Université Sultan Moulay Slimane Faculté Polydisciplinaire Béni Mellal
Département INFORMATIQUE (MIP)

Filière : Science de données et sécurité des systèmes d'information
A.U : 2023-2024

Sujet

# Compte Rendu de TP06 – Python pour la science des données

Présenté Par :
MAFTOUH Omar

Encadré Par :
Z. BOUSALEM
Y. MADANI

## Introduction :

L'analyse des données immobilières est cruciale pour comprendre les tendances du marché et prendre des décisions éclairées en matière d'investissement immobilier. Dans ce compte rendu, nous explorons un ensemble de données immobilières en suivant une approche méthodique. Nous commençons par importer les bibliothèques nécessaires, charger les données, puis explorer leur structure et leur qualité. Nous analysons ensuite les statistiques descriptives, la distribution des prix, la corrélation entre les caractéristiques, et visualisons les données à l'aide de graphiques. En interprétant les résultats, nous tirons des conclusions sur les dynamiques du marché, les variations de prix selon les quartiers et les caractéristiques des propriétés, ainsi que l'évolution des prix au fil du temps. Ces informations sont précieuses pour les professionnels de l'immobilier, les investisseurs et les acheteurs potentiels.

## Application :

### Question 01 Importer les bibliothèques nécessaires :

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sb
```

### Question 02 Charger le fichier de données :

```
df = pd.read_csv('donnees_maisons_T.csv')
```

### Question 03 Exploration initiale des données :

```
# 3-1 Les 5 premier ligne
print(f"5 permier ligne du df : \n {df[ : 5]}")

# 3-2 les types de données de chaque colonne
print(f"Les types de données de chaque colone : \n{df.dtypes}")

# 3-2 Vérifier s'il y a des valeurs manquantes
print(f"Valeurs manquantes : \n{df.isnull}")

# 3-4 Le nombre de caractéristiques (colonnes)
print(f"Nombre de colone : {df.shape[1]}")

# 3-5 Le nombre de caractéristiques (colonnes)
print(f"Nombre de ligne : {df.shape[0]}")
```

### Interprétation des résultats :

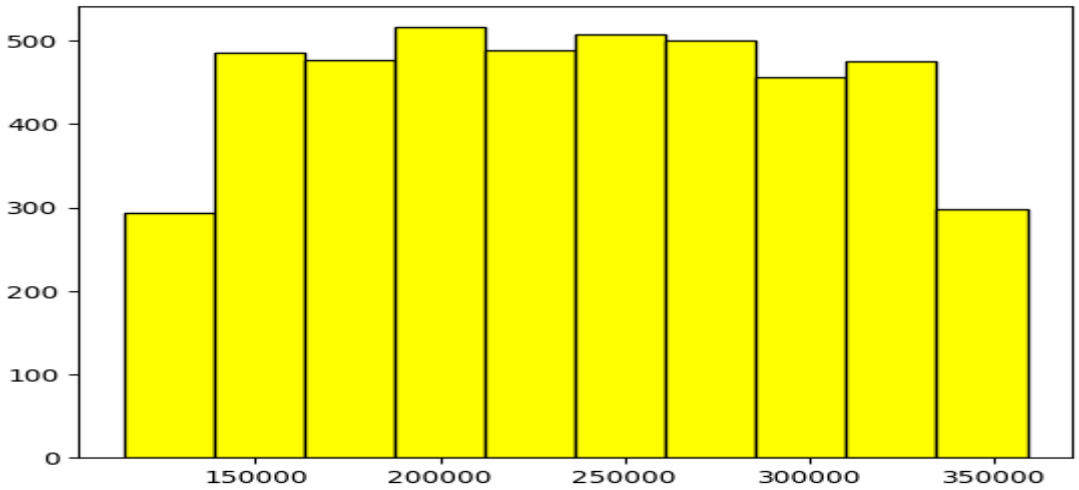
Les données présentent un total de 5000 entrées, chacune caractérisée par 8 attributs comprenant des informations sur les quartiers, la surface, le nombre de chambres, l'année de construction, la distance au centre-ville, la qualité du quartier, l'adresse et le prix. Les types de données indiquent principalement des valeurs numériques, à l'exception des colonnes "Quartier" et "Adresse". Quelques valeurs manquantes sont observées dans la colonne "Quartier" et quelques autres dans la colonne "Prix". Ces lacunes nécessiteront un traitement pour une analyse plus complète et précise des données.

- **Question 04 Afficher les statistiques descriptives pour les caractéristiques numériques :**

```
df.describe()
```

- **Question 05 Afficher la distribution des prix des maisons en utilisant un histogramme :**

```
plt.hist(df["Prix"] , bins=10 , color="yellow" , edgecolor = 'black' ,
label='Distribution Prix')
```




- **Question 6 la matrice de corrélation entre les caractéristiques numériques :**

```
# 12 Résumez le DataFrame
print(f"Matrice de corrélation : {df.corr(numeric_only = True)}")
```

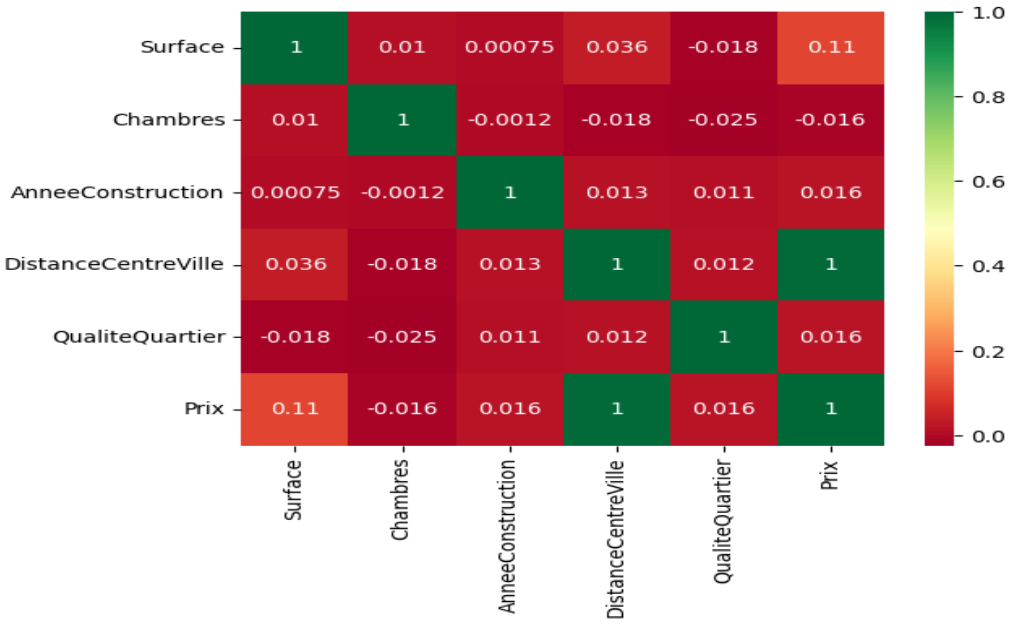
- **Résultat :**

```
Matrice de corrélation :                          Surface  Chambres
AnneeConstruction  \
Surface              1.000000  0.010449           0.000752
Chambres             0.010449  1.000000          -0.001222
AnneeConstruction    0.000752 -0.001222           1.000000
DistanceCentreVille  0.035519 -0.018213           0.013154
QualiteQuartier     -0.018375 -0.024590           0.011255
Prix                 0.113083 -0.015644           0.016346

                     DistanceCentreVille  QualiteQuartier      Prix
Surface                         0.035519        -0.018375  0.113083
Chambres                       -0.018213        -0.024590 -0.015644
AnneeConstruction               0.013154         0.011255  0.016346
DistanceCentreVille             1.000000         0.011824  0.995717
QualiteQuartier                 0.011824         1.000000  0.015966
Prix                            0.995717         0.015966  1.000000
```

### Question 7 une heatmap de la matrice de corrélation :

```
sb.heatmap(df.corr(numeric_only = True) , annot=True , cmap="RdYlGn")
```




### Interprétation :

Corrélation positive : Les variables Surface et Chambres sont positivement corrélées avec le Prix.
Corrélation négative : AnneeConstruction est négativement corrélée avec le Prix.
Variables indépendantes : DistanceCentreVille a une faible corrélation avec les autres variables.

### Question 8 Le décompte du nombre de maisons dans chaque quartier :

```
df.groupby("Quartier")["Quartier"].value_counts()
```

### Résultat :

```
Quartier
Alatlas       638
Almassira1    624
Elhouda       637
Elkasba       613
Haycharaf     634
RiadSalam     632
Sisalem       629
Taqaddom      583
Name: count, dtype: int64
```

**Question 9 le prix moyen des maisons pour chaque quartier :**

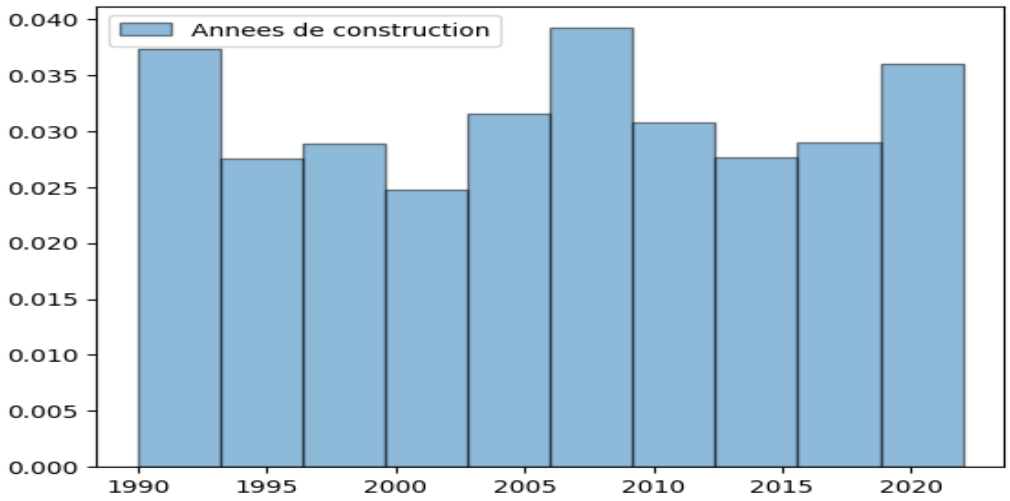
```
df.groupby("Quartier")["Prix"].mean()
```

**Résultat :**

```
Quartier
Alatlas       234621.100694
Almassira1    240180.588542
Elhouda       234094.189895
Elkasba       239130.454206
Haycharaf     239779.487544
RiadSalam     234829.789381
Sisalem       233970.500000
Taqaddom      232806.571970
Name: Prix, dtype: float64
```

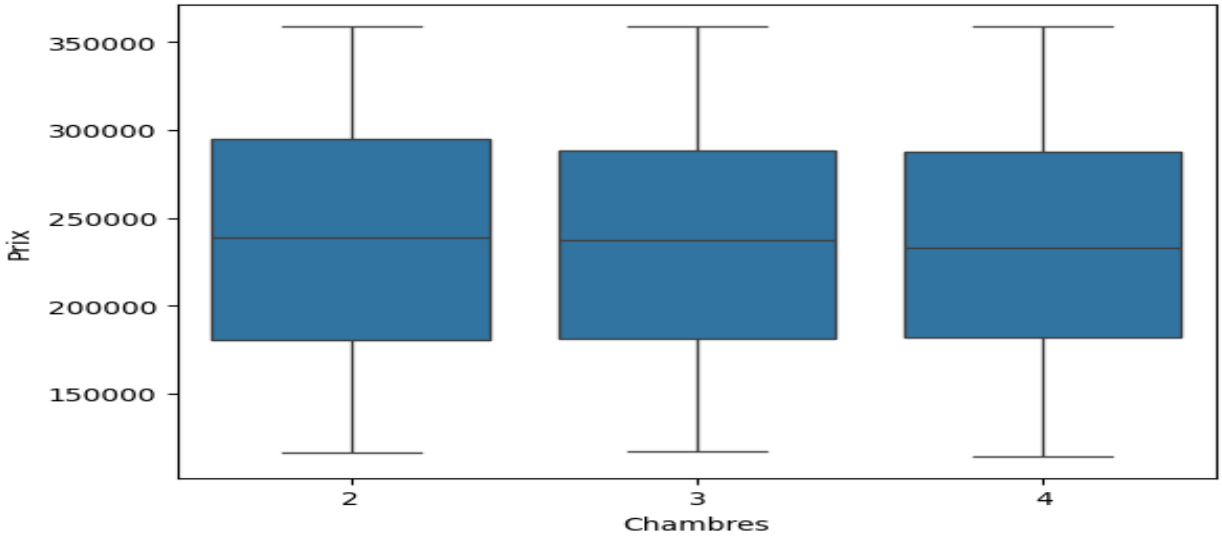
**Question 10 un histogramme de la répartition des années de construction:**

```
plt.hist(df['AnneeConstruction'] , density = True , edgecolor = "black" ,
alpha = 0.5 , label = "Annees de construction")
plt.legend()
```




**Question 11 la variation du prix en fonction du nombre de chambres :**

```
sb.boxplot(x=df["Chambres"] , y=df["Prix"] , data=df)
```



- **Question 12 les données manquantes dans la colonne 'Prix' avec la moyenne :**

```
df["Prix"].isnull().sum().mean()
```

- **Question 13 Les données manquantes dans les colonnes Surface et DistanceCentreVille avec la médiane:**

```
df.loc[:,["Surface" , "DistanceCentreVille"]].isnull().sum().median()
```

- **Question 14 Remplacer les données manquantes dans la colonne "Quartier" par la dernière valeur non manquante :**

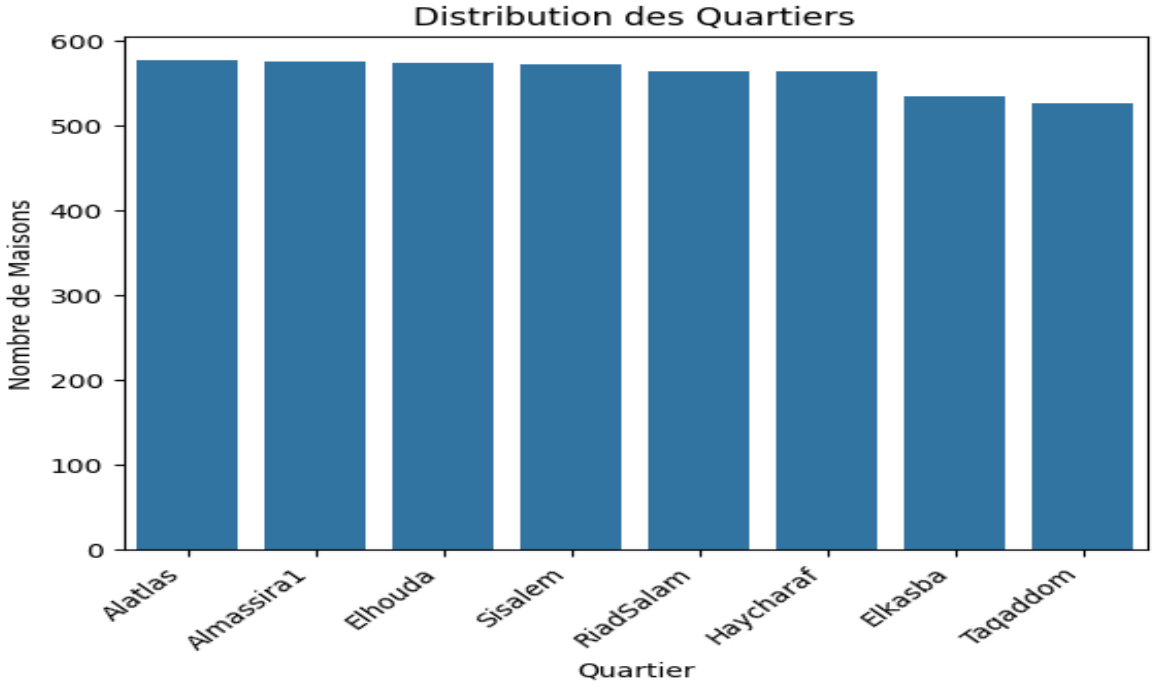
```
df["Quartier"].fillna(method="ffill" , inplace=True)
```

- **Question 15 Supprimer les lignes contenant des données manquantes:**

```
df.dropna(axis=0 , inplace=True)
```

- **Question 16 Visualiser la distribution des quartiers avec un diagramme en barres:**

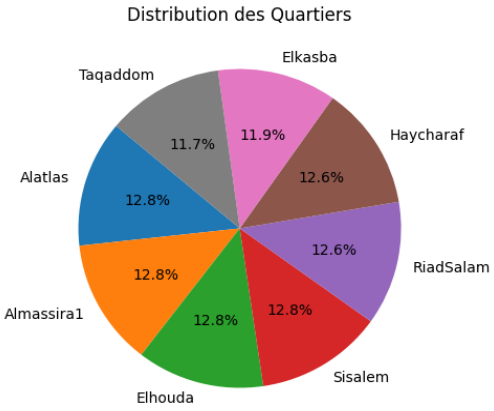
```
sb.countplot(x='Quartier', data=df, order=df['Quartier'].value_counts().index)
plt.xticks(rotation=45, ha='right')
plt.title('Distribution des Quartiers')
plt.xlabel('Quartier')
plt.ylabel('Nombre de Maisons')
plt.show()
```



- **Question 17 Visualiser la distribution des quartiers avec un diagramme en secteurs (pie chart) :**

```
distribution_quartiers = df['Quartier'].value_counts()

plt.figure(figsize=(5, 5))
plt.pie(distribution_quartiers, labels=distribution_quartiers.index,
autopct='%1.1f%%', startangle=140)
plt.title('Distribution des Quartiers')
plt.show()
```




- **Question 18 Extraire les villes à partir des adresses en ajoutant une nouvelle colonne « Nom Ville » :**

```
df['Nom Ville'] = df['Adresse'].str.extract(r"Rue\s\D\s([A-Za-z]+\D[A-Za-
z]+)")
```

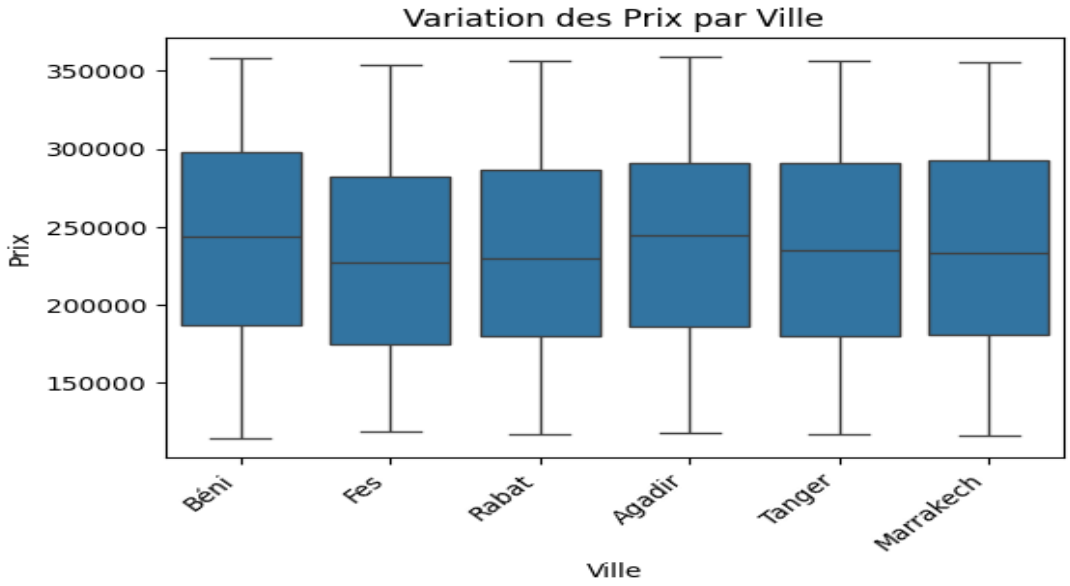
- **Question 19 Supprimer les espaces de début de chaque valeur dans la colonne Nom Ville :**

```
df['Nom Ville'] = df['Nom Ville'].str.strip()
```

- **Question 20 Visualiser les prix par ville avec une boîte à moustaches :**

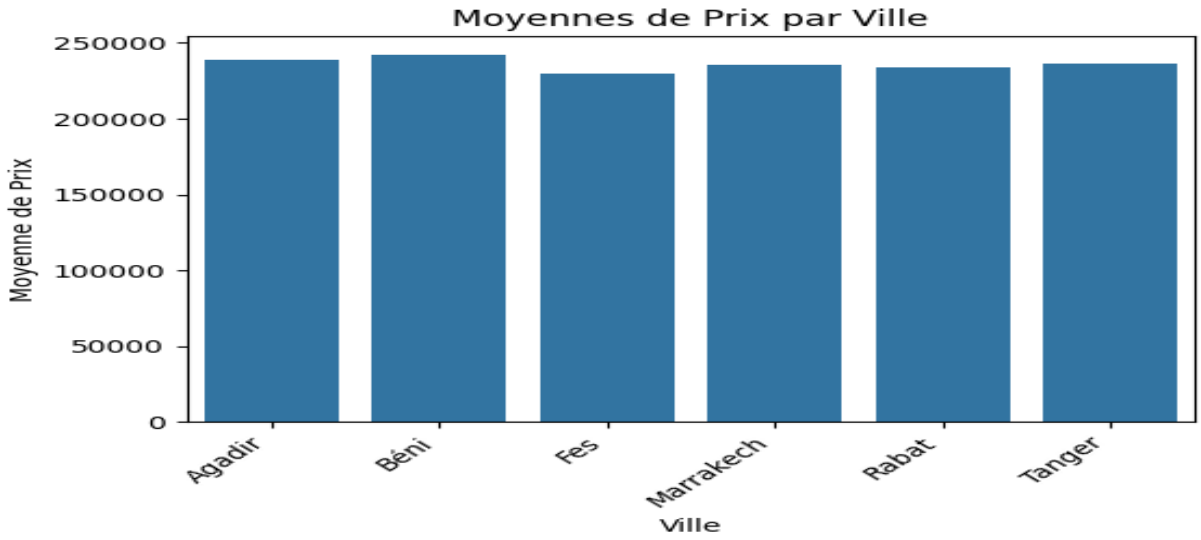
```
df['Prix'] = pd.to_numeric(df['Prix'], errors='coerce')

plt.figure(figsize=(6, 4))
sb.boxplot(x='Nom Ville', y='Prix', data=df)
plt.xticks(rotation=45, ha='right')
plt.title('Variation des Prix par Ville')
plt.xlabel('Ville')
plt.ylabel('Prix')
plt.show()
```




- **Question 21 Visualiser les moyennes de prix par ville avec un diagramme en barres :**

```
df['Prix'] = pd.to_numeric(df['Prix'], errors='coerce')
moyennes_prix_par_ville = df.groupby('Nom Ville')['Prix'].mean().reset_index()
plt.figure(figsize=(6, 4))
sb.barplot(x='Nom Ville', y='Prix', data=moyennes_prix_par_ville)
plt.xticks(rotation=45, ha='right')
plt.title('Moyennes de Prix par Ville')
plt.xlabel('Ville')
plt.ylabel('Moyenne de Prix')
plt.show()
```

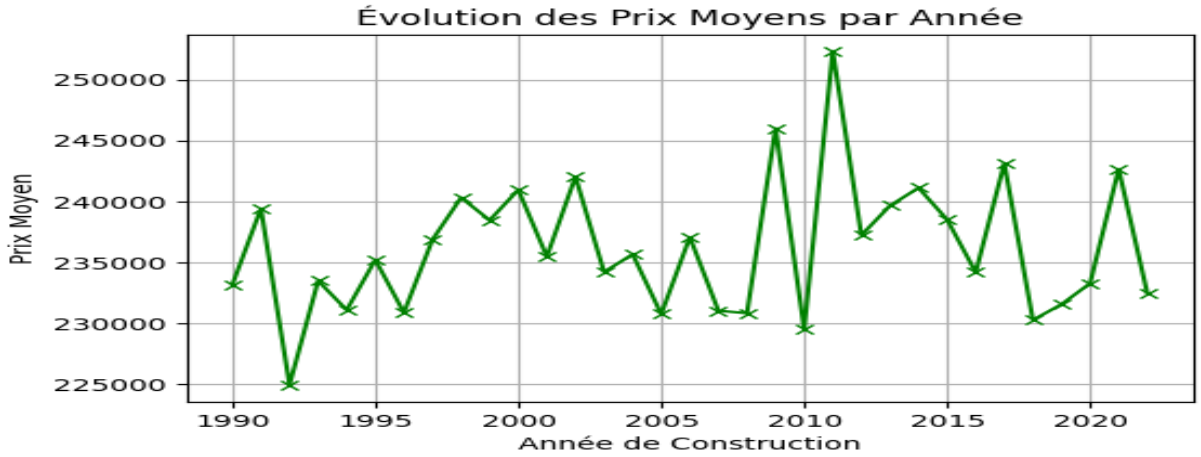


- **Question 22 Analyser comment les prix évoluent au fil des années avec un diagramme en ligne :**

```
df['Prix'] = pd.to_numeric(df['Prix'], errors='coerce')
df['AnneeConstruction'] = pd.to_numeric(df['AnneeConstruction'],
errors='coerce')

moyennes_prix_par_annee =
df.groupby('AnneeConstruction')['Prix'].mean().reset_index()
moyennes_prix_par_annee =
moyennes_prix_par_annee.sort_values(by='AnneeConstruction')

plt.figure(figsize=(6, 4))
plt.plot(moyennes_prix_par_annee['AnneeConstruction'],
moyennes_prix_par_annee['Prix'],
        marker='x', color='green', linestyle='-')
plt.title('Évolution des Prix Moyens par Année')
plt.xlabel('Année de Construction')
plt.ylabel('Prix Moyen')
plt.grid(True)
plt.show()
```




```
df.to_csv('TP6.csv', index=False)
```

## Conclusion :

En conclusion, l'analyse des données immobilières a permis d'identifier des tendances significatives, telles que les relations entre les caractéristiques des propriétés et les variations de prix selon les quartiers et les années de construction. Ces informations sont essentielles pour les professionnels de l'immobilier et les investisseurs, fournissant une base solide pour la prise de décisions éclairées dans le marché immobilier.