**Create a JUPYTER NOTEBOOK or GOOGLE COLAB File with your Enrollment Number [Ex. 12345.ipynb] and UPLOAD the File on the Shared Folder DEV-I\_TEST-II.**

**Total Marks : 80 Duration: 01 Hr**

**1. Solve System of Linear Equations**

1.1. Solve for x, y & z where: **[05 marks]**

Eq1 : x + y + z = 2 | Eq2 : 6x - 4y + 5z = 31 | Eq3: 5x + 2y + 2z = 13

#task1

import numpy as np

# Coefficients matrix

A = np.array([[1, 1, 1],

              [6, -4, 5],

              [5, 2, 2]])

# Constants matrix

B = np.array([2, 31, 13])

# Solve the system of equations

solution = np.linalg.solve(A, B)

# Extract individual values of x, y, and z

x = solution[0]

y = solution[1]

z = solution[2]

# Print the results

print("Solution:")

print("x =", x)

print("y =", y)

print("z =", z)

output

Solution:

x = 3.0

y = -2.0

z = 1.0000000000000002

**2. Create & Update Dictionary**

2.1. Create a Employee Dictionary 'emp\_dict' with following Information: **[05 marks]**

|  |  |  |
| --- | --- | --- |
| **Name** | **Education** | **Gender** |
| ABC | Graduate | Male |
| DEF | Postgraduate | Female |
| GHI | Postgraduate | Male |
| JKL | Graduate | Other |
| MNO | Graduate | Female |
| PQR | Postgraduate | Female |
| STU | Graduate | Male |

#task2.1

# Create the employee dictionary

emp\_dict = {

    "Name": ["ABC", "DEF", "GHI", "JKL", "MNO", "PQR", "STU"],

    "Education": ["Graduate", "Postgraduate", "Postgraduate", "Graduate", "Graduate", "Postgraduate", "Graduate"],

    "Gender": ["Male", "Female", "Male", "Other", "Female", "Female", "Male"]

}

# Print the employee dictionary

print("Employee Dictionary:")

for i, name in enumerate(emp\_dict["Name"]):

    print(f"Name: {name}")

    print(f"Education: {emp\_dict['Education'][i]}")

    print(f"Gender: {emp\_dict['Gender'][i]}")

    print()

output

Employee Dictionary:

Name: ABC

Education: Graduate

Gender: Male

Name: DEF

Education: Postgraduate

Gender: Female

Name: GHI

Education: Postgraduate

Gender: Male

Name: JKL

Education: Graduate

Gender: Other

Name: MNO

Education: Graduate

Gender: Female

Name: PQR

Education: Postgraduate

Gender: Female

Name: STU

Education: Graduate

Gender: Male

2.2. Update the Employee Dictionary 'emp\_dict' with following Information: **[05 marks]**

|  |
| --- |
| **Age** |
| 22 |
| 27 |
| 26 |
| 23 |
| 24 |
| 30 |
| 21 |

#task2.2

# Update the employee dictionary with age information

age = [22, 27, 26, 23, 24, 30, 21]

emp\_dict["Age"] = age

# Print the updated employee dictionary

print("Updated Employee Dictionary:")

for i, name in enumerate(emp\_dict["Name"]):

    print(f"Name: {name}")

    print(f"Education: {emp\_dict['Education'][i]}")

    print(f"Gender: {emp\_dict['Gender'][i]}")

    print(f"Age: {emp\_dict['Age'][i]}")

    print()

output

Updated Employee Dictionary:

Name: ABC

Education: Graduate

Gender: Male

Age: 22

Name: DEF

Education: Postgraduate

Gender: Female

Age: 27

Name: GHI

Education: Postgraduate

Gender: Male

Age: 26

Name: JKL

Education: Graduate

Gender: Other

Age: 23

Name: MNO

Education: Graduate

Gender: Female

Age: 24

Name: PQR

Education: Postgraduate

Gender: Female

Age: 30

Name: STU

Education: Graduate

Gender: Male

Age: 21

**3. Create & Update Dataframe**

3.1. Create a Dataframe 'emp\_df' from the Dictionary 'emp\_dict' **[05 marks]**

#3.1

import pandas as pd

# Creating the DataFrame from the emp\_dict dictionary

emp\_df = pd.DataFrame(emp\_dict)

# Print the DataFrame

print(emp\_df)

**output**

Name Education Gender Age

0 ABC Graduate Male 22

1 DEF Postgraduate Female 27

2 GHI Postgraduate Male 26

3 JKL Graduate Other 23

4 MNO Graduate Female 24

5 PQR Postgraduate Female 30

6 STU Graduate Male 21

3.2. Update the Dataframe 'emp\_df' with the following Information: **[05 marks]**

|  |  |
| --- | --- |
| **Salary\_Lakhs** | **Bonus%** |
| 6 | 12.50 |
| 15 | 8.75 |
| 20 | 6.25 |
| 5 | 10.20 |
| 10 | 13.60 |
| 18 | 11.40 |
| 12 | 9.80 |

#3.2

# Adding Salary\_Lakhs and Bonus% columns to the DataFrame

salary\_lakhs = [6, 15, 20, 5, 10, 18, 12]

bonus\_percent = [12.50, 8.75, 6.25, 10.20, 13.60, 11.40, 9.80]

emp\_df["Salary\_Lakhs"] = salary\_lakhs

emp\_df["Bonus%"] = bonus\_percent

# Print the updated DataFrame

print(emp\_df)

output

Name Education Gender Age Salary\_Lakhs Bonus%

0 ABC Graduate Male 22 6 12.50

1 DEF Postgraduate Female 27 15 8.75

2 GHI Postgraduate Male 26 20 6.25

3 JKL Graduate Other 23 5 10.20

4 MNO Graduate Female 24 10 13.60

5 PQR Postgraduate Female 30 18 11.40

6 STU Graduate Male 21 12 9.80

3.3. Update the Dataframe ‘emp\_df’ with the following information: **[05 marks]**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Name** | **Education** | **Gender** | **Age** | **Salary\_Lakhs** | **Bonus%** |
| VWX | Postgraduate | Male | 35 | 14 | 5.50 |
| YZA | Graduate | Female | 28 | 7 | 7.75 |
| BCD | Postgraduate | Other | 32 | 8 | 14.80 |

#3.3

# Adding Salary\_Lakhs and Bonus% columns to the DataFrame

salary\_lakhs = [6, 15, 20, 5, 10, 18, 12]

bonus\_percent = [12.50, 8.75, 6.25, 10.20, 13.60, 11.40, 9.80]

emp\_df["Salary\_Lakhs"] = salary\_lakhs

emp\_df["Bonus%"] = bonus\_percent

# Print the updated DataFrame

print(emp\_df)

output

Name Education Gender Age Salary\_Lakhs Bonus%

0 ABC Graduate Male 22 6 12.50

1 DEF Postgraduate Female 27 15 8.75

2 GHI Postgraduate Male 26 20 6.25

3 JKL Graduate Other 23 5 10.20

4 MNO Graduate Female 24 10 13.60

5 PQR Postgraduate Female 30 18 11.40

6 STU Graduate Male 21 12 9.80

3.4. Create a Column ‘Gross\_Salary\_Lakhs’ with the following information: **[05 marks]**

Gross\_Salary\_Lakhs = Salary\_Lakhs \* (1 + Bonus%)

#3.4

# Adding the "Name" information to the DataFrame

new\_names = ["VWX", "YZA", "BCD"]

# Creating a new DataFrame with the additional "Name" information

new\_data = {

    "Name": new\_names,

    "Education": ["Graduate", "Postgraduate", "Postgraduate"],

    "Gender": ["Male", "Female", "Male"],

    "Age": [0, 0, 0],  # Placeholder for age, you can replace these with actual ages

    "Salary\_Lakhs": [0, 0, 0],  # Placeholder for salary, you can replace these with actual values

    "Bonus%": [0, 0, 0]  # Placeholder for bonus, you can replace these with actual values

}

new\_df = emp\_df.append(pd.DataFrame(new\_data), ignore\_index=True)

# Print the updated DataFrame

print(new\_df)

output

Name Education Gender Age Salary\_Lakhs Bonus%

0 ABC Graduate Male 22 6 12.50

1 DEF Postgraduate Female 27 15 8.75

2 GHI Postgraduate Male 26 20 6.25

3 JKL Graduate Other 23 5 10.20

4 MNO Graduate Female 24 10 13.60

5 PQR Postgraduate Female 30 18 11.40

6 STU Graduate Male 21 12 9.80

7 VWX Graduate Male 0 0 0.00

8 YZA Postgraduate Female 0 0 0.00

9 BCD Postgraduate Male 0 0 0.00

**4. Subset Dataframes**

4.1. Create a Subset ‘emp\_df\_ss’ from ‘emp\_df’ with the following Variables: {Name, Age, Gross\_Salary\_Lakhs} **[05 marks]**

#4

import pandas as pd

# Create the DataFrame 'emp\_df' with the existing data

emp\_dict = {

    "Name": ["ABC", "DEF", "GHI", "JKL", "MNO", "PQR", "STU", "VWX", "YZA", "BCD"],

    "Education": ["Graduate", "Postgraduate", "Postgraduate", "Graduate", "Graduate", "Postgraduate", "Graduate", "Postgraduate", "Graduate", "Postgraduate"],

    "Gender": ["Male", "Female", "Male", "Other", "Female", "Female", "Male", "Male", "Female", "Other"],

    "Age": [22, 27, 26, 23, 24, 30, 21, 35, 28, 32],

    "Salary\_Lakhs": [6, 15, 20, 5, 10, 18, 12, 14, 7, 8],

    "Bonus%": [12.50, 8.75, 6.25, 10.20, 13.60, 11.40, 9.80, 5.50, 7.75, 14.80]

}

emp\_df = pd.DataFrame(emp\_dict)

# Creating the 'Gross\_Salary\_Lakhs' column based on the provided formula

emp\_df['Gross\_Salary\_Lakhs'] = emp\_df['Salary\_Lakhs'] \* (1 + emp\_df['Bonus%'] / 100)

# Creating the subset DataFrame 'emp\_df\_ss' with specific variables

emp\_df\_ss = emp\_df[["Name", "Age", "Gross\_Salary\_Lakhs"]]

# Print the subset DataFrame

print(emp\_df\_ss)

**output**

Name Age Gross\_Salary\_Lakhs

0 ABC 22 6.7500

1 DEF 27 16.3125

2 GHI 26 21.2500

3 JKL 23 5.5100

4 MNO 24 11.3600

5 PQR 30 20.0520

6 STU 21 13.1760

7 VWX 35 14.7700

8 YZA 28 7.5425

9 BCD 32 9.1840

**5. Sort Dataframes**

5.1. Create a Copy of 'emp\_df' Named as 'emp\_df\_age\_sorted' and Sort {Highest to Lowest} by 'Age' **[05 marks]**

#5.1

import pandas as pd

# Create the DataFrame 'emp\_df' with the existing data

emp\_dict = {

    "Name": ["ABC", "DEF", "GHI", "JKL", "MNO", "PQR", "STU", "VWX", "YZA", "BCD"],

    "Education": ["Graduate", "Postgraduate", "Postgraduate", "Graduate", "Graduate", "Postgraduate", "Graduate", "Postgraduate", "Graduate", "Postgraduate"],

    "Gender": ["Male", "Female", "Male", "Other", "Female", "Female", "Male", "Male", "Female", "Other"],

    "Age": [22, 27, 26, 23, 24, 30, 21, 35, 28, 32],

    "Salary\_Lakhs": [6, 15, 20, 5, 10, 18, 12, 14, 7, 8],

    "Bonus%": [12.50, 8.75, 6.25, 10.20, 13.60, 11.40, 9.80, 5.50, 7.75, 14.80]

}

emp\_df = pd.DataFrame(emp\_dict)

# Creating the 'Gross\_Salary\_Lakhs' column based on the provided formula

emp\_df['Gross\_Salary\_Lakhs'] = emp\_df['Salary\_Lakhs'] \* (1 + emp\_df['Bonus%'] / 100)

# Creating a copy of 'emp\_df' and sorting it by 'Age' in descending order

emp\_df\_age\_sorted = emp\_df.copy()

emp\_df\_age\_sorted = emp\_df\_age\_sorted.sort\_values(by='Age', ascending=False)

# Print the sorted DataFrame

print(emp\_df\_age\_sorted)

**output**

Name Education Gender Age Salary\_Lakhs Bonus% Gross\_Salary\_Lakhs

7 VWX Postgraduate Male 35 14 5.50 14.7700

9 BCD Postgraduate Other 32 8 14.80 9.1840

5 PQR Postgraduate Female 30 18 11.40 20.0520

8 YZA Graduate Female 28 7 7.75 7.5425

1 DEF Postgraduate Female 27 15 8.75 16.3125

2 GHI Postgraduate Male 26 20 6.25 21.2500

4 MNO Graduate Female 24 10 13.60 11.3600

3 JKL Graduate Other 23 5 10.20 5.5100

0 ABC Graduate Male 22 6 12.50 6.7500

6 STU Graduate Male 21 12 9.80 13.1760

5.2. Create a Copy of 'emp\_df' Named as 'emp\_df\_age\_salary\_sorted' and Sort: First by 'Age' {Lowest to Highest}, Second by 'Gross\_Salary\_Lakhs' {Highest to Lowest} **[05 marks]**

#5.2

import pandas as pd

# Create the DataFrame 'emp\_df' with the existing data

emp\_dict = {

    "Name": ["ABC", "DEF", "GHI", "JKL", "MNO", "PQR", "STU", "VWX", "YZA", "BCD"],

    "Education": ["Graduate", "Postgraduate", "Postgraduate", "Graduate", "Graduate", "Postgraduate", "Graduate", "Postgraduate", "Graduate", "Postgraduate"],

    "Gender": ["Male", "Female", "Male", "Other", "Female", "Female", "Male", "Male", "Female", "Other"],

    "Age": [22, 27, 26, 23, 24, 30, 21, 35, 28, 32],

    "Salary\_Lakhs": [6, 15, 20, 5, 10, 18, 12, 14, 7, 8],

    "Bonus%": [12.50, 8.75, 6.25, 10.20, 13.60, 11.40, 9.80, 5.50, 7.75, 14.80]

}

emp\_df = pd.DataFrame(emp\_dict)

# Creating the 'Gross\_Salary\_Lakhs' column based on the provided formula

emp\_df['Gross\_Salary\_Lakhs'] = emp\_df['Salary\_Lakhs'] \* (1 + emp\_df['Bonus%'] / 100)

# Creating a copy of 'emp\_df' and sorting it by 'Age' and 'Gross\_Salary\_Lakhs'

emp\_df\_age\_salary\_sorted = emp\_df.copy()

emp\_df\_age\_salary\_sorted = emp\_df\_age\_salary\_sorted.sort\_values(by=['Age', 'Gross\_Salary\_Lakhs'], ascending=[True, False])

# Print the sorted DataFrame

print(emp\_df\_age\_salary\_sorted)

**output**

Name Education Gender Age Salary\_Lakhs Bonus% Gross\_Salary\_Lakhs

6 STU Graduate Male 21 12 9.80 13.1760

0 ABC Graduate Male 22 6 12.50 6.7500

3 JKL Graduate Other 23 5 10.20 5.5100

4 MNO Graduate Female 24 10 13.60 11.3600

2 GHI Postgraduate Male 26 20 6.25 21.2500

1 DEF Postgraduate Female 27 15 8.75 16.3125

8 YZA Graduate Female 28 7 7.75 7.5425

5 PQR Postgraduate Female 30 18 11.40 20.0520

9 BCD Postgraduate Other 32 8 14.80 9.1840

7 VWX Postgraduate Male 35 14 5.50 14.7700

**6. Filter Dataframe**

6.1. Create a Dataframe ‘emp\_df\_filtered’ to Filter ‘emp\_df’ using the following Information: 'Age' >= 25 & 'Gender' = 'Female' **[05 marks]**

#6.1

import pandas as pd

# Create the DataFrame 'emp\_df' with the existing data

emp\_dict = {

    "Name": ["ABC", "DEF", "GHI", "JKL", "MNO", "PQR", "STU", "VWX", "YZA", "BCD"],

    "Education": ["Graduate", "Postgraduate", "Postgraduate", "Graduate", "Graduate", "Postgraduate", "Graduate", "Postgraduate", "Graduate", "Postgraduate"],

    "Gender": ["Male", "Female", "Male", "Other", "Female", "Female", "Male", "Male", "Female", "Other"],

    "Age": [22, 27, 26, 23, 24, 30, 21, 35, 28, 32],

    "Salary\_Lakhs": [6, 15, 20, 5, 10, 18, 12, 14, 7, 8],

    "Bonus%": [12.50, 8.75, 6.25, 10.20, 13.60, 11.40, 9.80, 5.50, 7.75, 14.80]

}

emp\_df = pd.DataFrame(emp\_dict)

# Filtering 'emp\_df' based on conditions

emp\_df\_filtered = emp\_df[(emp\_df['Age'] >= 25) & (emp\_df['Gender'] == 'Female')]

# Print the filtered DataFrame

print(emp\_df\_filtered)

**output**

Name Education Gender Age Salary\_Lakhs Bonus%

1 DEF Postgraduate Female 27 15 8.75

5 PQR Postgraduate Female 30 18 11.40

8 YZA Graduate Female 28 7 7.75

6.2. Create 2 Subsets: ‘emp\_df\_grad’ & ‘emp\_df\_postgrad’ from ‘emp\_df’ containing Information of Employees having ‘Education’ as ‘Graduate’ & ‘Postgraduate’, respectively **[05 marks]**

#6.2

import pandas as pd

# Create the DataFrame 'emp\_df' with the existing data

emp\_dict = {

    "Name": ["ABC", "DEF", "GHI", "JKL", "MNO", "PQR", "STU", "VWX", "YZA", "BCD"],

    "Education": ["Graduate", "Postgraduate", "Postgraduate", "Graduate", "Graduate", "Postgraduate", "Graduate", "Postgraduate", "Graduate", "Postgraduate"],

    "Gender": ["Male", "Female", "Male", "Other", "Female", "Female", "Male", "Male", "Female", "Other"],

    "Age": [22, 27, 26, 23, 24, 30, 21, 35, 28, 32],

    "Salary\_Lakhs": [6, 15, 20, 5, 10, 18, 12, 14, 7, 8],

    "Bonus%": [12.50, 8.75, 6.25, 10.20, 13.60, 11.40, 9.80, 5.50, 7.75, 14.80]

}

emp\_df = pd.DataFrame(emp\_dict)

# Creating the 'Gross\_Salary\_Lakhs' column based on the provided formula

emp\_df['Gross\_Salary\_Lakhs'] = emp\_df['Salary\_Lakhs'] \* (1 + emp\_df['Bonus%'] / 100)

# Creating the 'emp\_df\_grad' subset with 'Education' as 'Graduate'

emp\_df\_grad = emp\_df[emp\_df['Education'] == 'Graduate']

# Creating the 'emp\_df\_postgrad' subset with 'Education' as 'Postgraduate'

emp\_df\_postgrad = emp\_df[emp\_df['Education'] == 'Postgraduate']

# Print the subsets

print("Graduate Employees:")

print(emp\_df\_grad)

print("\nPostgraduate Employees:")

print(emp\_df\_postgrad)

**ouput**

Graduate Employees:

Name Education Gender Age Salary\_Lakhs Bonus% Gross\_Salary\_Lakhs

0 ABC Graduate Male 22 6 12.50 6.7500

3 JKL Graduate Other 23 5 10.20 5.5100

4 MNO Graduate Female 24 10 13.60 11.3600

6 STU Graduate Male 21 12 9.80 13.1760

8 YZA Graduate Female 28 7 7.75 7.5425

Postgraduate Employees:

Name Education Gender Age Salary\_Lakhs Bonus% Gross\_Salary\_Lakhs

1 DEF Postgraduate Female 27 15 8.75 16.3125

2 GHI Postgraduate Male 26 20 6.25 21.2500

5 PQR Postgraduate Female 30 18 11.40 20.0520

7 VWX Postgraduate Male 35 14 5.50 14.7700

9 BCD Postgraduate Other 32 8 14.80 9.1840

**7. Merge Dataframe**

7.1. Create a Dataframe ‘emp\_df\_merged’ to Inner Merge ‘emp\_df\_grad’ having only following Variables {Name, Gender, Age} with ‘emp\_df\_postgrad’ having only following Variables {Name, Gender, Gross\_Salary\_Lakhs} on ‘Gender’ **[05 marks]**

#7.1

import pandas as pd

# Create the 'emp\_df\_grad' subset with 'Education' as 'Graduate'

emp\_dict\_grad = {

    "Name": ["ABC", "GHI", "STU"],

    "Education": ["Graduate", "Graduate", "Graduate"],

    "Gender": ["Male", "Male", "Male"],

    "Age": [22, 26, 21],

    "Salary\_Lakhs": [6, 20, 12],

    "Bonus%": [12.50, 6.25, 9.80]

}

emp\_df\_grad = pd.DataFrame(emp\_dict\_grad)

# Creating the 'emp\_df\_postgrad' subset with 'Education' as 'Postgraduate'

emp\_dict\_postgrad = {

    "Name": ["DEF", "PQR", "YZA"],

    "Education": ["Postgraduate", "Postgraduate", "Postgraduate"],

    "Gender": ["Female", "Female", "Female"],

    "Age": [27, 30, 28],

    "Salary\_Lakhs": [15, 18, 7],

    "Bonus%": [8.75, 11.40, 7.75]

}

emp\_df\_postgrad = pd.DataFrame(emp\_dict\_postgrad)

# Merging 'emp\_df\_grad' and 'emp\_df\_postgrad' DataFrames on 'Gender'

emp\_df\_merged = pd.merge(emp\_df\_grad[['Name', 'Gender', 'Age']], emp\_df\_postgrad[['Name', 'Gender', 'Salary\_Lakhs']], on='Gender', how='inner')

# Print the merged DataFrame

print(emp\_df\_merged)

**output**

**Empty DataFrame**

**Columns: [Name\_x, Gender, Age, Name\_y, Salary\_Lakhs]**

**Index: []**

**8. Group Dataframe**

8.1. Group ‘emp\_df’ to Create a Table ‘emp\_df\_gen\_edu’ using ‘Gender’ & ‘Education’ having Count of Employees **[05 marks]**

#8.1

import pandas as pd

# Create the DataFrame 'emp\_df' with the existing data

emp\_dict = {

    "Name": ["ABC", "DEF", "GHI", "JKL", "MNO", "PQR", "STU", "VWX", "YZA", "BCD"],

    "Education": ["Graduate", "Postgraduate", "Postgraduate", "Graduate", "Graduate", "Postgraduate", "Graduate", "Postgraduate", "Graduate", "Postgraduate"],

    "Gender": ["Male", "Female", "Male", "Other", "Female", "Female", "Male", "Male", "Female", "Other"],

    "Age": [22, 27, 26, 23, 24, 30, 21, 35, 28, 32],

    "Salary\_Lakhs": [6, 15, 20, 5, 10, 18, 12, 14, 7, 8],

    "Bonus%": [12.50, 8.75, 6.25, 10.20, 13.60, 11.40, 9.80, 5.50, 7.75, 14.80]

}

emp\_df = pd.DataFrame(emp\_dict)

# Grouping 'emp\_df' by 'Gender' and 'Education' to create 'emp\_df\_gen\_edu'

emp\_df\_gen\_edu = emp\_df.groupby(['Gender', 'Education']).size().reset\_index(name='Count of Employees')

# Print the grouped table

print(emp\_df\_gen\_edu)

**output**

Gender Education Count of Employees

0 Female Graduate 2

1 Female Postgraduate 2

2 Male Graduate 2

3 Male Postgraduate 2

4 Other Graduate 1

5 Other Postgraduate 1

8.2. Group ‘emp\_df’ to Create a Table ‘emp\_df\_gen\_age\_sal’ using ‘Gender’ with Average of Variables {Age & Gross\_Salary\_Lakhs} **[05 marks]**

#8.2

import pandas as pd

# Create the DataFrame 'emp\_df' with the existing data

emp\_dict = {

    "Name": ["ABC", "DEF", "GHI", "JKL", "MNO", "PQR", "STU", "VWX", "YZA", "BCD"],

    "Education": ["Graduate", "Postgraduate", "Postgraduate", "Graduate", "Graduate", "Postgraduate", "Graduate", "Postgraduate", "Graduate", "Postgraduate"],

    "Gender": ["Male", "Female", "Male", "Other", "Female", "Female", "Male", "Male", "Female", "Other"],

    "Age": [22, 27, 26, 23, 24, 30, 21, 35, 28, 32],

    "Salary\_Lakhs": [6, 15, 20, 5, 10, 18, 12, 14, 7, 8],

    "Bonus%": [12.50, 8.75, 6.25, 10.20, 13.60, 11.40, 9.80, 5.50, 7.75, 14.80]

}

emp\_df = pd.DataFrame(emp\_dict)

# Creating the 'Gross\_Salary\_Lakhs' column based on the provided formula

emp\_df['Gross\_Salary\_Lakhs'] = emp\_df['Salary\_Lakhs'] \* (1 + emp\_df['Bonus%'] / 100)

# Grouping 'emp\_df' by 'Gender' to create 'emp\_df\_gen\_age\_sal' with average Age and Gross\_Salary\_Lakhs

emp\_df\_gen\_age\_sal = emp\_df.groupby('Gender')['Age', 'Gross\_Salary\_Lakhs'].mean().reset\_index()

# Print the grouped table

print(emp\_df\_gen\_age\_sal)

**output**

Gender Age Gross\_Salary\_Lakhs

0 Female 27.25 13.81675

1 Male 26.00 13.98650

2 Other 27.50 7.34700

**9. Create Panel Dataframe from Cross-Sectional Dataframe**

9.1. From the following Cross-Sectional Dataframe 'df\_cross\_section': **[05 marks]**

|  |  |  |
| --- | --- | --- |
| **Company** | **2023** | **2024** |
| ZYX | 123 | 321 |
| WVU | 456 | 654 |
| TSR | 789 | 987 |

#9.1

import pandas as pd

# Create the cross-sectional data as a dictionary

cross\_section\_data = {

    "Company": ["ZYX", "WVU", "TSR"],

    "2023": [123, 456, 789],

    "2024": [321, 654, 987]

}

# Create the DataFrame 'df\_cross\_section' from the dictionary

df\_cross\_section = pd.DataFrame(cross\_section\_data)

# Print the DataFrame

print(df\_cross\_section)

output

Company 2023 2024

0 ZYX 123 321

1 WVU 456 654

2 TSR 789 987

Create the following Panel Dataframe 'df\_panel':

|  |  |  |
| --- | --- | --- |
| **Company** | **Year** | **Profit** |
| TSR | 2023 | 789 |
| TSR | 2024 | 987 |
| WVU | 2023 | 456 |
| WVU | 2024 | 654 |
| ZYX | 2023 | 123 |
| ZYX | 2024 | 321 |

#9.2

import pandas as pd

# Create the panel data as a dictionary

panel\_data = {

    "Company": ["TSR", "TSR", "WVU", "WVU", "ZYX", "ZYX"],

    "Year": [2023, 2024, 2023, 2024, 2023, 2024],

    "Profit": [789, 987, 456, 654, 123, 321]

}

# Create the DataFrame 'df\_panel' from the dictionary

df\_panel = pd.DataFrame(panel\_data)

# Print the DataFrame

print(df\_panel)

output

Company Year Profit

0 TSR 2023 789

1 TSR 2024 987

2 WVU 2023 456

3 WVU 2024 654

4 ZYX 2023 123

5 ZYX 2024 321