

شرح تكليف مروة الهادي :

---

### 1. استيراد المكتبات اللازمة

```
import numpy as np
```

```
import pandas as pd
```

```
import matplotlib.pyplot as plt
```

```
import seaborn as sns
```

- **numpy**: تُستخدم لإنشاء بيانات رقمية ومعالجتها، مثل الأعمار والرواتب والخبرة.
- **pandas**: تُستخدم لإنشاء إطارات بيانات (DataFrames) لتنظيم البيانات وتحليلها.
- **matplotlib.pyplot**: تُستخدم لإنشاء الرسوم البيانية.
- **seaborn**: مكتبة مبنية على **matplotlib** تُستخدم لإنشاء رسوم بيانية جميلة وإحصائية.

---

### 2. إنشاء بيانات افتراضية

```
np.random.seed(42) # لضمان تكرار النتائج
```

```
n_records = 10000
```

```
data = {
```

```
    "Age": np.random.randint(18, 65, n_records),
```

```
    "Salary": np.random.randint(3000, 15000, n_records),
```

```
    "Experience": np.random.randint(1, 30, n_records),
```

```
    "Department": np.random.choice(["HR", "IT", "Finance", "Marketing"], n_records)
```

```
}
```

```
df = pd.DataFrame(data)
```

- **np.random.seed(42)**: يجعل النتائج عشوائية ثابتة (قابلة للتكرار).
- **n\_records = 10000**: يُحدد عدد السجلات التي سيتم إنشاؤها.
- **data**: قاموس يحتوي على الأعمدة التالية:
  - **Age**: أعمار عشوائية بين 18 و 65.
  - **Salary**: رواتب عشوائية بين 3000 و 15000.
  - **Experience**: عدد سنوات الخبرة بين 1 و 30.
  - **Department**: قسم عشوائي من الأقسام الأربعة.
- **pd.DataFrame(data)**: تحويل القاموس إلى إطار بيانات (جدول منظم).

---

### 3. إضافة الضوضاء إلى البيانات

#### أ. إضافة القيم الشاذة

```
df.loc[np.random.choice(df.index, 50), "Salary"] *= 10
```

- `np.random.choice(df.index, 50)`: اختيار 50 صفًا عشوائيًا.
- `df["Salary"] *= 10`: ضرب القيم في عمود "Salary" المختارة بعامل 10 لإضافة قيم شاذة.

#### ب. إضافة القيم المفقودة

```
df.loc[np.random.choice(df.index, 100), "Experience"] = np.nan
```

- `np.random.choice(df.index, 100)`: اختيار 100 صف عشوائي.
- `df["Experience"] = np.nan`: تعيين قيمة "خبرة" مفقودة (NaN) لهذه الصفوف.

---

### 4. إنشاء الرسوم البيانية

#### أ. الرسم الأول: توزيع الرواتب حسب الأقسام

```
plt.figure(figsize=(12, 6))
```

```
sns.boxplot(x="Department", y="Salary", data=df)
```

```
plt.title("Salary Distribution by Department (with Noise)")
```

```
plt.show(block=False)
```

- `plt.figure(figsize=(12, 6))`: إنشاء نافذة رسم بحجم 12x6 بوصات.
- `sns.boxplot(x="Department", y="Salary", data=df)`: رسم صندوق يوضح توزيع الرواتب لكل قسم.
- `plt.title()`: إضافة عنوان للرسم.
- `plt.show(block=False)`: عرض الرسم في نافذة منفصلة دون تعطيل البرنامج.

#### ب. الرسم الثاني: توزيع الأعمار

```
plt.figure(figsize=(8, 6))
```

```
sns.histplot(df["Age"], kde=True, bins=30)
```

```
plt.title("Age Distribution")
```

```
plt.show(block=False)
```

- `sns.histplot(df["Age"], kde=True, bins=30)`:
  - رسم مخطط أعمدة يوضح توزيع الأعمار.
  - `kde=True`: إضافة منحنى كثافة احتمالية فوق المخطط.
  - `bins=30`: تقسيم القيم إلى 30 حاوية.

- `plt.title()`: إضافة عنوان للرسم.
  - `plt.show(block=False)`: عرض الرسم في نافذة منفصلة.
- 

## 5. إبقاء الرسوم مرئية

`plt.show()`

- هذا السطر يضمن بقاء جميع الرسوم البيانية مفتوحة حتى يتم إغلاقها يدويًا.
- 

## ملخص الكود

- يقوم بإنشاء بيانات عشوائية تمثل سجلات موظفين.
- يضيف القيم الشاذة والمفقودة للبيانات لتكون أقرب إلى الواقع.
- يعرض رسومًا بيانية لتحليل البيانات :
  - توزيع الرواتب حسب الأقسام.
  - توزيع الأعمار.
- يسمح بإظهار جميع الرسوم في نفس الوقت دون تعطيل البرنامج.