

# Projet Portfolio : Analyse de ventes avec Python, Jupyter Notebook et SQL

## Objectif

Créer un projet interactif combinant **Python**, **Jupyter Notebook** et **SQL** pour analyser les ventes d'une entreprise fictive. Mettre en avant vos compétences en **requêtes SQL**, manipulation de données et visualisation.

## Structure du projet

### 1. Introduction

- Présentation du projet et des objectifs : identifier les produits les plus vendus, chiffre d'affaires par région, clients fidèles.

### 3. Création d'une base SQL

- Utilisation de **SQLite** pour stocker les données.

- Tables possibles :

- clients(id, nom, ville, date\_inscription)
- produits(id, nom, catégorie, prix\_unitaire)
- ventes(id, id\_client, id\_produit, quantite, date\_achat)

### 6. Insertion de données

- Données fictives générées via Python ( pandas ou faker ) ou import de datasets CSV publics.

### 8. Requêtes SQL dans Jupyter Notebook

- Exemples :

- Sélection de données : `SELECT * FROM ventes LIMIT 10;`
- Chiffre d'affaires par produit :

```
SELECT nom, SUM(quantite*prix_unitaire) AS chiffre_affaires
FROM ventes
JOIN produits ON ventes.id_produit = produits.id
GROUP BY nom
ORDER BY chiffre_affaires DESC;
```

- Clients par ville : `SELECT ville, COUNT(DISTINCT id_client) FROM clients GROUP BY ville;`

### 10. Manipulation et visualisation Python

11. Charger les résultats SQL dans `pandas` avec `pd.read_sql_query()`.
12. Graphiques avec `matplotlib` ou `seaborn` : évolution du chiffre d'affaires, top 10 clients, histogramme par catégorie produit.
13. **Conclusion**
14. Insights business tirés des analyses.
15. Recommandations possibles pour l'entreprise.

## Pourquoi ce projet est pertinent

- **SQL + Python + Jupyter** dans un seul projet.
- Démonstration de **jointures, agrégations et visualisations**.
- Concret et compréhensible par un recruteur.
- Scalable : possibilité d'ajouter des requêtes SQL avancées pour plus de complexité.