Python Lab Exercises - NumPy, Matplotlib, Pandas

NumPy Lab

```
1. Create 1D, 2D, and 3D arrays using NumPy.
-----------------------------------------------
import numpy as np
a1 = np.array([1, 2, 3])
a2 = np.array([[1, 2], [3, 4]])
a3 = np.array([[[1, 2], [3, 4]], [[5, 6], [7, 8]]])
print(a1); print(a2); print(a3)

2. Create a 5×5 identity matrix using np.eye().
-----------------------------------------------
I = np.eye(5)
print(I)

3. Generate random numbers.
-----------------------------------------------
rand_ints = np.random.randint(1, 101, 10)
rand_floats = np.random.rand(3, 3)
print(rand_ints); print(rand_floats)

4. Perform arithmetic on arrays.
-----------------------------------------------
a = np.array([10, 20, 30])
b = np.array([1, 2, 3])
print(a+b, a-b, a*b, a/b)

5. Broadcasting demonstration.
-----------------------------------------------
arr2D = np.array([[1,2,3],[4,5,6],[7,8,9]])
arr1D = np.array([10,20,30])
print(arr2D + arr1D)

6. Compute statistical measures.
-----------------------------------------------
scores = np.array([50,60,70,80,90])
print("Mean:", scores.mean())
print("Median:", np.median(scores))
print("Std:", scores.std())
print("Var:", scores.var())

7. Slice 2D array to get 2x2 center submatrix.
-----------------------------------------------
arr = np.arange(16).reshape(4,4)
print(arr[1:3,1:3])

8. Reshape 1D array into 2D forms.
-----------------------------------------------
arr = np.arange(12)
print(arr.reshape(3,4))
print(arr.reshape(2,6))

9. Negative and boolean indexing.
-----------------------------------------------
arr = np.array([1,2,3,4,5])
print(arr[-1])
print(arr[arr>2])

10. Flatten arrays with flatten() and ravel().
-----------------------------------------------
arr = np.array([[1,2],[3,4]])
print(arr.flatten())
print(arr.ravel())
```

Matplotlib Lab

```
1. Plot a line graph for daily temperatures.
-----------------------------------------------
import matplotlib.pyplot as plt
days = ["Mon","Tue","Wed","Thu","Fri","Sat","Sun"]
temps = [30,32,31,29,35,36,34]
plt.plot(days, temps, marker="o")
```

```
plt.show()

2. Bar chart for sales over 5 months.
-------------------------------------------------
months = ["Jan","Feb","Mar","Apr","May"]
sales = [100,150,120,180,200]
plt.bar(months, sales)
plt.show()

3. Multi-line graph for stock prices.
-------------------------------------------------
days = [1,2,3,4,5,6,7]
A = [100,102,101,105,107,106,108]
B = [90,91,93,92,95,94,96]
C = [120,119,121,122,124,123,125]
plt.plot(days,A,label="Company A")
plt.plot(days,B,label="Company B")
plt.plot(days,C,label="Company C")
plt.legend(); plt.show()

4. Horizontal bar chart for population by city.
-------------------------------------------------
cities = ["City1","City2","City3"]
pop = [1.2,2.5,3.0]
plt.barh(cities,pop)
plt.show()

5. Pie chart of smartphone market share.
-------------------------------------------------
brands = ["A","B","C","D","E"]
share = [30,25,20,15,10]
plt.pie(share,labels=brands,autopct="%1.1f%%")
plt.show()

6. Histogram of student scores.
-------------------------------------------------
scores = [45,55,65,75,85,95,55,65,75]
plt.hist(scores, bins=5)
plt.show()

7. Scatter plot for height vs weight.
-------------------------------------------------
height = [150,160,170,180]
weight = [50,60,70,80]
plt.scatter(height, weight)
plt.show()

8. Bubble chart with variable size.
-------------------------------------------------
x=[5,7,8,7]; y=[99,86,87,88]; sizes=[50,100,200,300]
plt.scatter(x,y,s=sizes,alpha=0.5)
plt.show()
```

Pandas Lab

```
1. Load dataset and display first 5 rows.
-------------------------------------------------
import pandas as pd
df = pd.read_csv("employees.csv")
print(df.head())

2. Column names and data types.
-------------------------------------------------
print(df.columns)
print(df.dtypes)

3. Summary statistics.
-------------------------------------------------
print(df.describe())

4. Salary > 50000.
-------------------------------------------------
print(df[df["Salary"] > 50000])

5. Employees in HR.
-------------------------------------------------
print(df[df["Department"]=="HR"]["Name"])
```

```
6. Add Tax column (10% Salary).
-------------------------------------------------
df["Tax"] = df["Salary"]*0.1

7. Group by Department.
-------------------------------------------------
print(df.groupby("Department")["Salary"].mean())

8. Sort by Age ascending.
-------------------------------------------------
print(df.sort_values(by="Age"))

9. Drop missing values.
-------------------------------------------------
df_clean = df.dropna()

10. Save cleaned dataset.
-------------------------------------------------
df_clean.to_csv("employees_cleaned.csv", index=False)

11. Bar chart for salary of employees.
-------------------------------------------------
import matplotlib.pyplot as plt
plt.bar(df["Name"], df["Salary"])
plt.show()

12. Pie chart for employees per department.
-------------------------------------------------
dept_counts = df["Department"].value_counts()
plt.pie(dept_counts, labels=dept_counts.index, autopct="%1.1f%%")
plt.show()
```