

## Employee Data Analysis using Pandas, Matplotlib, and Seaborn

- This project demonstrates data analysis and visualization using the Pandas, Matplotlib, and Seaborn libraries in Python.
  - The dataset (data.csv) contains employee information such as Name, Department, Experience, and Salary.
  - The project performs data cleaning, computes key statistics, and generates various visualizations.
- 

### Technologies Used

- **Python 3.x**
  - **Pandas** → for data handling and analysis
  - **Matplotlib** → for plotting charts
  - **Seaborn** → for advanced visualizations
- 

### Dataset Structure (`data.csv`)

Name	Department	Experience	Salary
Asha	HR	3	30000
Ravi	IT	5	50000
Neha	Finance	2	35000

---

### Project Workflow

#### Step 1: Load the CSV File

```
data = pd.read_csv("data.csv")
```

---

#### Step 2: Data Cleaning

```
data['Salary'] = pd.to_numeric(data['Salary'], errors='coerce')
data['Experience'] = pd.to_numeric(data['Experience'], errors='coerce')
data = data.dropna(subset=['Salary', 'Experience'])
```

- Converts string data (like "Asha") to numeric safely.
- Removes invalid or missing numeric values.

#### Step 3: Display First Few Records

```
print(data.head())
```

- Shows the first few rows to verify data loading.
- 

#### Step:4 Calculate Average Salary

```
average_salary = data['Salary'].mean()
```

```
print("Average Salary:", average_salary)
```

- Finds the overall mean salary of all employees.
- 

#### Step 5:Department-wise Average Salary

```
avg_salary_dept = data.groupby('Department')['Salary'].mean()
```

```
print(avg_salary_dept)
```

- Calculates the average salary for each department.
- 

#### Step 6: Visualizations

##### Bar Chart – Department-wise Average Salary

```
avg_salary_dept.plot(kind='bar', color='skyblue')
```

```
plt.title("Department-wise Average Salary")
```

- Shows salary comparison across departments.
- 

##### Scatter Plot – Experience vs Salary

```
plt.scatter(data['Experience'], data['Salary'], color='green')
```

```
plt.title("Experience vs Salary")
```

- Displays the relationship between experience and salary.
- 

##### Heatmap – Correlation

```
corr = data.select_dtypes(include='number').corr()
```

```
sns.heatmap(corr, annot=True, cmap='coolwarm')
```

```
plt.title("Correlation Heatmap")
```

- Shows how different numeric columns are related.

# Outputs

- ✓ Bar Chart – Department Salary Comparison
  - ✓ Scatter Plot – Experience vs Salary
  - ✓ Heatmap – Correlation Matrix
- 

## How to Run

1. Save data.csv and the Python file (e.g., Program.py) in the same folder.
  2. Open VS Code or any IDE.
  3. Run:
  4. python Program.py
  5. Visualizations will appear one by one, close, then the next visualization will appear.
- 

```
6. import pandas as pd
7. import matplotlib.pyplot as plt
8. import seaborn as sns
9.
10.# [1] Load CSV file
11.data = pd.read_csv("data.csv")
12.
13.# [2] Show first 2-3 lines of data
14.print("Data:\n", data.head())
15.
16.# [3] Convert Salary & Experience columns to numeric (ignore errors like
     'Asha')
17.data['Salary'] = pd.to_numeric(data['Salary'], errors='coerce')
18.data['Experience'] = pd.to_numeric(data['Experience'], errors='coerce')
19.
20.# [4] Drop rows with missing values in key columns
21.data = data.dropna(subset=['Salary', 'Experience'])
22.
23.# [5] Average Salary
24.average_salary = data['Salary'].mean()
25.print("\nAverage Salary:", average_salary)
26.
27.# [6] Department-wise Average Salary
28.avg_salary_dept = data.groupby('Department')['Salary'].mean()
29.print("\nDepartment-wise Average Salary:\n", avg_salary_dept)
30.
31.# [7] Bar Chart
32.avg_salary_dept.plot(kind='bar', color='skyblue')
33.plt.title("Department-wise Average Salary")
```

```

34.plt.xlabel("Department")
35.plt.ylabel("Average Salary")
36.plt.show()
37.
38.# ❸ Scatter Plot (Experience vs Salary)
39.plt.scatter(data['Experience'], data['Salary'], color='green')
40.plt.title("Experience vs Salary")
41.plt.xlabel("Experience (Years)")
42.plt.ylabel("Salary")
43.plt.show()
44.
45.# ❹ Heatmap (Correlation)
46.# Select only numeric columns for correlation
47.corr = data.select_dtypes(include='number').corr()
48.sns.heatmap(corr, annot=True, cmap='coolwarm')
49.plt.title("Correlation Heatmap")
50.plt.show()
51.

```

```

PS D:\Priti_Data\Python_Internship\Project1(visualization)> python -u "d:\Priti_Data\Python_Internship\Project1(visualization)\program.py"
Data:
   Name Department  Experience   Salary
0  Asha        IT          2  35000
1  Ravi        HR          5  40000
2  Sneha       IT          7  65000
3  Kiran    Finance          4  45000
4  Meena       HR          8  55000

Average Salary: 52100.0

Department-wise Average Salary:
Department
Finance    53666.666667
HR         47666.666667
IT         54250.000000
Name: Salary, dtype: float64

```



