Exercice Simplon :

1.IMPORT DE LA BASE DE DONNÉES VENTES SUR SQLITEONLINE

But : charger les données CSV dans une table SQL nommée `ventes` pour pouvoir interroger avec du SQL.

2.VÉRIFIER QUE LA BASE A ÉTÉ IMPORTER

```sql
SELECT *
FROM ventes;
```

date	produit	prix	qte	region
2022-01-01	Produit A	10	100	Nord
2022-01-02	Produit B	15	50	Nord
2022-01-02	Produit A	10	75	Sud
2022-01-03	Produit C	20	30	Nord
2022-01-03	Produit A	10	150	Sud
2022-01-04	Produit B	15	75	Nord
2022-01-04	Produit C	20	50	Sud
2022-01-05	Produit A	10	125	Nord

3. CRÉEZ LES REQUÊTES SQL POUR RÉPONDRE AUX QUESTIONS CLÉS SUR LES VENTES DE L'ENTREPRISE :

a. le chiffre d'affaires total :

Run SQLite

```sql
SELECT SUM(prix * qte) AS chiffre_affaires_total
FROM ventes;
```

chiffre_affaires_total
44825

Résultat : 44825

CA par produit :

```sql
SELECT produit,SUM(qte*prix) AS total_CA
FROM ventes
GROUP BY produit;
```

produit	total_CA
Produit A	17500
Produit B	15825
Produit C	11500

b. les ventes par produit :

```sql
SELECT produit,SUM(qte) AS total_ventes
FROM ventes
GROUP BY produit;
```

produit	total_ventes
Produit A	1750
Produit B	1055
Produit C	575

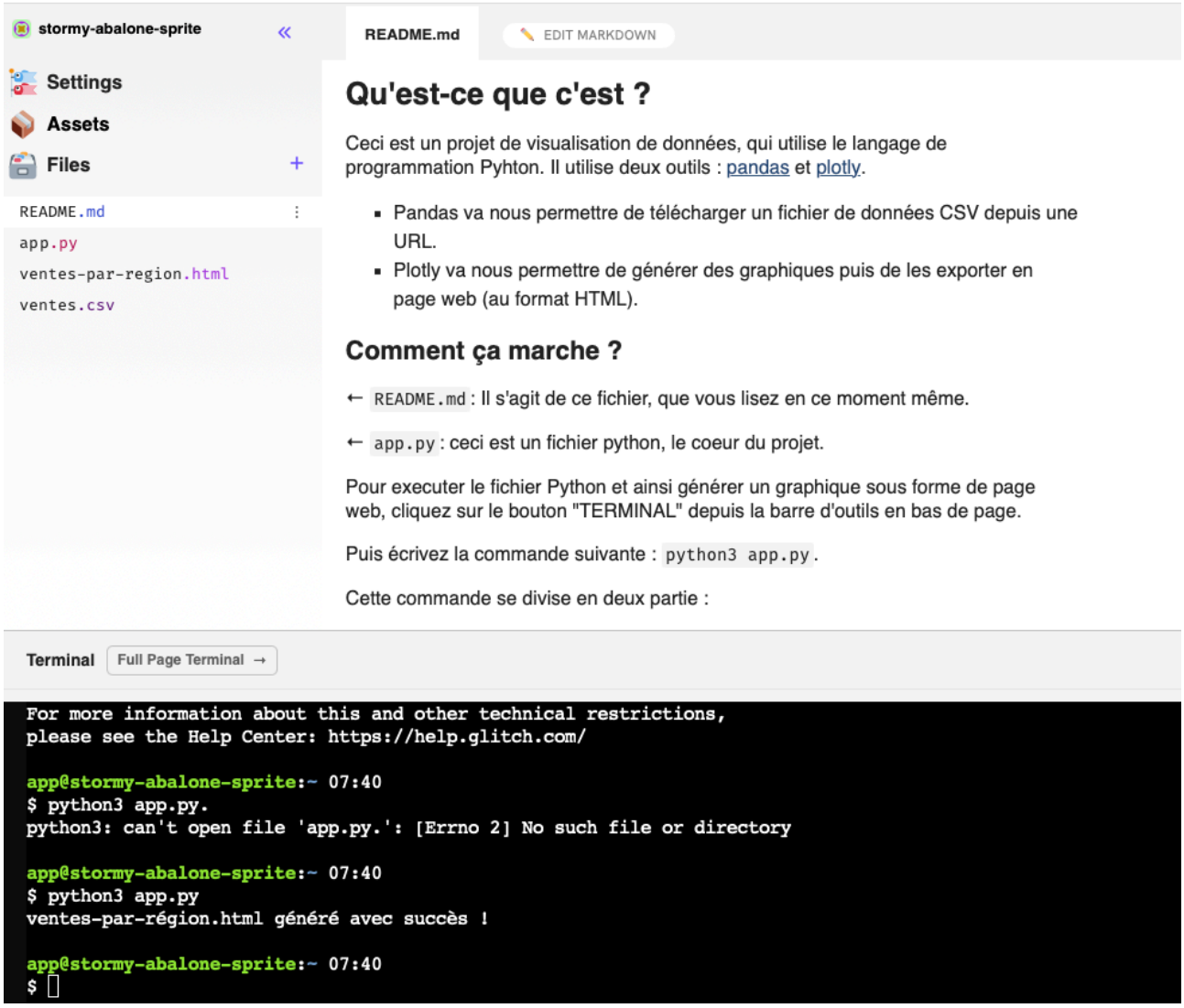
c. les ventes par région.

```sql
SELECT region,SUM(qte) AS ventes_region
FROM ventes
GROUP BY region;
```

region	ventes_region
Nord	1605
Sud	1775

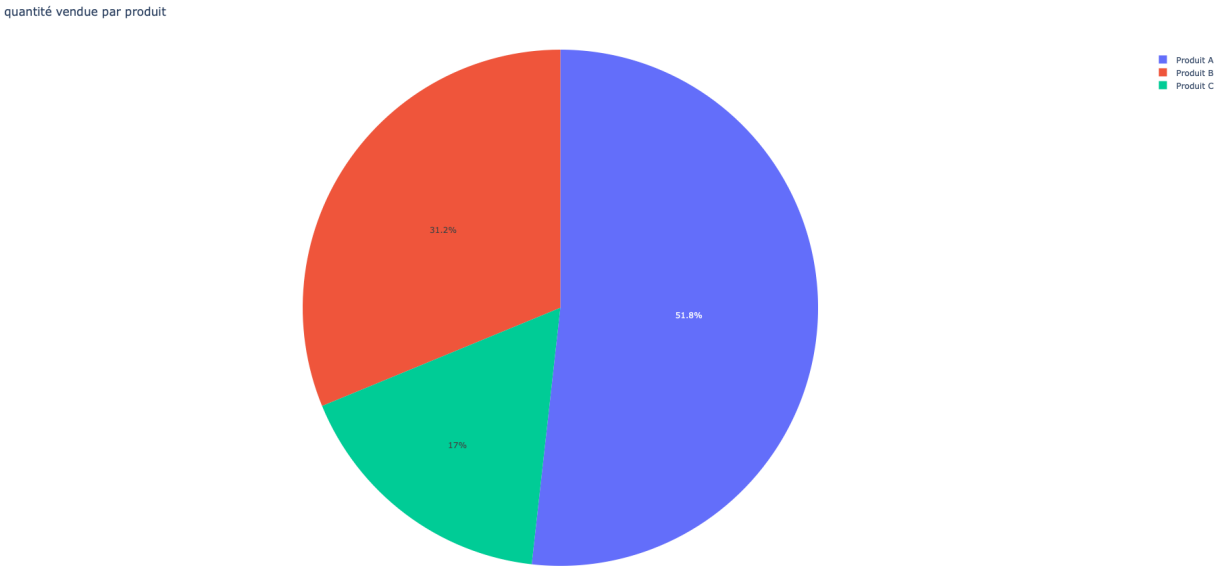
4. RENDEZ-VOUS SUR GLITCH, UN LOGICIEL EN LIGNE PERMETTANT DE PROGRAMMER DES SCRIPTS ET DES APPLICATIONS WEB, EN DUPLIQUANT (“REMIXER”) LA BASE DE PROJET PYTHON.

5. LISEZ LES INSTRUCTIONS DU FICHIER “README.MD”.

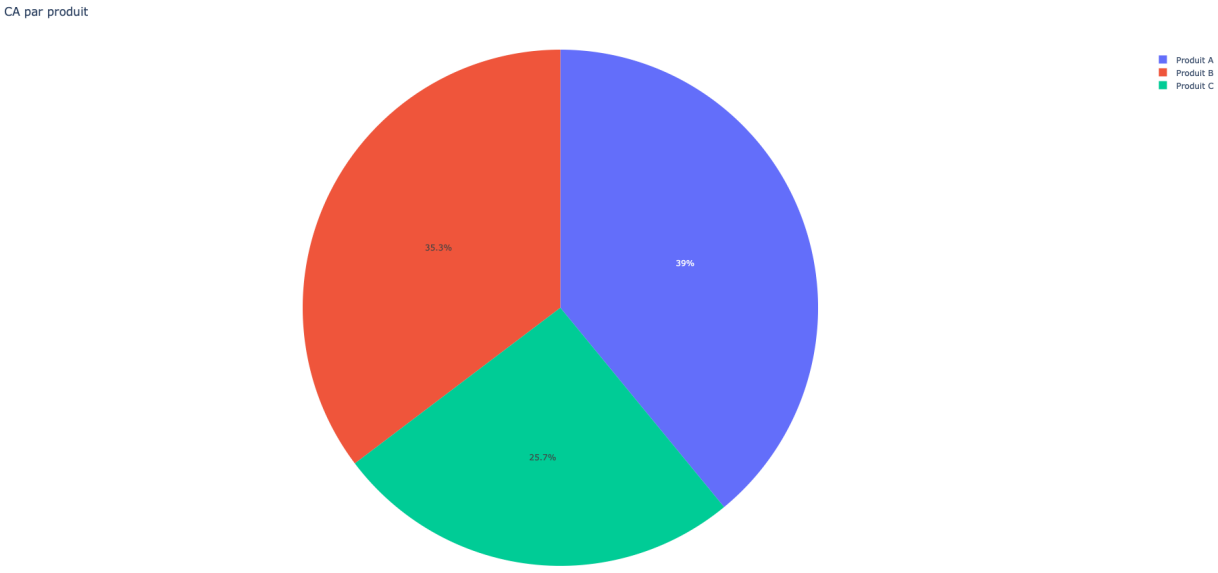


6. EN VOUS APPUYANT SUR L'EXEMPLE DONNÉ, CRÉER DEUX NOUVEAUX GRAPHIQUES :

a. les ventes par produit,



b. le chiffre d'affaires par produit.



ANNEXE : CODE TAPÉ SUR GLITCH :

```
# import de plotly pour tracer les graphiques
import plotly.express as px

# import de pandas pour manipuler les données
import pandas as pd

#télécharger puis lecture du fichier CSV ventes depuis une URL grace à pandas.
données = pd.read_csv('https://docs.google.com/spreadsheets/d/e/
2PACX-1vSC4KusfFzvOsr8WJRgozzsCxrELW4G4PopUkiDbvrrV2lg0S19-
zeryp02MC9WYSVBuzGCUtn8ucZW/pub?output=csv')

#créér le graphique"vente par produit" avec les données voulues grace à plotly
figure = px.pie(données, values='qte', names='produit', title='quantité vendue par produit')

#exporter le graphique "vente par produit" en format page web grace à plotly
figure.write_html('ventes-par-produit.html')

#afficher le message "vente par produit" généré avec succès apres execution du code
print('ventes-par-produit.html généré avec succès !')

#créér le graphique"vente par region" avec les données voulues grace à plotly
figure = px.pie(données, values='qte', names='region', title='quantité vendue par region')

#exporter le graphique "vente par region" en format page web grace à plotly
figure.write_html('ventes-par-region.html')

#afficher le message "vente par region" généré avec succès apres execution du code
print('ventes-par-region.html généré avec succès !')

# Calcul du chiffre d'affaires par produit (qte * prix)
données['chiffre_affaire'] = données['qte'] * données['prix']

#créér le graphique"CA par produit" avec les données voulues grace à plotly
figure = px.pie(données, values= 'chiffre_affaire', names='produit', title='CA par produit')

#exporter le graphique "CA par produit" en format page web grace à plotly
figure.write_html('CA-par-produit.html')

#afficher le message "CA par produit" généré avec succès apres execution du code
print('CA-par-produit.html généré avec succès !')
```