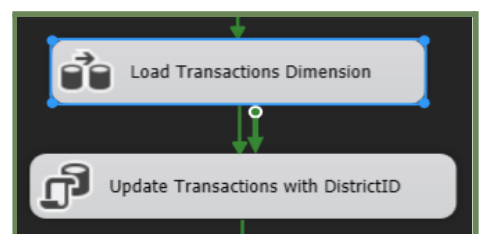
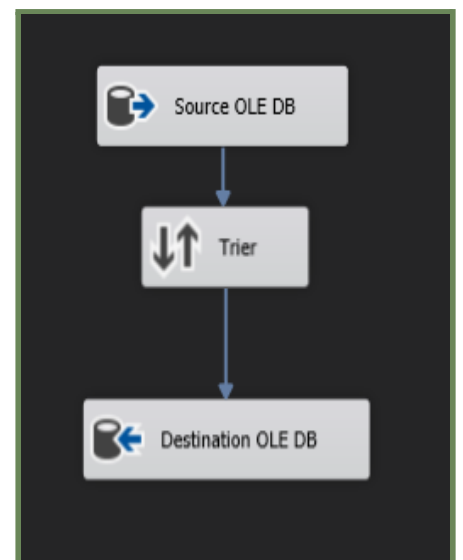
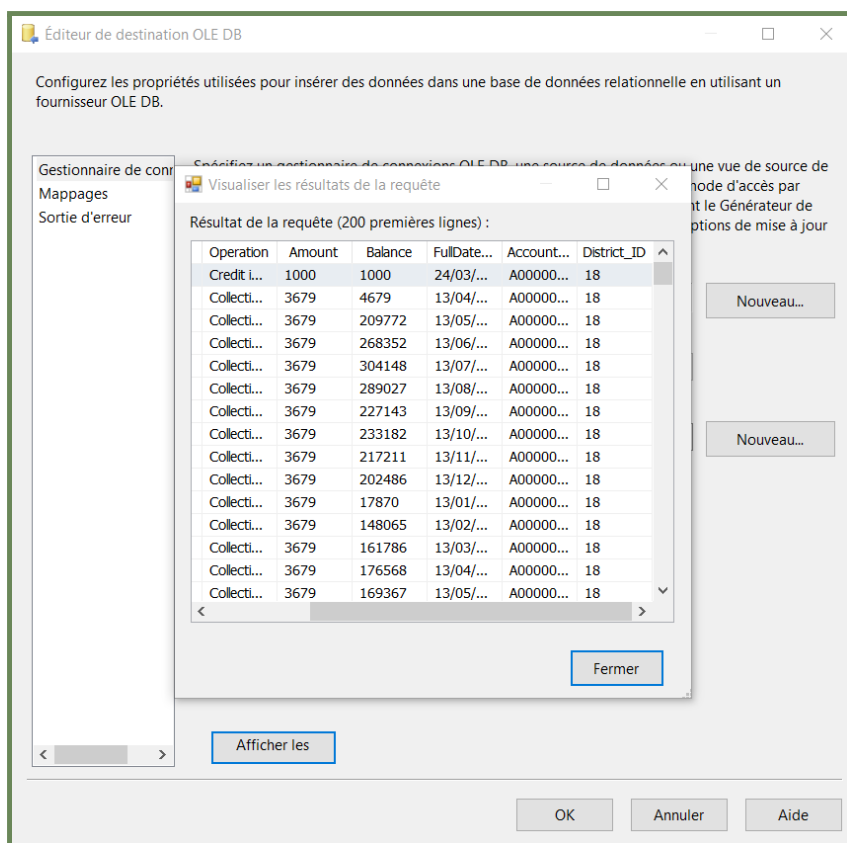
Pour la Dimension “Dispositions” :



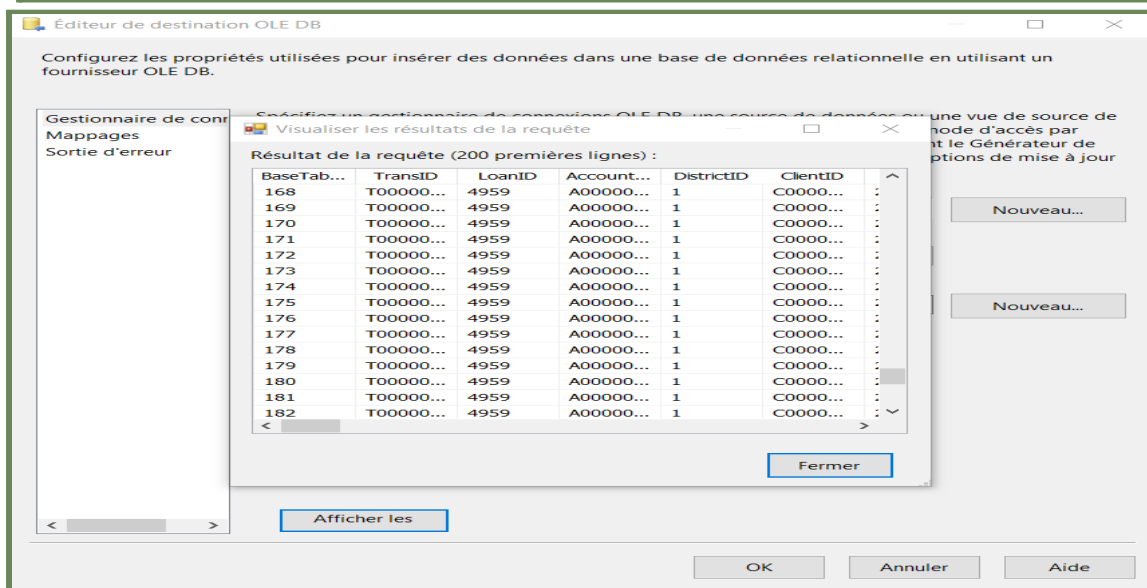
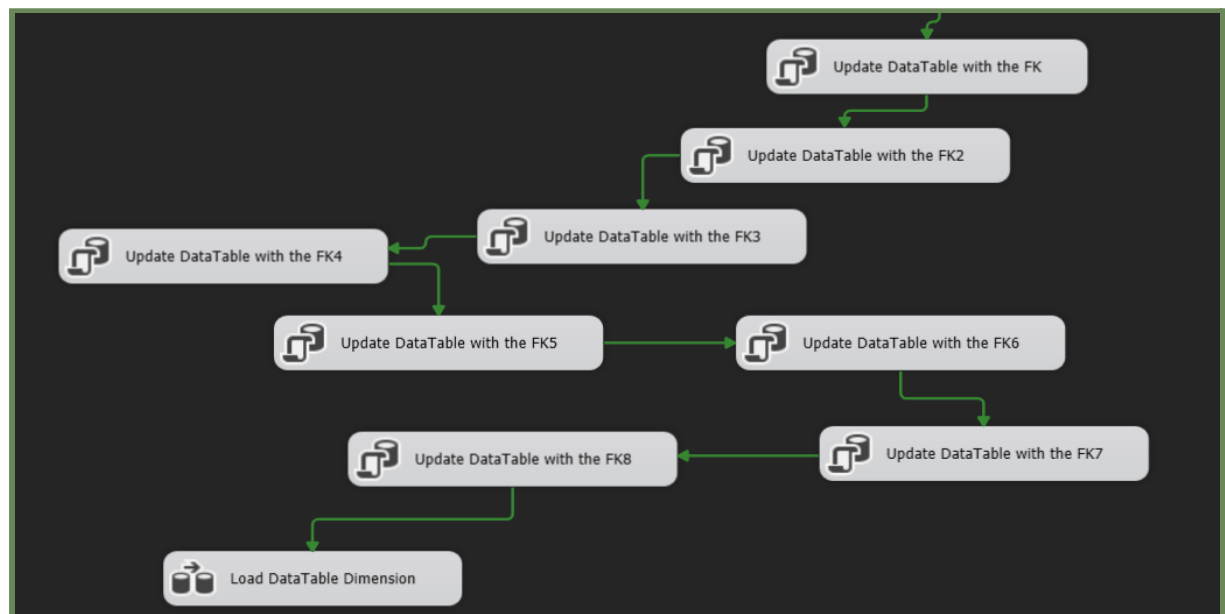
## Pour la Dimension “Cards” :



## Pour la Dimension “Transaction” :



Pour la Dimension “DataTable” - Table de fait :



→ Nous avons réalisé une importation des données afin de remplir la dimension - table “DataTable” mais il y avait plusieurs attributs qui sont restés nuls d'où la nécessité d'effectuer des requêtes SQL d' UPDATE afin de remplir les valeurs manquantes.

### UPDATE of RegionID

```

UPDATE BaseTable
SET      BaseTable.RegionID = Regions.RegionID
FROM      BaseTable
INNER JOIN
          Regions ON BaseTable.DistrictID = Regions.DistrictID
  
```

### UPDATE of RegionAnalysisID

```
UPDATE BaseTable
SET     BaseTable.RegionAnalysisID = RegionsAnalysis.RegionAnalysisID
FROM   BaseTable
INNER JOIN
        RegionsAnalysis ON BaseTable.RegionID = RegionsAnalysis.RegionID
```

### UPDATE of AccountID

```
UPDATE BaseTable
SET     BaseTable.AccountID = Accounts.AccountID
FROM   BaseTable
INNER JOIN
        Accounts ON BaseTable.DistrictID = Accounts.DistrictID
```

### UPDATE of DispositionID

```
UPDATE BaseTable
SET     BaseTable.DispoID = Dispositions.DispoID
FROM   BaseTable
INNER JOIN
        Dispositions ON BaseTable.AccountID = Dispositions.AccountID
```

### UPDATE of ClientID

```
UPDATE BaseTable
SET     BaseTable.ClientID = Dispositions.ClientID
FROM   BaseTable
INNER JOIN
        Dispositions ON BaseTable.AccountID = Dispositions.AccountID
```

### UPDATE of OrderID

```
UPDATE BaseTable
SET     BaseTable.OrderID = Orders.OrderID
FROM   BaseTable
INNER JOIN
        Orders ON BaseTable.AccountID = Orders.AccountID
```

### UPDATE of ClientID

```
UPDATE BaseTable
SET     BaseTable.ClientID = Dispositions.ClientID
FROM   BaseTable
INNER JOIN
        Dispositions ON BaseTable.AccountID = Dispositions.AccountID
```

### UPDATE of LoanID

```
UPDATE BaseTable
SET     BaseTable.LoanID = Loans.LoanID
FROM   BaseTable
INNER JOIN
        Loans ON BaseTable.AccountID = Loans.AccountID
```

### UPDATE of TransID



```

UPDATE BaseTable
SET     BaseTable.TransID =Transactions.TransID
FROM   BaseTable
INNER JOIN
        Transactions ON BaseTable.AccountID = Transactions.Account_ID

```

## ETL Complet :

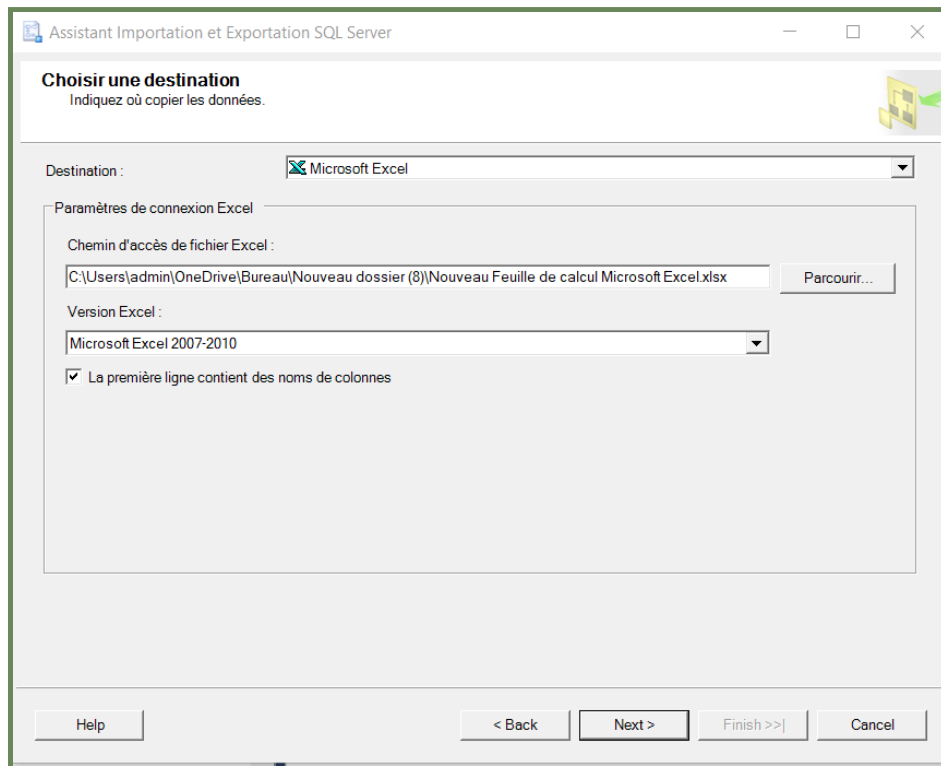
Nous remarquons ci-dessous le succès et la réussite de notre Pipeline en entier. Il est important de vérifier les tables de sorties pour s'assurer que tout a été importé en bel et en forme avant de commencer l'analyse.

**Dans notre cas, la conception du DataWarehouse a été changé deux fois en faisant des aller retour vers la partie analyse vu que les besoins changeait et donc la conception devrait être adéquate aux besoins et aux résultats qu'on voudrait avoir.**



# Analyse et Reporting

## 1- Exportation des données - dimensions sous format .CSV



Avant de commencer l'analyse, il est important d'exporter les données qu'on a eu comme résultats.

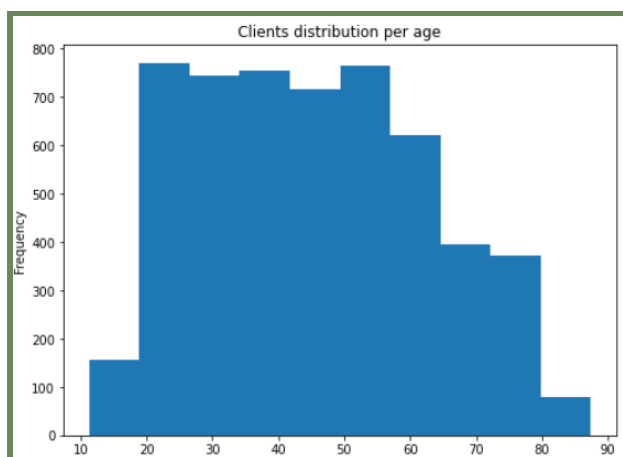
Pour exporter les données, nous utilisons l'assistant Importation et Exportation et nous précisons le type de fichier sur lequel il faut exporter nos données.

→ Dans notre cas, nous n'utilisons pas un logiciel de Dashboarding tel que Tableau et Power Bi vu que notre analyse est principalement basée sur les déductions que nous ferons à travers les visualisations réalisées.

## 2- Utilisation de Jupyter Notebook (Anaconda) pour l'analyse des données et la création de graphiques

Notre analyse se divise en parties tout en prenant en considération chaque dimension vu la richesse de la base de données et la richesse de chaque dimension.

Nous commençons par une analyse des données de base tel que **l'âge, le nombre de population par district, le salaire moyen par district et le pourcentage d'habitants urbains par district.**



### La distribution d'âge des clients de la banque :

On remarque d'après le graphique en bâtons suivant que nous avons une diversification en termes de répartition d'âge des clients de la banque.

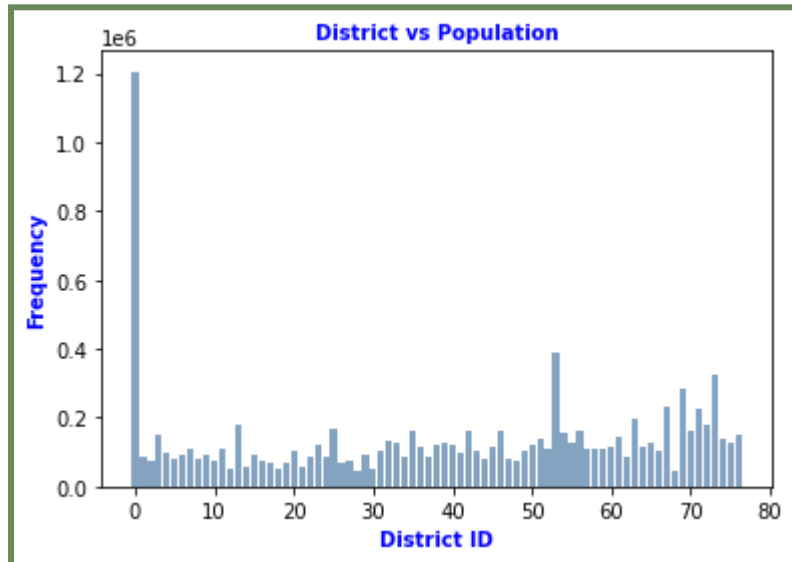
La majorité des clients de la banque ont un âge qui est compris entre 20 ans et 57 ans. A partir de 65

ans, le nombre de clients de la banque diminue progressivement jusqu'à atteindre les centaines (au max) vers l'âge de 80 ans.

Par contre, on a des clients qui sont toujours adolescents dont le nombre atteint les 150 clients. Leur âge est compris entre 12 et 20 ans.

→ **La banque offre alors une diversité d'offres adaptable à chaque client peu importe son âge.**

### Le nombre de population par district

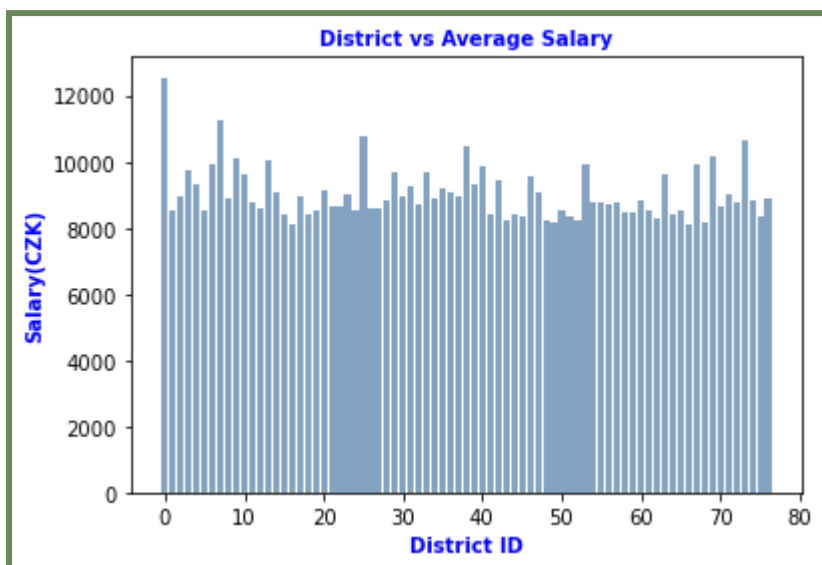


L'analyse nous donne comme résultat que parmi tous les districts, le seul district ayant un grand nombre d'habitants est le district 1 dont le nom est : Praha. Praha est en fait la capitale du pays, donc le district est considéré tel une ville aussi d'où le grand nombre d'habitants.

Tous les autres districts ont un nombre d'habitants qui atteint au plus 400k habitants. Ceci nous donne comme conclusion que nous avons une concentration d'habitants au niveau de la capitale plus que

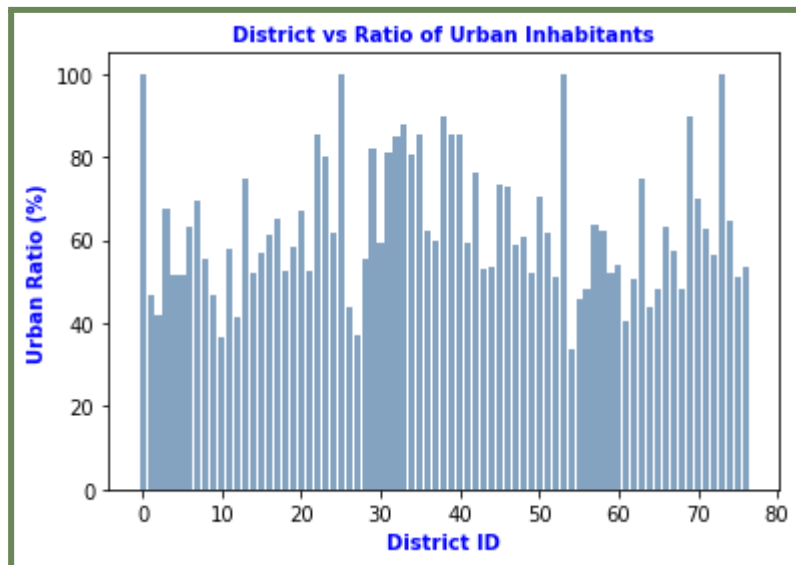
d'autres districts.

### La distribution du salaire moyen par district



Ce graphique nous donne une indication sur la répartition du salaire moyen par district. Nous remarquons qu'il n'y a pas une grande différence entre les différents salaires moyens au sein de chaque district. Le minimum du Salaire Moyen atteint environ les 9000 CZK alors que le max dépassé les 12000 CZK (Au niveau de la Capitale). Le salaire moyen le plus haut est celui des habitants qui vivent au niveau de la capitale.

### Le pourcentage de la population urbaine par district



Nous remarquons à travers le graphique que les districts ayant un Urban Ratio qui atteint les 100% c'est-à-dire les districts urbains à 100% sont les suivants : 1, 26, 54 et 74.

→ On pourrait déduire à partir des deux graphiques ci-dessus que la théorie affirmant que seuls les habitants urbains se font payer plus cher ou ont un salaire plus haut que les habitants dans les zones rurales ou semi urbaines, est certainement rejetée. Les districts 100% urbains tel que le 26 a un salaire moyen qui est moindre par rapport à une zone rurale.

### Analyse approfondi de la dimension “Loan” :

Au niveau de la dimension “Credit” et précisément au niveau de la colonne StatusLoan, on utilise des lettres telles que A, B, C et D. Nous considérons que les bons clients - clients fidèles représentent les lettres A et C. Pour les mauvais clients, ils seraient représentés par les lettres B et D.

Une échelle est représentée ci-dessous pour bien comprendre le statut de chaque clients et pourquoi on les considère bons ou mauvais.

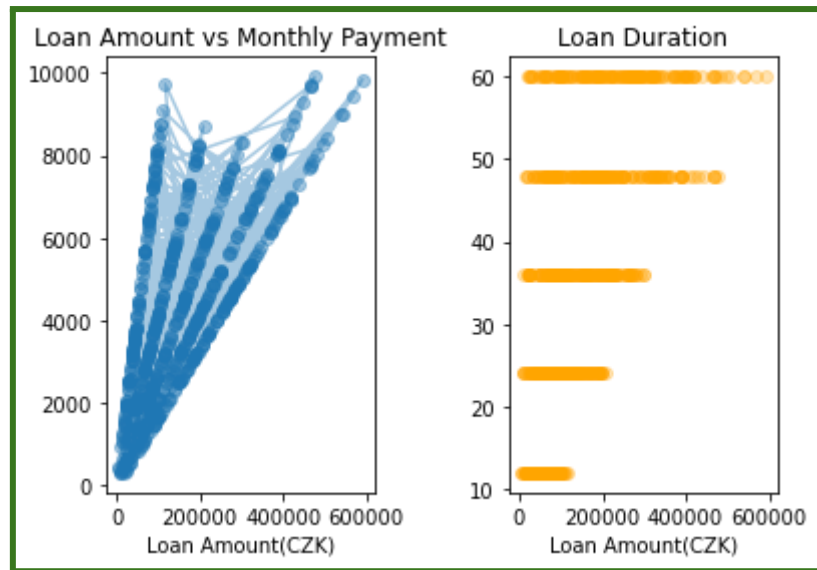
'A' stands for contract finished, no problems,

'B' stands for contract finished, loan not payed,

'C' stands for running contract, OK so far,

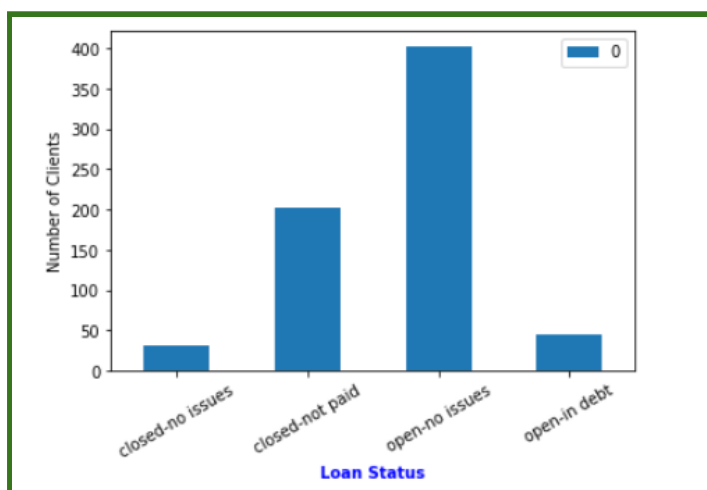
'D' stands for running contract, client in debt

**On commence par analyser le montant total du crédit et le paiement par mois proposés par la banque. Ensuite il est important d’analyser aussi la variation du montant du crédit avec la duration de ce dernier.**

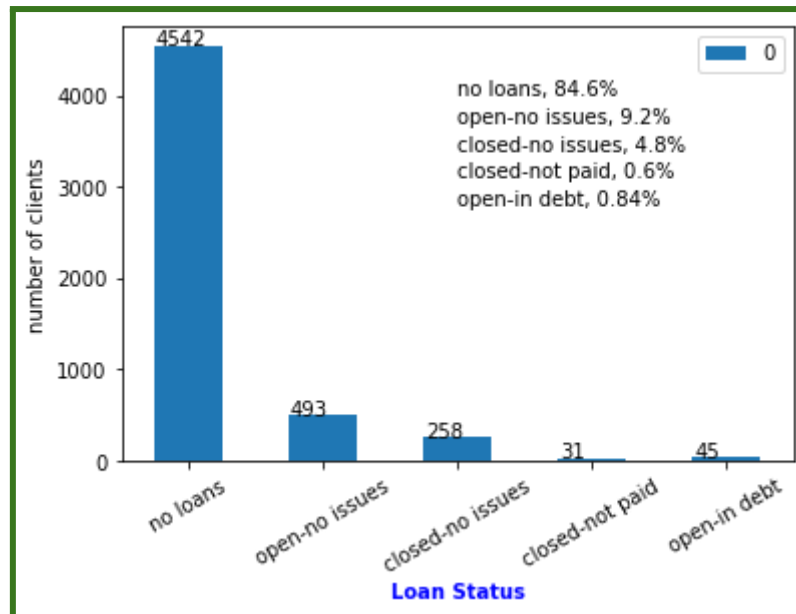


- Au niveau de la première figure, on pourrait remarquer que les personnes ayant pris un petit crédit pourraient être amenées à payer mensuellement une grande somme d'argent. Ceci a une relation avec le nombre de mois de paiement accordé ou choisi par le client. Un client avec une grande somme de crédit pourrait payer mensuellement une petite somme d'argent.
- Au niveau de la deuxième figure, on remarque que les personnes ayant pris un très grand crédit ont nécessairement un "loan duration" qui est plus grand et atteint des années mêmes d'où les petits montants payés mensuellement par ce dernier. Par contre les personnes avec des petits crédits choisissent une petite durée d'où le paiement mensuelle de grand montants.

**On analyse ensuite le LoanStatus, c'est-à-dire la variation des types de client (A-B-C ou D) au sein de la banque.**



On remarque au niveau de ce graphique que le nombre de clients qui régularise toujours leur situations au sein de la banque atteint au max les 400. Par contre, ceux qui ne payent pas leur Monthly Payment ont atteint les 50. Par contre, Pour les personnes qui ont pu finir le paiement de leur crédit en entier ont atteint les 40 clients alors que les personnes qui n'ont pas payé leur crédit a atteint les 200 clients. Ce qui montre qu'au niveau de la banque, dès que le crédit est terminé, un grand pourcentage de personnes finissent par ne pas payer. Presque 83% finissent par ne pas payer.

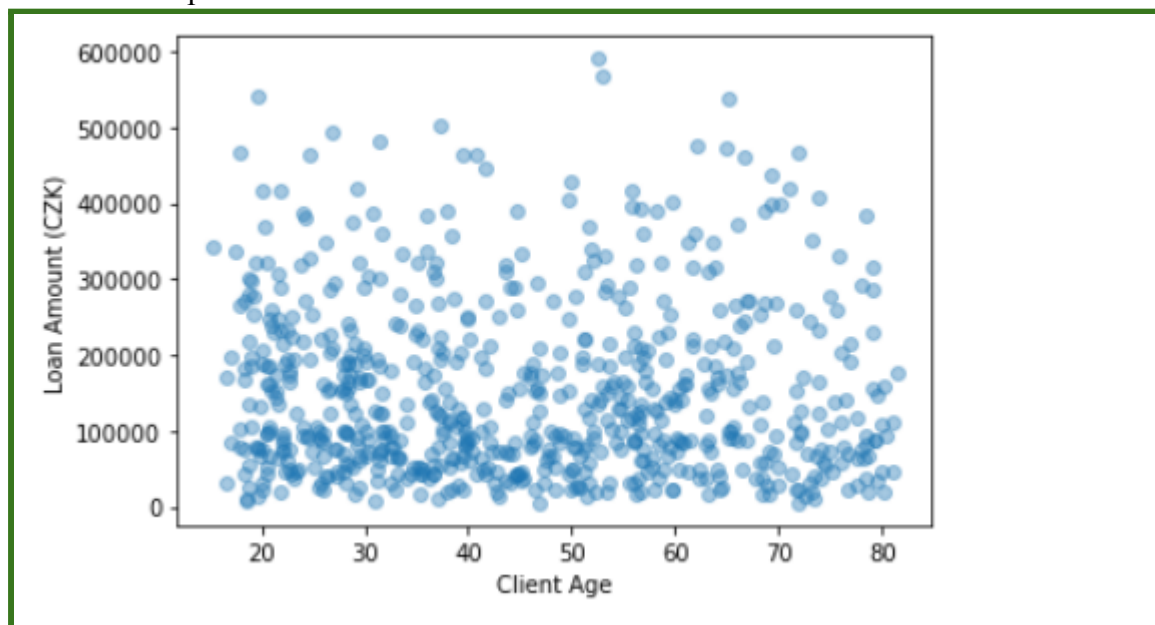


Dans ce graphique, on pourrait déduire que pas tous les clients de la banque sont des personnes ayant des crédits ou ayant pris des crédits. On remarque que 84,6%, donc presque la plupart des clients n'ont pas de crédits. Et le pourcentage des mauvais clients atteint les 0,90% donc presque 1%. La banque pourrait alors facilement gérer les preneurs de crédits.

### La distribution de l'âge des clients et le montant des crédits qui sont pris.

Au niveau du graphique ci-dessous, on pourrait conclure que la majorité des clients peu importe leur âge prennent des crédits compris entre 50000 CZK et 150000 CZK.

Ce qui est plus ou moins choquant c'est qu'il y a des adolescents preneurs de crédits ayant un grand montant et qui atteignent 350000 CZK. Est ce qu'ils arrivent à payer leur crédit en entier ? En respectant le délai accordé ?



Au niveau des deux graphes ci-dessous, on remarque qu'il n'y a pas de différences entre les clients "hommes" et les clientes "femmes". Donc être un bon ou un mauvais client ne dépend

pas du sexe du client. Une autre raison pour rejeter la théorie affirmant que les femmes font partie des mauvais clients vu leur "incapacité" de payer le crédit en entier en respectant le délai accordé.



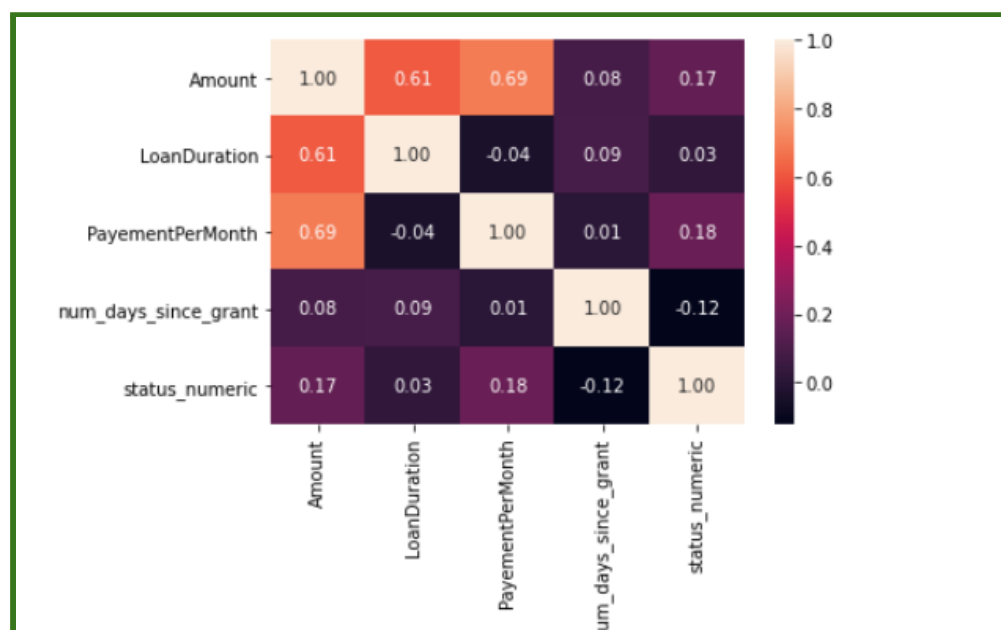
D'après notre analyse précédente, on déduit donc:

- Les hommes et les femmes sont presque également présents parmi les clients de la banque, 50,74 % pour les hommes et 49,26 % pour les femmes.
- La majorité des clients ont un âge compris entre 20 et 60 ans.
- La plupart des clients vivent dans le Nord, au contraire du Sud et de l'Ouest.
- Le salaire moyen est presque également réparti entre les régions, à l'exception de la capitale du pays où le salaire moyen est le plus élevé.

### Matrice de corrélation entre les éléments appartenant à la dimension "Credit"

Nous allons créer une heatmap liant les différents éléments qui interagissent avec les crédits qui sont pris.

**\*Une astuce trouvée c'est qu'il suffisait de considérer le A et C = 1 et le B et D = -1 vu que le A et C représentent les bons clients et B et D représentent les mauvais clients, d'où l'utilisation de status\_numeric.\***





- Ce que nous pouvons conclure à partir de la matrice de corrélation, qu'une durée plus longue augmente (plus ou moins) légèrement la probabilité qu'un prêt ne soit pas payé.
- Nous pouvons aussi remarquer que quand on a un montant plus élevé de payment per month, il est corrélé avec des dettes qui sont généralement impayées, tandis qu'un montant inférieur dans la plupart des cas représente un cas OK.

### Analyse de la répartition des types de clients ou différents statuts de crédits en fonction des différents districts

| status_desc  | Contract finised, loan was not paid | Contract finished, no problem | Runing contract, OK so far | Runing contract, client in debt |
|--------------|-------------------------------------|-------------------------------|----------------------------|---------------------------------|
| DistrictName |                                     |                               |                            |                                 |
| Benesov      | 0.00                                | 16.67                         | 83.33                      | 0.00                            |
| Beroun       | 0.00                                | 16.67                         | 50.00                      | 33.33                           |
| Blansko      | 0.00                                | 42.86                         | 42.86                      | 14.29                           |
| Breclav      | 14.29                               | 14.29                         | 71.43                      | 0.00                            |
| Brno - mesto | 8.33                                | 16.67                         | 62.50                      | 12.50                           |

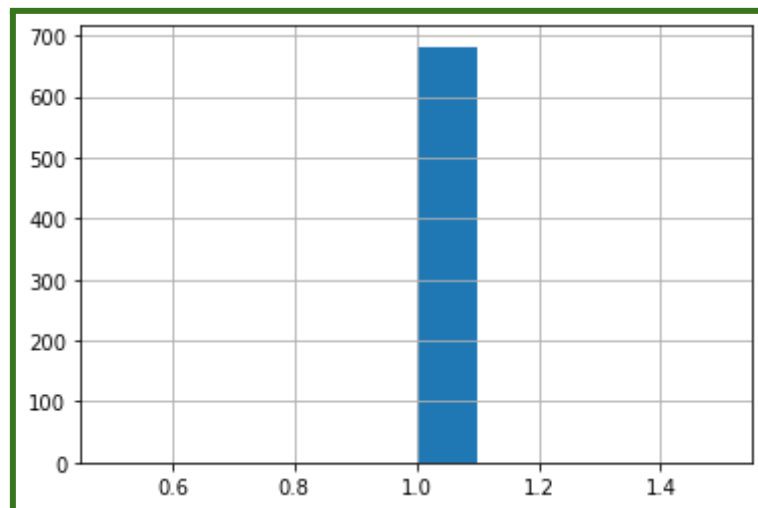
Nous pourrions remarquer au niveau de la HeatMap ci dessous qu'il y a des districts ou on a une abondance de mauvais clients tel que : **Strakonice, Sokolov, Opava, Kutna Hora, Klatovy, Domazlice, Bruntal et Beroun.** (Avec un pourcentage qui dépasse les 30%).

|                     |       |       |        |       |
|---------------------|-------|-------|--------|-------|
| Benesov -           | 0.00  | 16.67 | 83.31  | 0.00  |
| Beroun -            | 0.00  | 16.67 | 50.00  | 33.34 |
| Blansko -           | 0.00  | 42.86 | 42.86  | 14.29 |
| Breclav -           | 14.29 | 14.29 | 71.44  | 0.00  |
| Brno - mesto        | 8.33  | 16.67 | 62.50  | 12.50 |
| Brno - venkov -     | 0.00  | 40.00 | 50.00  | 10.00 |
| Bruntal -           | 16.67 | 33.34 | 16.67  | 33.34 |
| Ceska Lipa -        | 0.00  | 33.34 | 66.69  | 0.00  |
| Ceske Budejovice -  | 0.00  | 12.50 | 75.00  | 12.50 |
| Cesky Krumlov -     | 14.29 | 42.86 | 42.86  | 0.00  |
| Cheb -              | 0.00  | 50.00 | 50.00  | 0.00  |
| Chomutov -          | 11.11 | 33.34 | 55.56  | 0.00  |
| Chrudim -           | 0.00  | 25.00 | 50.00  | 25.00 |
| Decin -             | 0.00  | 25.00 | 75.00  | 0.00  |
| Domazlice -         | 0.00  | 0.00  | 50.00  | 50.00 |
| Frydek - Mistek -   | 6.25  | 43.75 | 50.00  | 0.00  |
| Havlickuv Brod -    | 0.00  | 14.29 | 71.44  | 14.29 |
| Hl.m. Praha -       | 3.57  | 40.47 | 51.19  | 4.76  |
| Hodonin -           | 0.00  | 33.34 | 66.69  | 0.00  |
| Hradec Kralove -    | 0.00  | 20.00 | 60.00  | 20.00 |
| Jablonec n. Nisou - | 0.00  | 0.00  | 100.00 | 0.00  |
| Jesenik -           | 0.00  | 25.00 | 50.00  | 25.00 |
| Jicin -             | 0.00  | 42.86 | 57.12  | 0.00  |
| Jihlava -           | 0.00  | 0.00  | 100.00 | 0.00  |
| Jindrichuv Hradec - | 9.09  | 36.38 | 54.55  | 0.00  |
| Karlovy Vary -      | 25.00 | 37.50 | 37.50  | 0.00  |
| Karvina -           | 0.00  | 37.50 | 50.00  | 12.50 |
| Kladno -            | 0.00  | 28.56 | 57.12  | 14.29 |
| Klatovy -           | 0.00  | 0.00  | 66.69  | 33.34 |
| Kolin -             | 0.00  | 20.00 | 80.00  | 0.00  |
| Kromeriz -          | 0.00  | 11.11 | 77.75  | 11.11 |
| Kutna Hora -        | 22.22 | 0.00  | 66.69  | 11.11 |
| Liberec -           | 0.00  | 0.00  | 100.00 | 0.00  |
| Litomerice -        | 0.00  | 28.56 | 71.44  | 0.00  |
| Louny -             | 0.00  | 30.77 | 69.25  | 0.00  |
| Melnik -            | 0.00  | 33.34 | 66.69  | 0.00  |
| Mlada Boleslav -    | 0.00  | 0.00  | 80.00  | 20.00 |
| Most -              | 0.00  | 60.00 | 40.00  | 0.00  |
| Nachod -            | 0.00  | 16.67 | 83.31  | 0.00  |
| Novy Jicin -        | 0.00  | 16.67 | 66.69  | 16.67 |
| Nymburk -           | 0.00  | 55.56 | 44.44  | 0.00  |
| Olomouc -           | 7.14  | 21.44 | 64.31  | 7.14  |
| Opava -             | 12.50 | 12.50 | 50.00  | 25.00 |
| Ostrava - mesto     | 10.00 | 30.00 | 55.00  | 5.00  |
| Pardubice -         | 0.00  | 10.00 | 90.00  | 0.00  |
| Peihrimov -         | 0.00  | 25.00 | 75.00  | 0.00  |
| Pisek -             | 0.00  | 33.34 | 66.69  | 0.00  |
| Pilsen - uh.        | 0.00  | 22.22 | 77.75  | 0.00  |

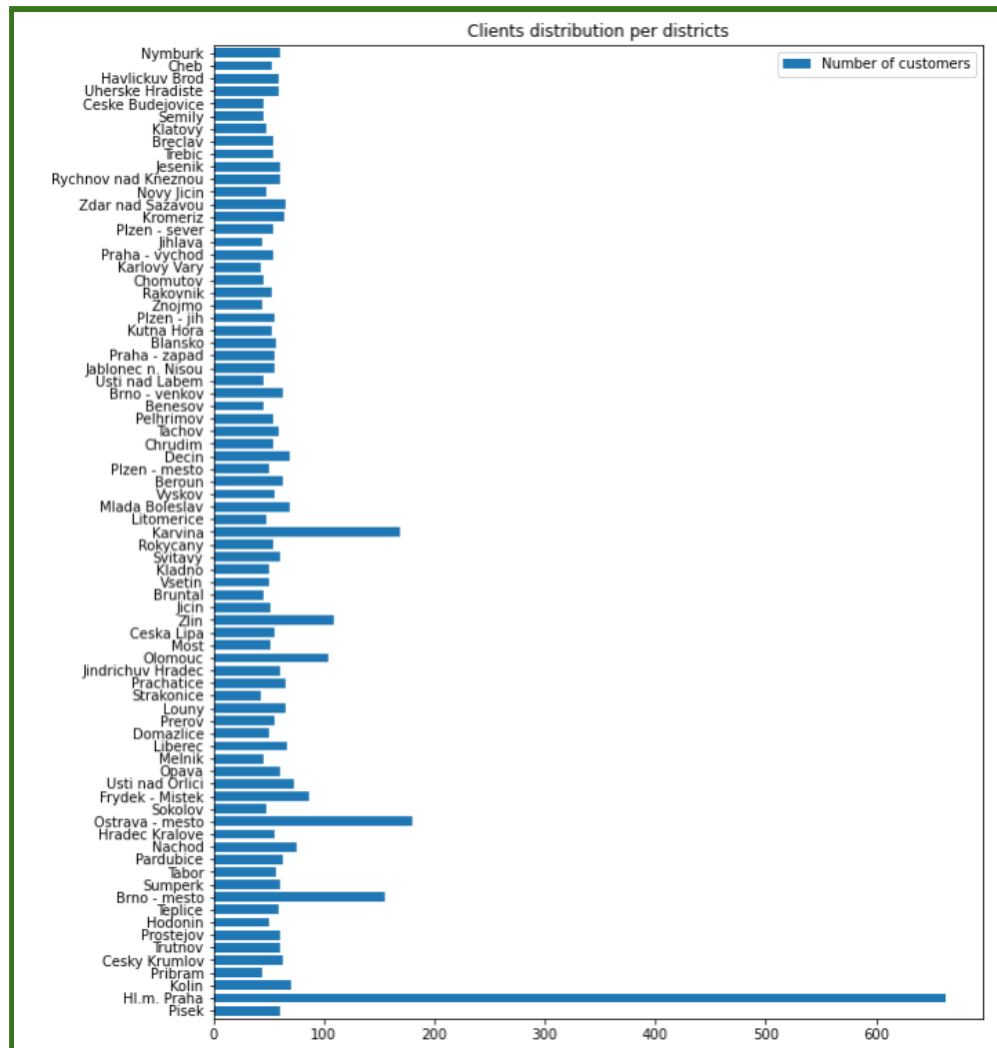


|                       |                                     |                               |                            |                                 |
|-----------------------|-------------------------------------|-------------------------------|----------------------------|---------------------------------|
| Pisek -               | 0.00                                | 33.34                         | 66.69                      | 0.00                            |
| Plzen - jih           | 0.00                                | 22.22                         | 77.75                      | 0.00                            |
| Plzen - mesto         | 0.00                                | 16.67                         | 66.69                      | 16.67                           |
| Plzen - sever         | 0.00                                | 28.56                         | 71.44                      | 0.00                            |
| Prachatic             | 0.00                                | 35.72                         | 57.12                      | 7.14                            |
| Praha - vychod        | 0.00                                | 40.00                         | 60.00                      | 0.00                            |
| Praha - zapad         | 9.09                                | 54.55                         | 36.38                      | 0.00                            |
| Prerov                | 0.00                                | 75.00                         | 25.00                      | 0.00                            |
| Pribram               | 12.50                               | 25.00                         | 62.50                      | 0.00                            |
| Prostejov             | 11.11                               | 22.22                         | 66.69                      | 0.00                            |
| Rakovnik              | 12.50                               | 37.50                         | 50.00                      | 0.00                            |
| Rokycany              | 0.00                                | 12.50                         | 75.00                      | 12.50                           |
| Starychov nad Kneznou | 0.00                                | 16.67                         | 66.69                      | 16.67                           |
| Semily                | 0.00                                | 33.34                         | 66.69                      | 0.00                            |
| Sokolov               | 50.00                               | 50.00                         | 0.00                       | 0.00                            |
| Strakonice            | 16.67                               | 16.67                         | 33.34                      | 33.34                           |
| Sumperk               | 0.00                                | 0.00                          | 100.00                     | 0.00                            |
| SVitavy               | 0.00                                | 16.67                         | 58.34                      | 25.00                           |
| Tabor                 | 28.56                               | 28.56                         | 42.88                      | 0.00                            |
| Tachov                | 25.00                               | 62.50                         | 12.50                      | 0.00                            |
| Teplice               | 0.00                                | 33.34                         | 66.69                      | 0.00                            |
| Trebic                | 16.67                               | 50.00                         | 33.34                      | 0.00                            |
| Trutnov               | 0.00                                | 16.67                         | 83.31                      | 0.00                            |
| Uherske Hradiste      | 0.00                                | 10.00                         | 90.00                      | 0.00                            |
| Usti nad Labem        | 0.00                                | 66.69                         | 33.34                      | 0.00                            |
| Usti nad Orlici       | 0.00                                | 35.72                         | 57.12                      | 7.14                            |
| Vsetin                | 0.00                                | 16.67                         | 83.31                      | 0.00                            |
| Vyskov                | 0.00                                | 42.88                         | 57.12                      | 0.00                            |
| Zdar nad Sazavou      | 0.00                                | 57.12                         | 42.88                      | 0.00                            |
| Zlin                  | 0.00                                | 35.28                         | 58.81                      | 5.88                            |
| Znojmo                | 16.67                               | 16.67                         | 66.69                      | 0.00                            |
|                       |                                     |                               |                            |                                 |
|                       | Contract finised, loan was not paid | Contract finished, no problem | Runing contract, OK so far | Runing contract, client in debt |

Il est important de noter aussi qu'en analysant le nombre de crédit et sa relation avec le nombre de comptes, on pourrait conclure que chaque compte a pris un unique crédit tout comme le montre la figure ci-dessous. ( Ce résultat est obtenu en utilisant un Group By)



La relation entre la dimension Clients et la Distribution géographique

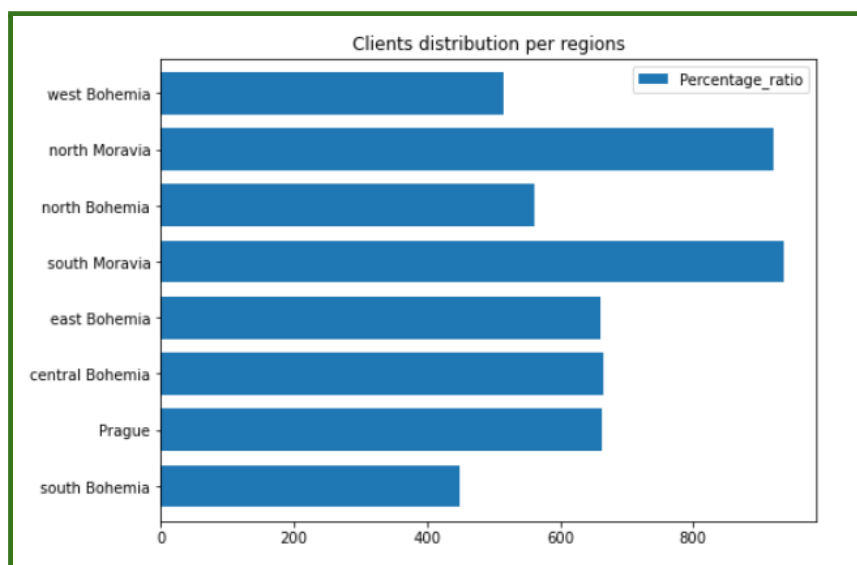
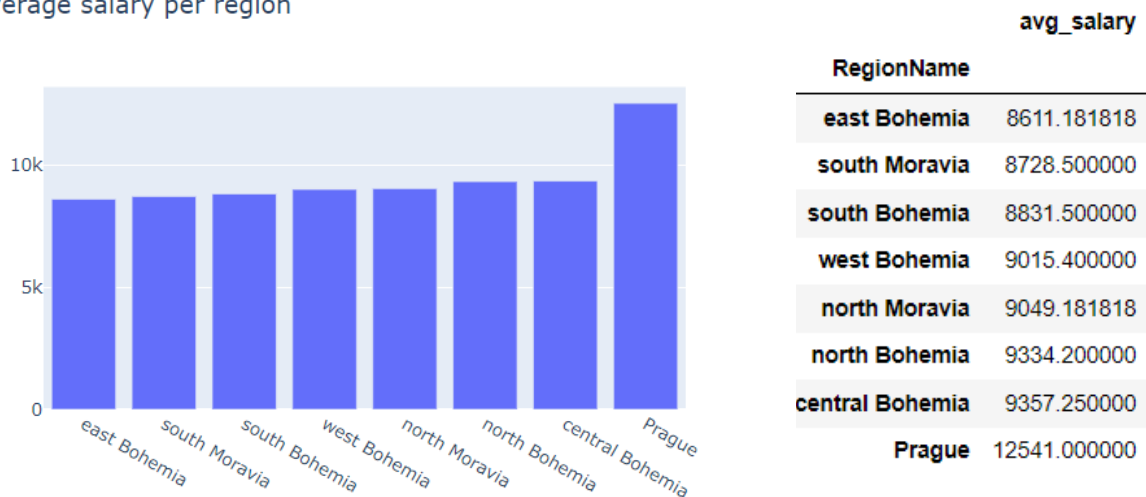


Encore une fois, nous pouvons conclure que la majorité des clients sont des clients appartenant à la capitale et à des districts 100% urbains tels que Brno et Ostrava qui sont classés seconds après Praha qui a atteint un nombre de client qui dépasse les 700 clients.

Nous pouvons conclure que la capitale dont le salaire moyen est le plus grand, la capitale qui est 100% urbanisée et maintenant est classée première en termes de nombre des clients de la banque, prend le dessus par rapport aux autres villes.

Cette répartition inéquitable pourrait rendre la Banque dépendante de la capitale. Et donc empêcherait cette dernière d'acquérir de nouveaux clients et de développer son activité pour atteindre plus de clients.

Average salary per region



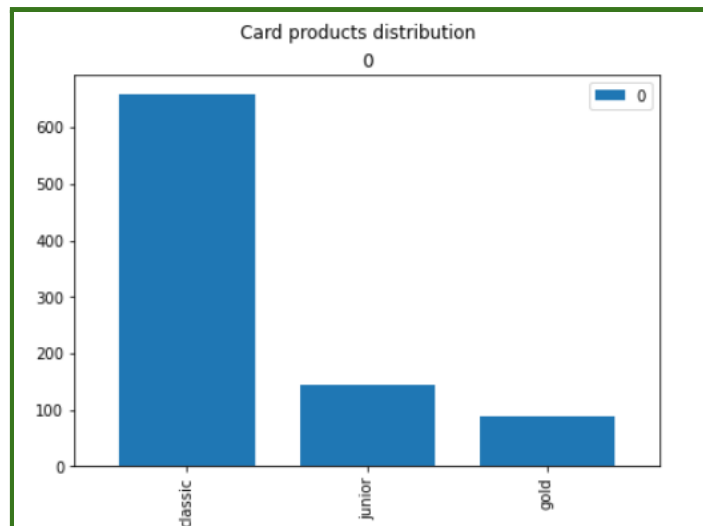
Cette analyse nous permet d'avantage de comprendre la distribution des clients par rapport aux régions du pays. Encore une fois, Praha a un grand nombre de clients avec un salaire moyen élevé aussi.

Ce qui nous mènerait à conclure que plus on a un salaire moyen élevé au niveau de la région, plus la banque aura des clients appartenant a cette region meme.

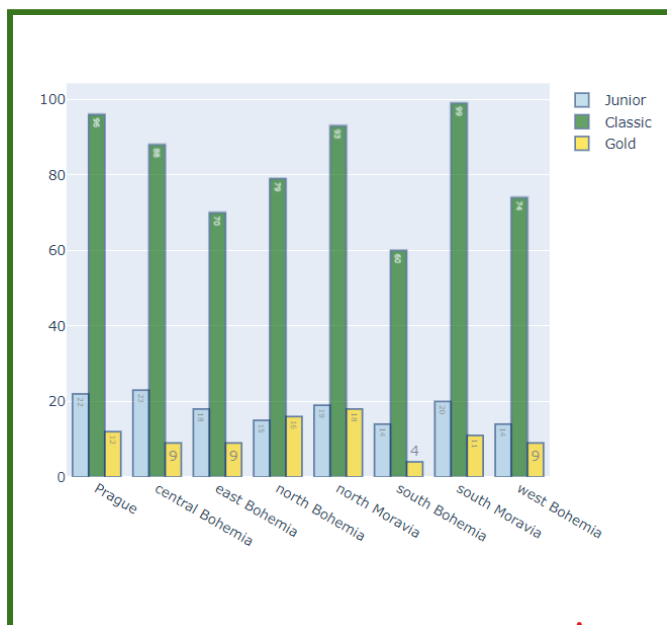
Ceci pourrait aider la banque à définir sa population cible et la population qui est facilement atteignable et ainsi mettre en place une stratégie marketing ou stratégie publicitaire afin d'attirer le plus de clients appartenant à la région ciblée.

#### - Analyse de la dimension Cards :

#### Distribution des types de cartes pour connaître les cartes les plus utilisées.



Les cartes les plus utilisées sont des cartes de type **“Classic”**, ensuite on a des cartes **“Junior”** qui sont consacrées pour les étudiants et en final on a des cartes de type **“Gold”** qui sont consacrées pour les clients fidèles effectuant des ordres presque permanents au niveau de la banque. On a ainsi **74%** des cartes émises sont des cartes classiques, **16%** des cartes junior et **10%** des cartes gold.



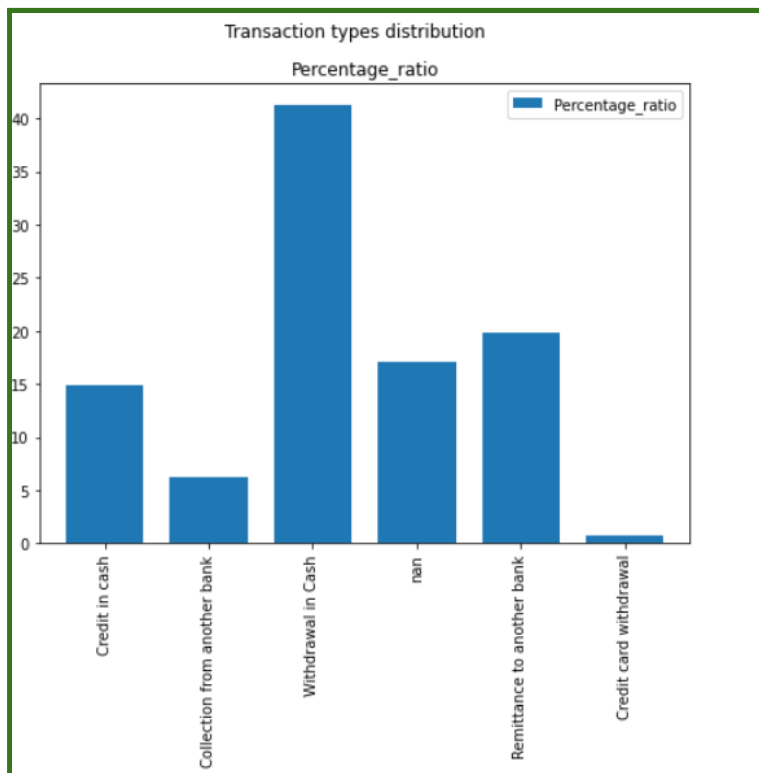
Tout comme les services proposés par la banque qui s'adaptent aux différentes tranches d'âges, les cartes aident aussi à gagner plus de clients et à définir la tranche d'âge de chaque client. Chaque carte donne à son client des services personnalisés même au niveau des offres de crédits.

La figure ci-joint nous affiche la distribution des types de cartes par région.

La majorité des cartes dans toutes les régions sont **“Classic”**, on estime qu'on en a plus de **70 %**, les cartes **“Junior”** prévalent sur celles **“Gold”** dans toutes les régions, à l'exception de la North Bohemia.

La North Bohemia est classée troisième en termes de salaire moyen, et c'est une région avec une forte orientation vers l'industrie lourde, ce qui pourrait expliquer l'abondance des cartes Classic et Gold.

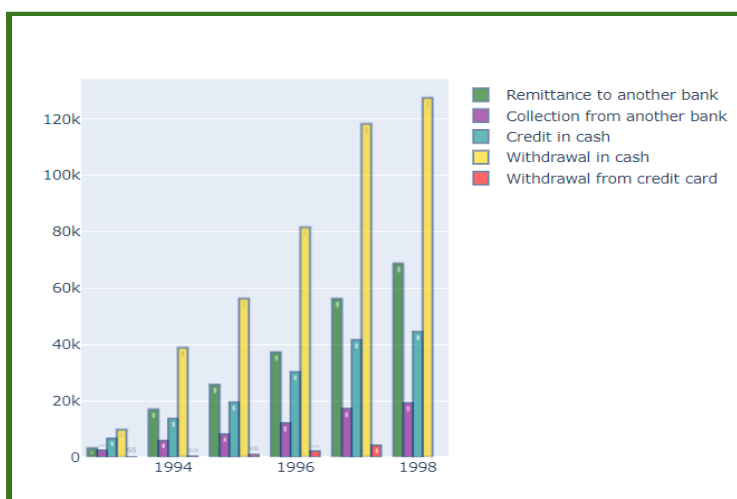
### Analyse de la Dimension Transactions :



| Percentage_ratio             |           |
|------------------------------|-----------|
| Credit in cash               | 14.907851 |
| Collection from another bank | 6.220442  |
| Withdrawal in Cash           | 41.234723 |
| NaN                          | 17.038648 |
| Remittance to another bank   | 19.863434 |
| Credit card withdrawal       | 0.734902  |

Pour la partie “**Transaction**”, nous pouvons analyser le type de transactions effectuées. Dans notre graphique ci-dessus, nous remarquons que le type de transaction le plus abondant et qui atteint les **41%** est celui de “**Withdrawal in Cash**”. Après, on a **19%** pour les remises d'une autre banque, **17%** pour les nan, définit par la suite en tant que crédit d'intérêts.

La moins utilisée est le retrait d'espèces par carte de crédit, ce qui est plus ou moins raisonnable car cette transaction est chargée au côté des clients. (**Ils payent après chaque transaction par carte de crédit**).

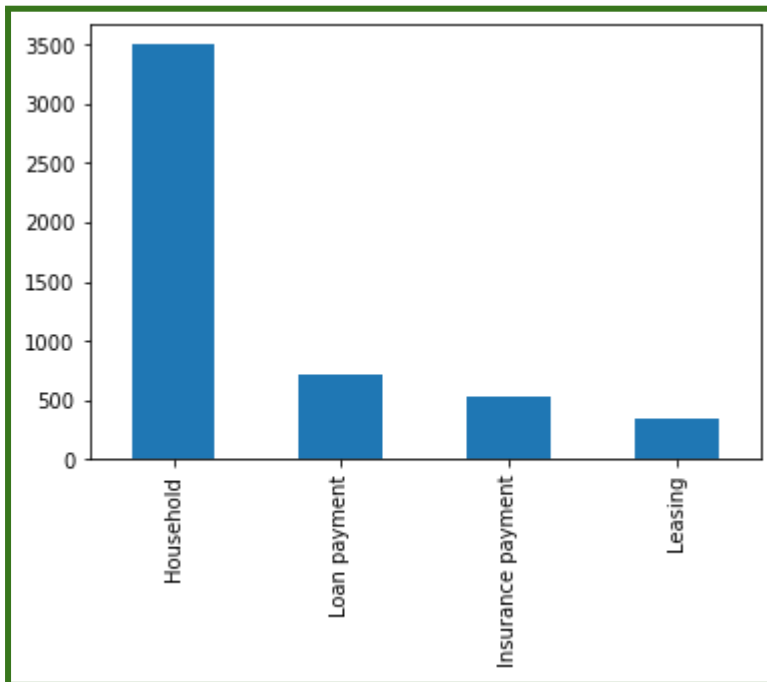


La répartition des transactions au cours des années (**1993-1998**) est essentiellement la même.

On a par contre le nombre de transactions pour “**Remittance**” est supérieur au nombre de collectes auprès d'une autre banque (plus de 2 fois certaines années).

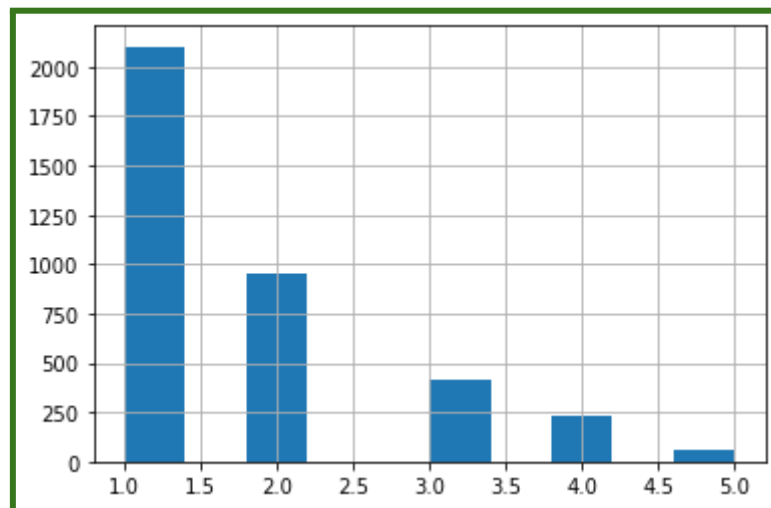
Par contre les Withdrawal in cash dépassent largement les Withdrawal from credit card même durant les 5 années.

### Analyse de la Dimension Orders :



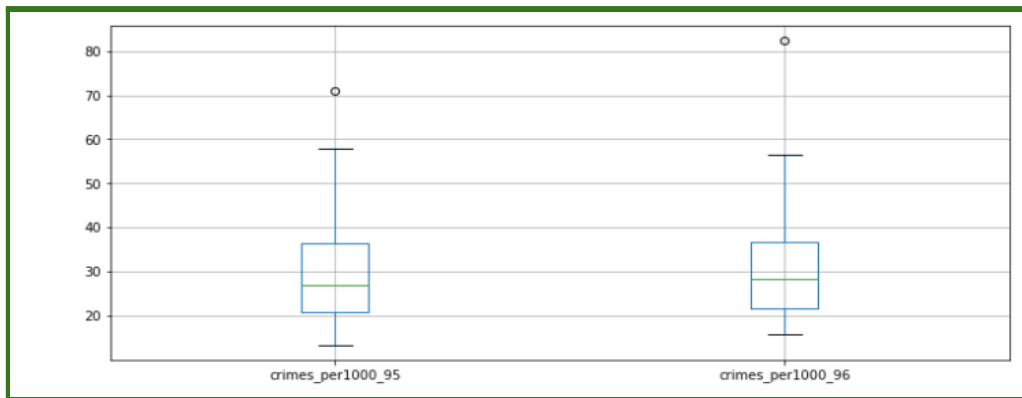
On remarque d'après cette figure que la plupart des “orders” sont liés à des “HouseHolds” en atteignant un nombre de 3500 ordres. Par contre pour les autres ordres sont réparties sur les différentes sections ou utilité tel que le Loan Payment, Insurance Paiement et en final le Leasing.

Ci-dessous on pourrait aussi déduire que la plupart des comptes effectuent un seul ordres, ce nombre atteint les 2100 comptes alors que les autres effectuent 2 ordres jusqu'à 5 ordres, mais le nombre est moindre.



### Analyse des attributs “Crimes” et "Unemployment" :

- Pour le taux de “Crimes” :



-Pour le "Unemployment" :



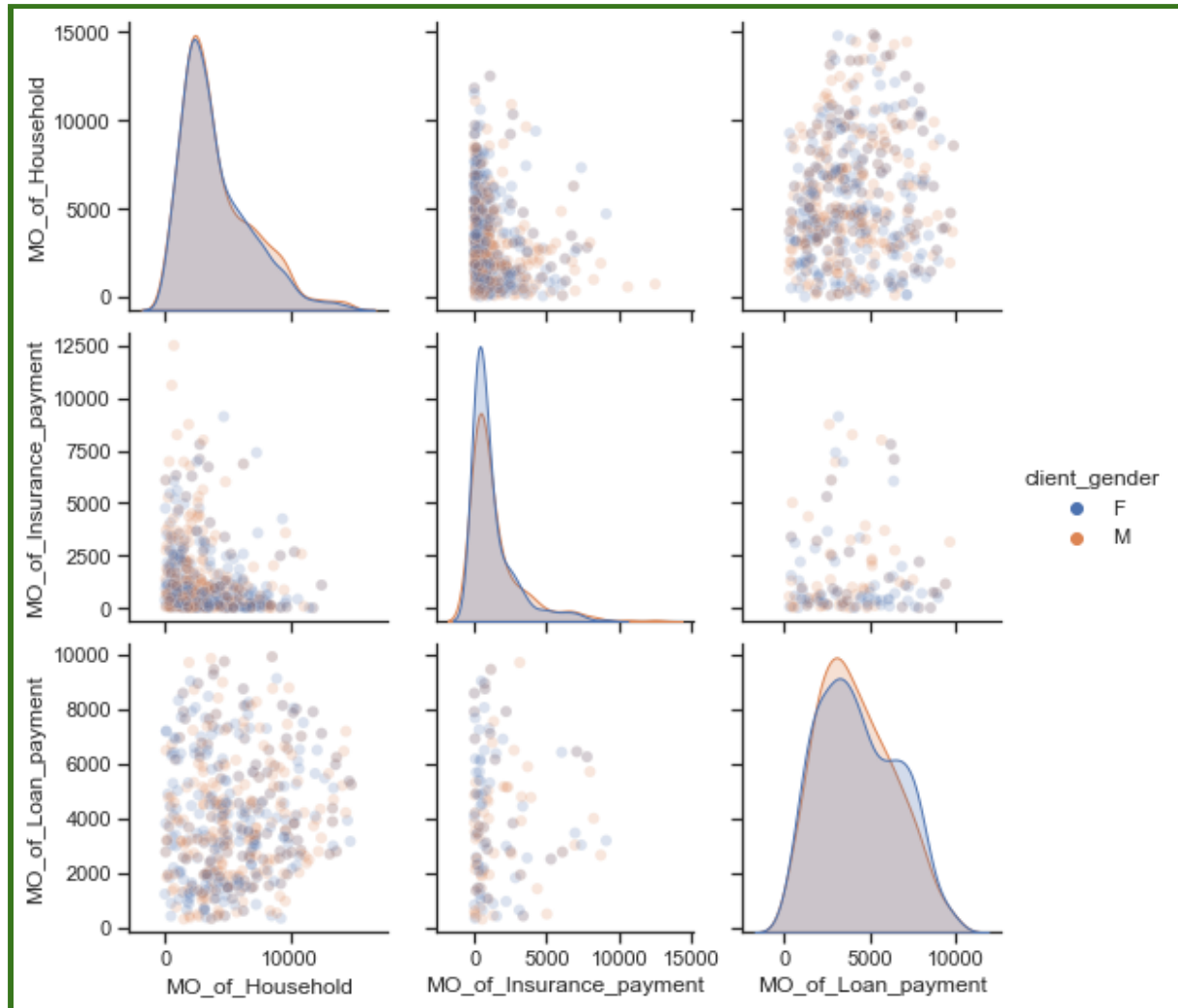
Analyse de l'attribut "Nombre d'entrepreneurs" :



Praha a encore le plus grand nombre d'entrepreneurs / 1000 personnes. Ceci est un indicateur qui pourrait nous expliquer le nombre de clients qui est en hausse au niveau de la capitale.

Mais encore, il est important pour la banque de prendre en compte le taux d'entrepreneurs au niveau de chaque districts afin de cibler mieux ses clients et offrir des offres et services répondant à leurs besoins et leur permettent aussi d'atteindre leurs objectifs entrepreneuriaux.

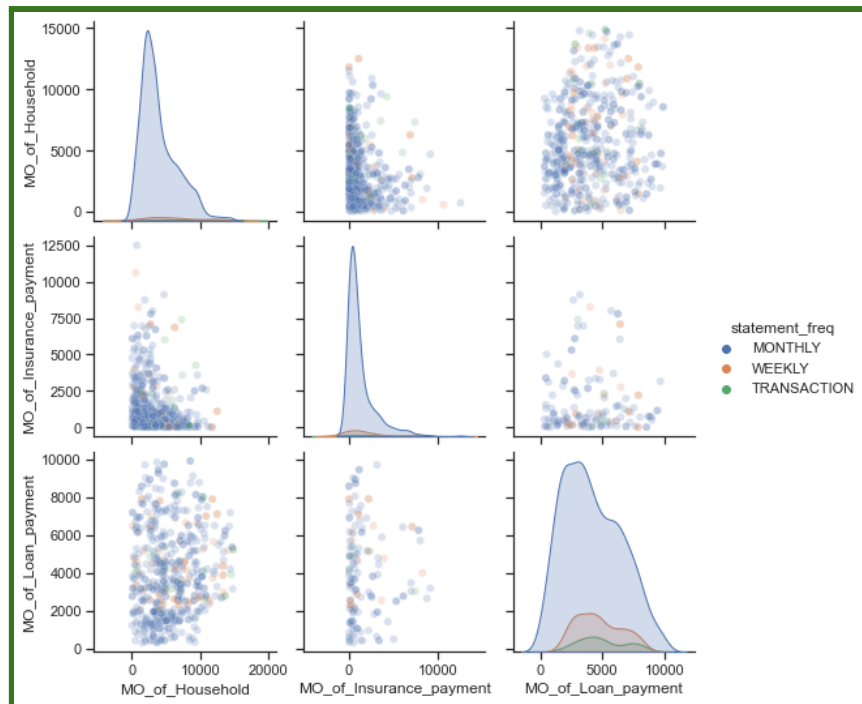
#### Distribution du gender des clients en fonction des types d'ordres :



Pour chaque catégorie, les différences entre les deux sexes sont insignifiantes. Ceci est conforme à la précédente analyse concernant le sexe des clients.

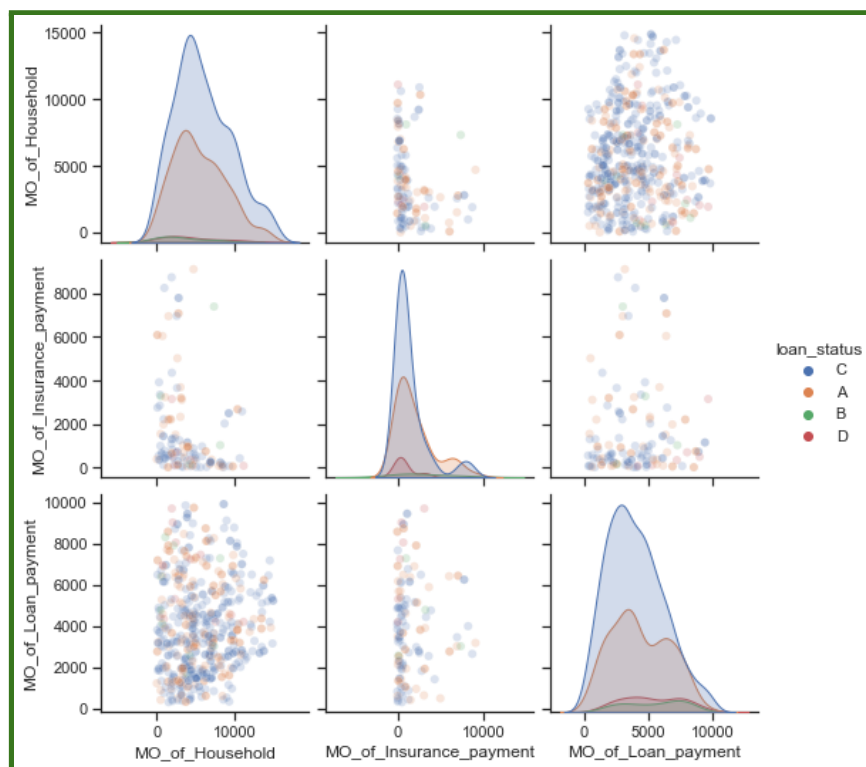
#### Distribution de la fréquence de relevé en fonction des types d'ordres :





La plupart des ordres « **HouseHold** » et « **Insurance Payment** » ont une fréquence de relevé « **Mensuelle** ».

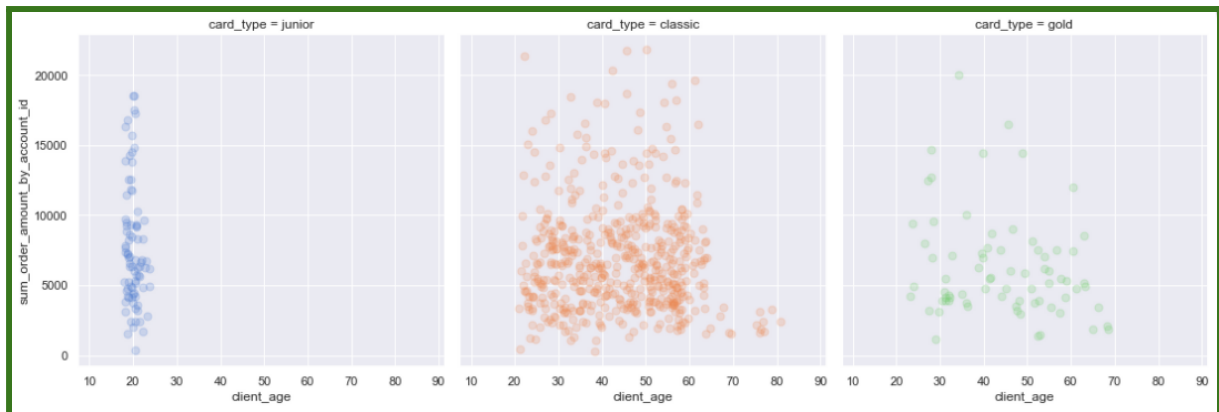
Cependant, pour le « **Loan Payment** », ils sont fréquemment déclarés sur une base « **hebdomadaire** » ou « **transaction** ».



Pour le statut du loan dans chaque type d'ordre, le groupe « **C** » est en tête de la catégorie « **HouseHolds** ». « **C** » et « **A** » sont fortement présents dans le groupe « **Insurance Payments** ». La répartition en « **Loan Payment** » affiche aussi une hausse au niveau des catégories C et A.

Un autre indicateur mettant en valeur les bon clients par rapport aux mauvais clients c'est leur capacité à payer leurs crédits à temps à travers les ordres.

### La variation du montant de l'ordre en fonction de l'âge des clients ainsi que le type de cartes.



La plupart des clients ayant un âge de 20 ans possèdent le type de carte « junior », le montant moyen des ordres se situe entre 2000 et 8000 CZK.

Par contre, les personnes âgées (entre 60 et 80 ans) et la plupart des clients ne possèdent que le type de carte « classique ».

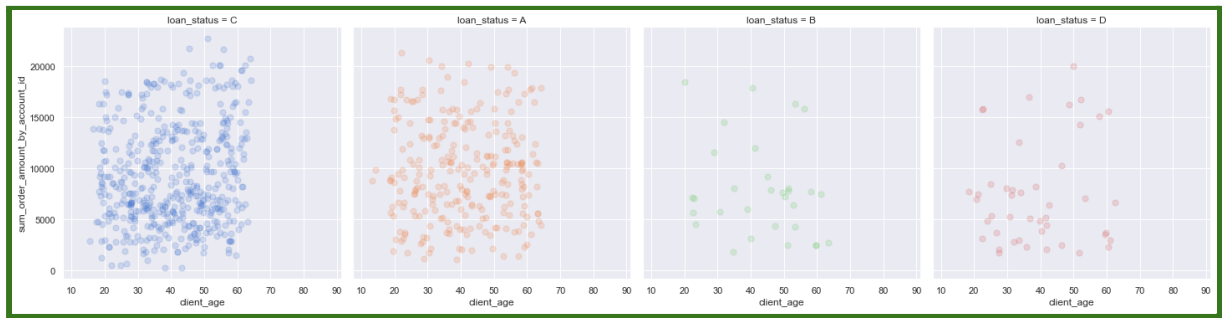
Le type de carte dépend alors de l'âge des clients et du montant d'ordre de ces derniers.

### La variation du montant de l'ordre en fonction de l'âge des clients ainsi que le sexe des clients.



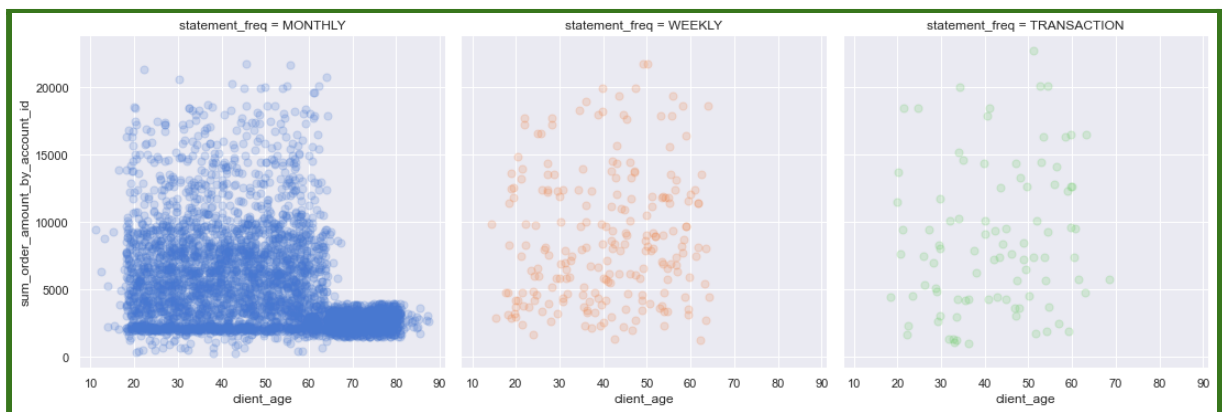
La répartition des clients par sexes est assez similaire. Même en prenant en considération l'âge et la somme totale des ordres.

### La variation du montant de l'ordre en fonction de l'âge des clients ainsi que le statut des clients preneurs de crédits.



On remarque que le montant moyen des ordres pour les groupes B et D est de 8 000 CZK par rapport aux clients dont le statut est C et A dont le montant moyen des ordres atteint les 10 000 CZK.

Les bons clients effectuent plus d'ordres et les montants sont très élevés par rapport aux mauvais clients, ceci pourrait être un indicateur à prendre en compte afin de prédire les bons et mauvais clients.



On remarque qu'au niveau des fréquences de statement que les clients seniors (âgés de plus de 60 ans) ont une fréquence principalement mensuelle. Aucun senior n'apparaît dans les groupes « Weekly » ou « Transaction ».

Les jeunes ont tendance à demander des relevés de compte chaque semaine ou même au niveau de chaque transaction, ceci pourrait être un élément à prendre en compte aussi lors du choix des mauvais ou bon clients.

Principalement les personnes âgées entre 20 et 60 se soucient plus de leur situation au niveau de la banque d'où la demande hebdomadaire des relevés de comptes. Il y a des cas où les banques donneurs de crédits exigent des relevés de comptes aussi d'où une autre raison nous poussant à conclure que les seniors ne demandent les relevés de comptes que mensuellement.

## Conclusion

Pour conclure, il est important d'effectuer des analyses quotidiennes et hebdomadaires des comptes, transactions et autres données au sein de la banque en question. Ceci va permettre aux preneurs de décisions de mieux cibler les clients, d'avoir une idée claire sur la population

et les districts qui l'entourent et d'éviter les Scams et les problèmes qui occurred après un non paiement des crédits.

Il est également important pour la banque de garantir sa protection et de pouvoir prédire à l'avance les mauvais clients afin de prendre les bonnes mesures au bon moment.