**تمارين العملي (التطبيق ص51)**

**عمل الطالبة: تحت اشراف:**

**أزهار احمد د . حسين الخضر**

* **لديك ملف للبيعات باسم sales\_data.csv ويحتوي على 10000 سجل المبيعات منتجات , المطلوب:**

**A : تحميل البيانات من الملف CSV واستكشافها:**

**صور لمخرجات الكود**

**B : معالجة البيانات المفقودة:**

**صور لمخرجات الكود**

**: C Visualization التصورات الأساسية :**

1. **توزيع المبيعات (Histogram):**

**صور لمخرجات الكود**

1. **المبيعات حسب المنطقة (Bar Chart):**

**صور لمخرجات الكود**

1. **علاقة عمر العميل بالمبيعات (Scatter Plot):**

**صور لمخرجات الكود**

**D : تحليل متقدم:**

1. **مخطط الصندوق للمبيعات حسب المنتج (التوزيع الطبيعي) والمنحرف:**

**صور لمخرجات الكود**

1. **تحليل (ANOVA):**

**صور لمخرجات الكود**

1. **المبيعات عبر الزمن Lane Tout:**

**صور لمخرجات الكود**

**E: تحليل الارتباط (Heatmap):**

**صور لمخرجات الكود**

**F: تمارين أخرى على نفس ملف البيانات Sales\_data.csv:**

1. **أنشئ مخطط دائرياً يوضح نسبة المبيعات في كل منطقة:**

**صور لمخرجات الكود**

1. **بين متوسط اعمار العملاء لكل منتج باستخدام مخطط الاعمدة:**

**صور لمخرجات الكود**

1. **استخرج اعلى خمسة أيام من حيث المبيعات و ارسمها:**

**صور لمخرجات الكود**

* **الكود كاملاً لكل الفقرات:-**
* **ملاحظة:-**

**نسخت كل الكود في أجزاء منفصلة نظراً لأنه كبير ولمعرفته بشكل مرتب سوف ارفع لك ملف الكود كاملاً في مجلد باسم تكليف العملي س 51\_ ازهار ويوجد بداخله ملف الكود كاملاً باسم F.py .**

import pandas as pd  
import numpy as np  
import matplotlib.pyplot as plt  
import seaborn as sns  
from scipy import stats  
from statsmodels.formula.api import ols  
from statsmodels.stats.anova import anova\_lm  
from datetime import datetime  
  
  
df = pd.read\_csv('sales\_data.csv')  
print("معلومات عن البيانات:")  
print(df.info())  
print("\nإحصائيات وصفية للبيانات:")  
print(df.describe())  
print("\nعرض أول 5 صفوف من البيانات:")  
print(df.head())  
print("\nعدد القيم المفقودة في كل عمود:")  
print(df.isnull().sum())  
df['Sales'] = df.groupby('Product')['Sales'].transform(lambda x: x.fillna(x.mean()))  
  
# إذا بقي أي قيم مفقودة (للمنتجات التي لا تحتوي على أي مبيعات مسجلة)، نملؤها بالصفر  
df['Sales'] = df['Sales'].fillna(0)  
  
# ملء القيم المفقودة في عمود المنتج بالقيمة "Unknown"  
df['Product'] = df['Product'].fillna('Unknown')  
  
# ملء القيم المفقودة في عمود المنطقة بالقيمة الأكثر تكراراً  
most\_common\_region = df['Region'].mode()[0]  
df['Region'] = df['Region'].fillna(most\_common\_region)  
  
# ملء القيم المفقودة في عمود عمر العميل بالمتوسط  
df['Customer\_Age'] = df['Customer\_Age'].fillna(df['Customer\_Age'].mean())  
  
# التحقق من عدم وجود قيم مفقودة بعد المعالجة  
print("\nعدد القيم المفقودة بعد المعالجة:")  
print(df.isnull().sum())  
  
plt.figure(figsize=(10, 6))  
plt.hist(df['Sales'], bins=50, color='skyblue', edgecolor='black')  
plt.title('توزيع المبيعات')  
plt.xlabel('مقدار المبيعات')  
plt.ylabel('عدد المعاملات')  
plt.grid(True, alpha=0.3)  
plt.savefig('sales\_distribution.png')  
plt.show()

plt.figure(figsize=(10, 6))  
region\_sales = df.groupby('Region')['Sales'].sum().sort\_values(ascending=False)  
region\_sales.plot(kind='bar', color=['#1f77b4', '#ff7f0e', '#2ca02c', '#d62728'])  
plt.title('إجمالي المبيعات حسب المنطقة')  
plt.xlabel('المنطقة')  
plt.ylabel('إجمالي المبيعات')  
plt.xticks(rotation=45)  
plt.grid(True, alpha=0.3)  
plt.savefig('sales\_by\_region.png')  
plt.show()  
  
plt.figure(figsize=(10, 6))  
plt.scatter(df['Customer\_Age'], df['Sales'], alpha=0.5, color='green')  
plt.title('علاقة عمر العميل بالمبيعات')  
plt.xlabel('عمر العميل')  
plt.ylabel('مقدار المبيعات')  
plt.grid(True, alpha=0.3)  
plt.savefig('age\_vs\_sales.png')  
plt.show()  
  
plt.figure(figsize=(12, 8))  
top\_products = df['Product'].value\_counts().nlargest(10).index  
df\_top\_products = df[df['Product'].isin(top\_products)]  
sns.boxplot(x='Product', y='Sales', data=df\_top\_products)  
plt.title('توزيع المبيعات لأهم 10 منتجات')  
plt.xlabel('المنتج')  
plt.ylabel('المبيعات')  
plt.xticks(rotation=45)  
plt.grid(True, alpha=0.3)  
plt.savefig('sales\_distribution\_by\_product.png')  
plt.show()

# تحليل ANOVA لمعرفة ما إذا كانت هناك فروق ذات دلالة إحصائية في المبيعات بين المناطق  
model = ols('Sales ~ C(Region)', data=df).fit()  
anova\_results = anova\_lm(model)  
  
print("\nنتائج تحليل ANOVA للمبيعات بين المناطق:")  
print(anova\_results)  
  
# تفسير النتائج  
alpha = 0.05  
if anova\_results['PR(>F)'][0] < alpha:  
 print("\nهناك فرق ذو دلالة إحصائية في المبيعات بين المناطق (p-value = {:.4f})".format(anova\_results['PR(>F)'][0]))  
else:  
 print("\nلا يوجد فرق ذو دلالة إحصائية في المبيعات بين المناطق (p-value = {:.4f})".format(anova\_results['PR(>F)'][0]))  
  
 # تحويل العمود Date إلى نوع datetime  
 df['Date'] = pd.to\_datetime(df['Date'])  
  
 # حساب المبيعات اليومية  
 daily\_sales = df.groupby('Date')['Sales'].sum()  
  
 plt.figure(figsize=(12, 6))  
 daily\_sales.plot(color='purple')  
 plt.title('المبيعات اليومية عبر الزمن')  
 plt.xlabel('التاريخ')  
 plt.ylabel('إجمالي المبيعات اليومية')  
 plt.grid(True, alpha=0.3)  
 plt.savefig('daily\_sales\_trend.png')  
 plt.show()  
  
 # حساب الارتباط بين الأعمدة العددية  
 corr = df[['Sales', 'Customer\_Age']].corr()  
  
 plt.figure(figsize=(8, 6))  
 sns.heatmap(corr, annot=True, cmap='coolwarm', center=0)  
 plt.title('خريطة الحرارة لتحليل الارتباط')  
 plt.savefig('correlation\_heatmap.png')  
 plt.show()  
  
 plt.figure(figsize=(8, 8))  
 region\_sales.plot(kind='pie', autopct='%1.1f%%', startangle=90, colors=['#ff9999', '#66b3ff', '#99ff99', '#ffcc99'])  
 plt.title('نسبة المبيعات في كل منطقة')  
 plt.ylabel('')  
 plt.savefig('sales\_pie\_chart.png')  
 plt.show()

plt.figure(figsize=(12, 6))  
product\_age = df.groupby('Product')['Customer\_Age'].mean().sort\_values(ascending=False).head(10)  
product\_age.plot(kind='bar', color='orange')  
plt.title('متوسط أعمار العملاء لأهم 10 منتجات')  
plt.xlabel('المنتج')  
plt.ylabel('متوسط عمر العميل')  
plt.xticks(rotation=45)  
plt.grid(True, alpha=0.3)  
plt.savefig('average\_age\_by\_product.png')  
plt.show()  
  
top\_days = daily\_sales.nlargest(5)  
  
plt.figure(figsize=(10, 6))  
top\_days.plot(kind='bar', color='green')  
plt.title('أعلى 5 أيام من حيث المبيعات')  
plt.xlabel('التاريخ')  
plt.ylabel('إجمالي المبيعات')  
plt.xticks(rotation=45)  
plt.grid(True, alpha=0.3)  
plt.savefig('top\_sales\_days.png')  
plt.show()