**# How to analyse a company's sales data?**

# Import packages

import pandas as pd

import numpy as np

import os

import glob

import matplotlib.pyplot as plt

import seaborn as sns

# to define the dataset path

path = r'C:\Users\Charles K\Desktop\Formations\Python\PROJETS\Sales\_Analyse'

# change the work repetory

os.chdir(path) # To list all files on repetory

all\_filenames = [i for i in glob.glob('\*.{}'.format('csv'))]

# Fusionnez tous les fichiers CSV en un seul DataFrame

all\_file = pd.concat([pd.read\_csv(f,sep=';') for f in all\_filenames ])

all\_file

# To save the combined dataframe

all\_file .to\_csv("fichier\_combine.csv", sep=';',index=False, encoding='utf-8-sig')

# To read a january data

data\_january = pd.read\_csv(path+'/supermarket\_sales\_January.csv',sep =';')

data\_january

# To read a january data

all\_file.shape

all\_file.dtypes

# To show january 5 first lines

data\_january.head()

# enumerate the number of null lines

all\_file.isnull().sum()

# delete the missed values

all\_file = all\_file.dropna()

all\_file.shape

# to show the 5 last lines

all\_file.head(3)

**PART 2 : Marketing answers**

# Question 1 : Which mounth did we achieve the turn over.?

# Convert the data column to datatime format

all\_file['Date']= pd.to\_datetime(all\_file['Date'])

all\_file['Date']

# To create a new column 'Mont\_sales' contauning the monbt extracted from the date column

all\_file['Month\_sales']= all\_file['Date'].dt.month

all\_file

# to check all mounth , I hace in my database

all\_file['Month\_sales'].unique()

# to verify the data type of the comumn Month\_sales

all\_file.dtypes

# To calculte the turn over of all month

all\_file['Sales']= all\_file['Quantity'] \* all\_file['Unit price']

all\_file

# To calculate the tur over by month

all\_file.groupby('Month\_sales')['Sales'].sum()

# Create a better visual to check a turn over by month

months = range(1,13)

plt.bar(months,all\_file.groupby('Month\_sales')['Sales'].sum())

plt.xticks(months)

plt.ylabel('Sales in USD')

plt.ylabel('Month number')

plt.show()

# Conclusion 1

The best sales are in January.

# Witch town , you realize the maximum ordder ?

all\_file.groupby('City')['City'].count()

colors = ['green','pink','yellow']

plt.bar(all\_file.groupby('City')['City'].count().index,all\_file.groupby('City')['City'].count().values,color =colors)

plt.xticks(rotation ='horizontal')

plt.ylabel('Received Orders')

plt.xlabel('City names')

plt.show()

Conclusion 2 : The city that realize the maximum orders is Yango, after comes Mandalay and, finally Naypitaw.

# When should we run an advertising campaign to have more sales?

all\_file

# create hour only column

all\_file['Hour']=pd.to\_datetime(all\_file['Time']).dt.hour

all\_file

keys = []

Hours = []

for key,hour in all\_file.groupby('Hour'):

keys.append(key)

Hours.append(len(hour))

Hours

plt.grid()

plt.plot(keys,Hours)

plt.xlabel('Heure de la journée')

plt.ylabel('Nombre de commandes')

Concluion 3 : The best time to start a marketing campagn to grow up the turn over and the orders in at 10 H A.M o'clock.

# Which product line a genrates the most sales ?

all\_file.groupby('Product line')['Quantity'].sum()

all\_file.groupby('Product line')['Quantity'].sum().plot(kind='bar')

Conclusion 4 : The line product thats genrates the most sales is heath and beauty

# Let's go to understanding why heath and beauty genrates the most sales

## impact price

all\_file .groupby('Product line')['Unit price'].mean()

products = all\_file.groupby('Product line')['Quantity'].sum().index

quantity =all\_file .groupby('Product line')['Quantity'].sum()

prices = all\_file .groupby('Product line')['Unit price'].mean()

plt.figure(figsize=(40,24))

fif,ax1=plt.subplots()

ax2=ax1.twinx()

ax1.bar(products,quantity,color='g')

ax2.plot(products,prices,'r')

ax1.set\_xticklabels(products,rotation= 'vertical',size=8)

Concluison 5 : The cheaper product line are the most sold.

Customer\_type = all\_file.groupby('Product line')['Customer type'].sum().index

gender =all\_file .groupby('Product line')['Gender'].sum()

sns.countplot(data=all\_file,x="Gender", hue='Customer type')

plt.title('Distribution of clients by type and Sexe')

plt.show()