Experiment 33

Aim

To Write a Python program to draw a scatter plot with empty circles taking a random distribution in X and Y and plotted against each other.

Code:

import numpy as np

import matplotlib.pyplot as plt

# Generate random data

x = np.random.rand(100) # 100 random values for X

y = np.random.rand(100) # 100 random values for Y

# Create a scatter plot with empty circles

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

plt.scatter(x, y, edgecolors='blue', facecolors='none', alpha=0.5)

plt.title('Scatter Plot with Empty Circles')

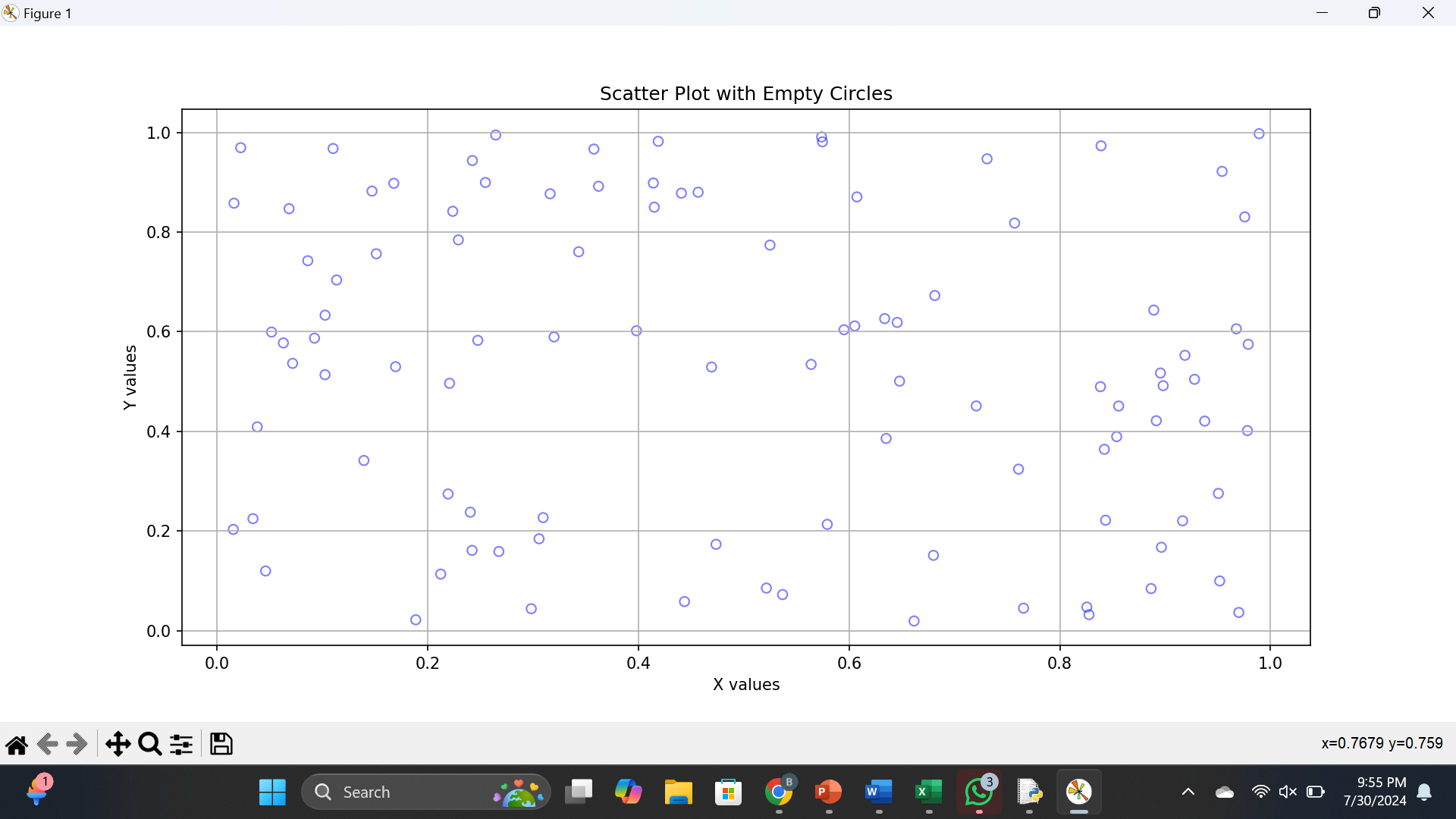
plt.xlabel('X values')

plt.ylabel('Y values')

plt.grid(True)

plt.show()

Output :



Results:

Thus a Python program to draw a scatter plot with empty circles taking a random distribution in X and Y and plotted against each other.

**Experiment 34**

**Aim:**

To Write a Python program to draw a scatter plot using random distributions to generate balls of different sizes.

**Code**

import matplotlib.pyplot as plt

import numpy as np

n = 30

x = np.random.rand(n)

y = np.random.rand(n)

colors = np.random.rand(n)

area = (30 \* np.random.rand(n))\*\*2 # 0 to 15 point radii

# Create the bubble plot

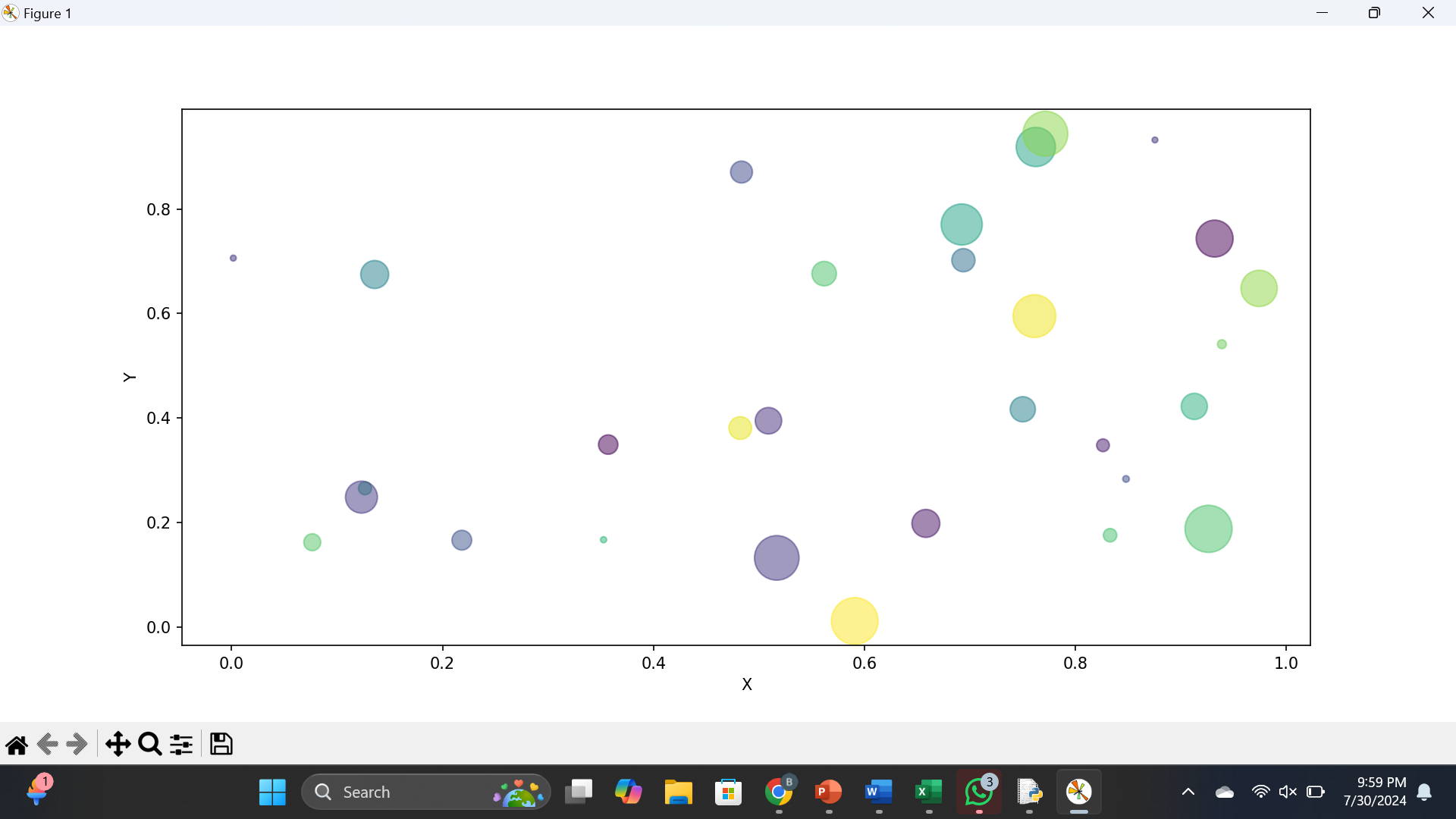
plt.scatter(x, y, s=area, c=colors, alpha=0.5)

plt.xlabel('X')

plt.ylabel('Y')

plt.show()

**Output:**

****

**Results:**

Thus a Python program to draw a scatter plot using random distributions to generate balls of different sizes.

Experiment 35

Aim:

To.Write a Python program to draw a scatter plot comparing two subject marks of Mathematics and Science. Use marks of 10 students.

Code:

import matplotlib.pyplot as plt

# Sample data

math\_marks = [88, 92, 80, 89, 100, 80, 60, 100, 80, 34]

science\_marks = [35, 79, 79, 48, 100, 88, 32, 45, 20, 30]

marks\_range = [10, 20, 30, 40, 50, 60, 70, 80, 90, 100]

# Create the scatter plot

plt.scatter(marks\_range, math\_marks, color='red', alpha=0.5, label='Math marks')

plt.scatter(marks\_range, science\_marks, color='green', alpha=0.5, label='Science marks')

# Add labels and title

plt.xlabel('Marks Range')

plt.ylabel('Marks Scored')

plt.title('Scatter Plot of Math vs. Science Marks')

# Add a legend

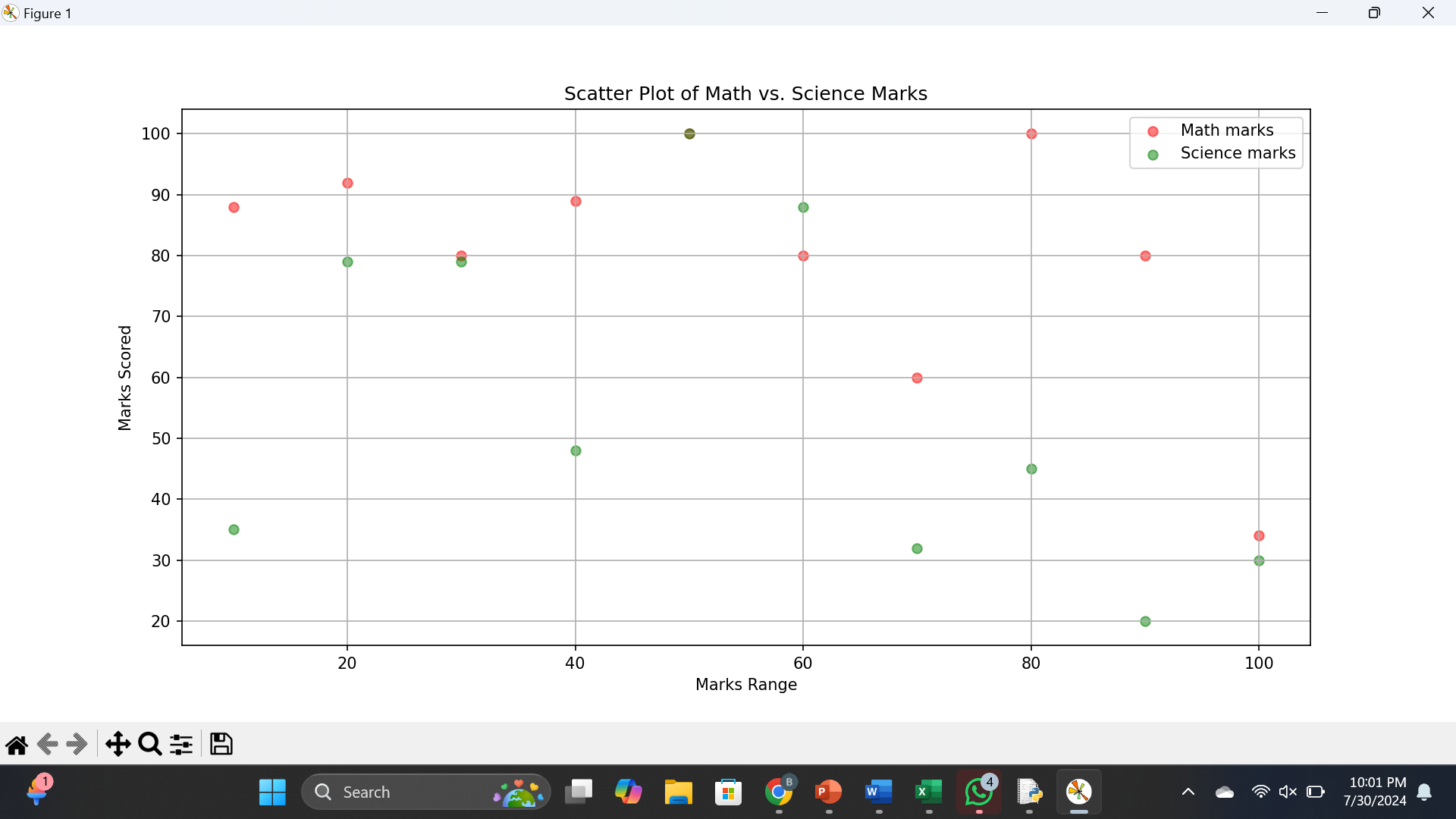
plt.legend()

# Display the plot

plt.grid(True)

plt.show()

Output:



Results:

Thus a Python program to draw a scatter plot comparing two subject marks of Mathematics and Science. Use marks of 10 students.

Experiment 36

Aim:

To Write a Python program to draw a scatter plot for three different groups comparing weights and heights.

Code

import matplotlib.pyplot as plt

# Sample data

group1\_weights = [60, 65, 70, 75, 80]

group1\_heights = [160, 165, 170, 175, 180]

group2\_weights = [55, 62, 64, 72, 76]

group2\_heights = [155, 160, 165, 170, 175]

group3\_weights = [71, 78, 85, 90, 95]

group3\_heights = [165, 170, 175, 180, 185]

# Create scatter plots

plt.figure(figsize=(10, 6))

plt.scatter(group1\_weights, group1\_heights, color='red', label='Group 1')

plt.scatter(group2\_weights, group2\_heights, color='blue', label='Group 2')

plt.scatter(group3\_weights, group3\_heights, color='green', label='Group 3')

# Add titles and labels

plt.title('Scatter Plot of Weights vs Heights for Three Groups')

plt.xlabel('Weight (kg)')

plt.ylabel('Height (cm)')

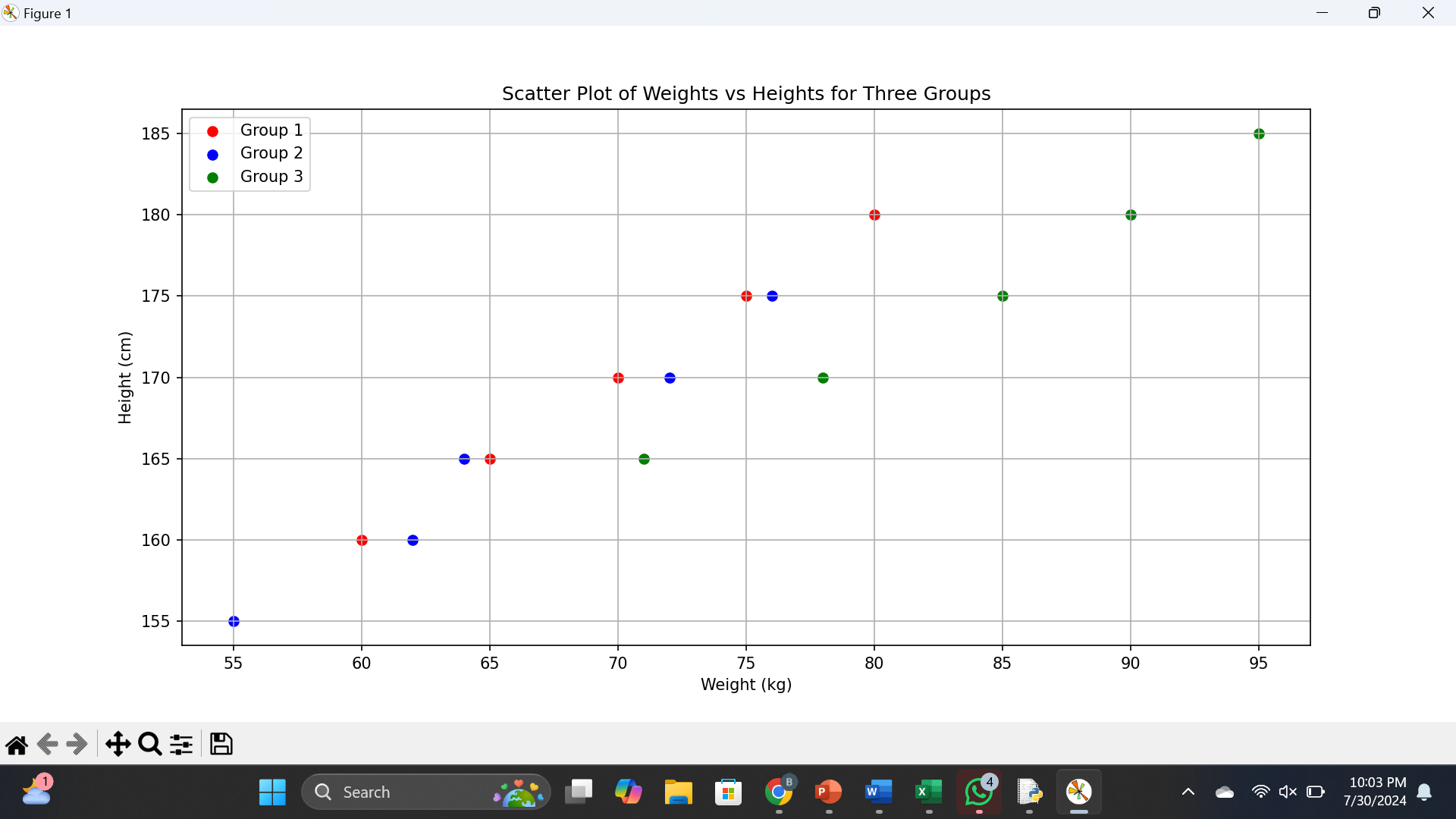
plt.legend()

# Show the plot

plt.grid(True)

plt.show()

Output:



Results:

Thus a Python program to draw a scatter plot for three different groups comparing weights and heights.

Experiment 37

Aim:

To Write a Pandas program to create a dataframe from a dictionary and display it.

Code

import pandas as pd

# Sample data

data = {'X': [78, 85, 96, 80, 86],

'Y': [84, 94, 89, 83, 86],

'Z': [86, 97, 96, 72, 83]}

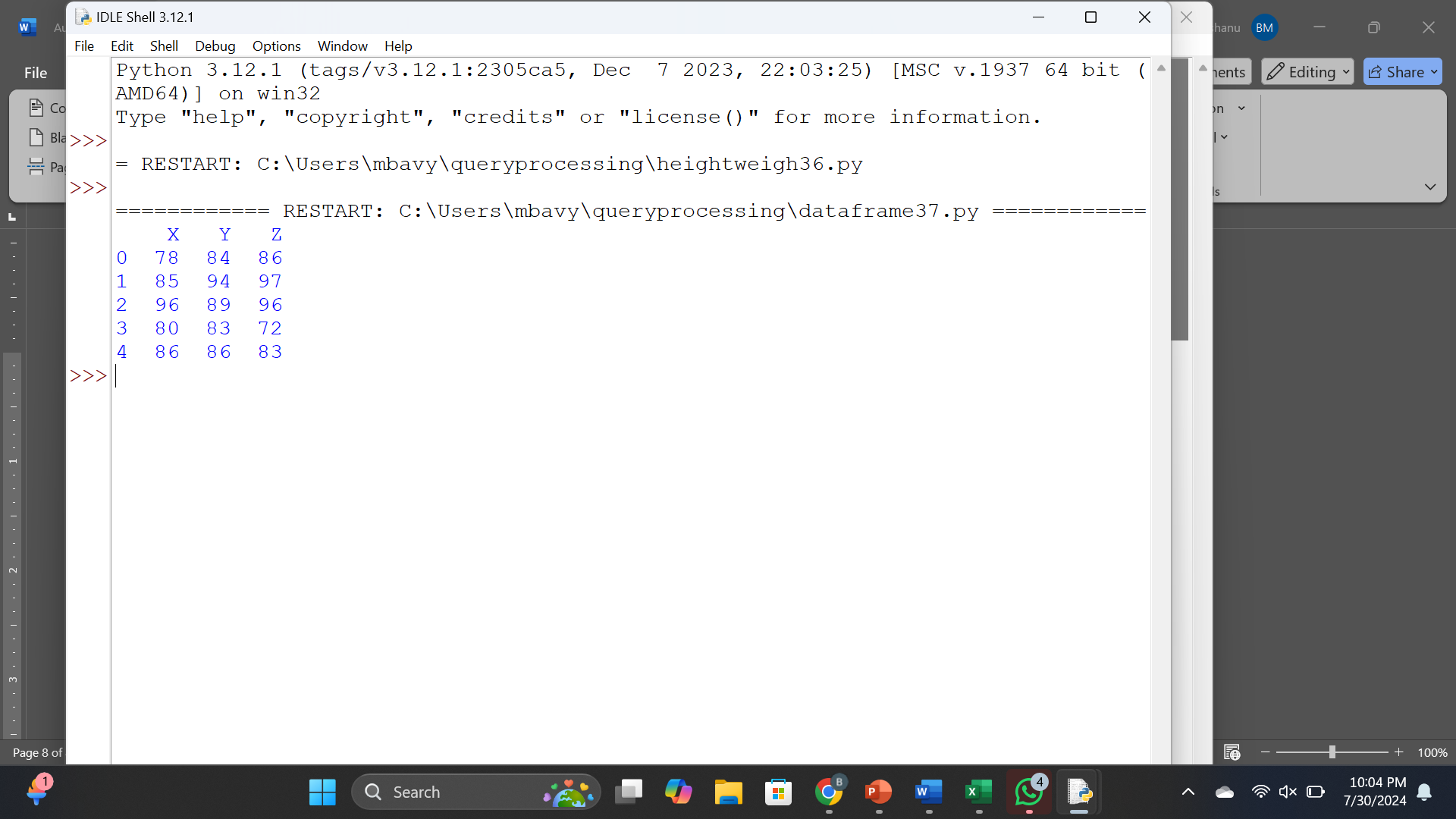
# Create DataFrame

df = pd.DataFrame(data)

# Display DataFrame

print(df)

Output:



Results:

Thus a Pandas program to create a dataframe from a dictionary and display it.

Experiment 38

Aim:

To Write a Pandas program to create and display a DataFrame from a specified dictionary data which has the index labels.

Code:

import pandas as pd

import numpy as np

# Dictionary data

exam\_data = {

'name': ['Anastasia', 'Dima', 'Katherine', 'James', 'Emily', 'Michael', 'Matthew', 'Laura', 'Kevin', 'Jonas'],

'score': [12.5, 9, 16.5, np.nan, 9, 20, 14.5, np.nan, 8, 19],

'attempts': [1, 3, 2, 3, 2, 3, 1, 1, 2, 1],

'qualify': ['yes', 'no', 'yes', 'no', 'no', 'yes', 'yes', 'no', 'no', 'yes']

}

# Index labels

labels = ['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j']

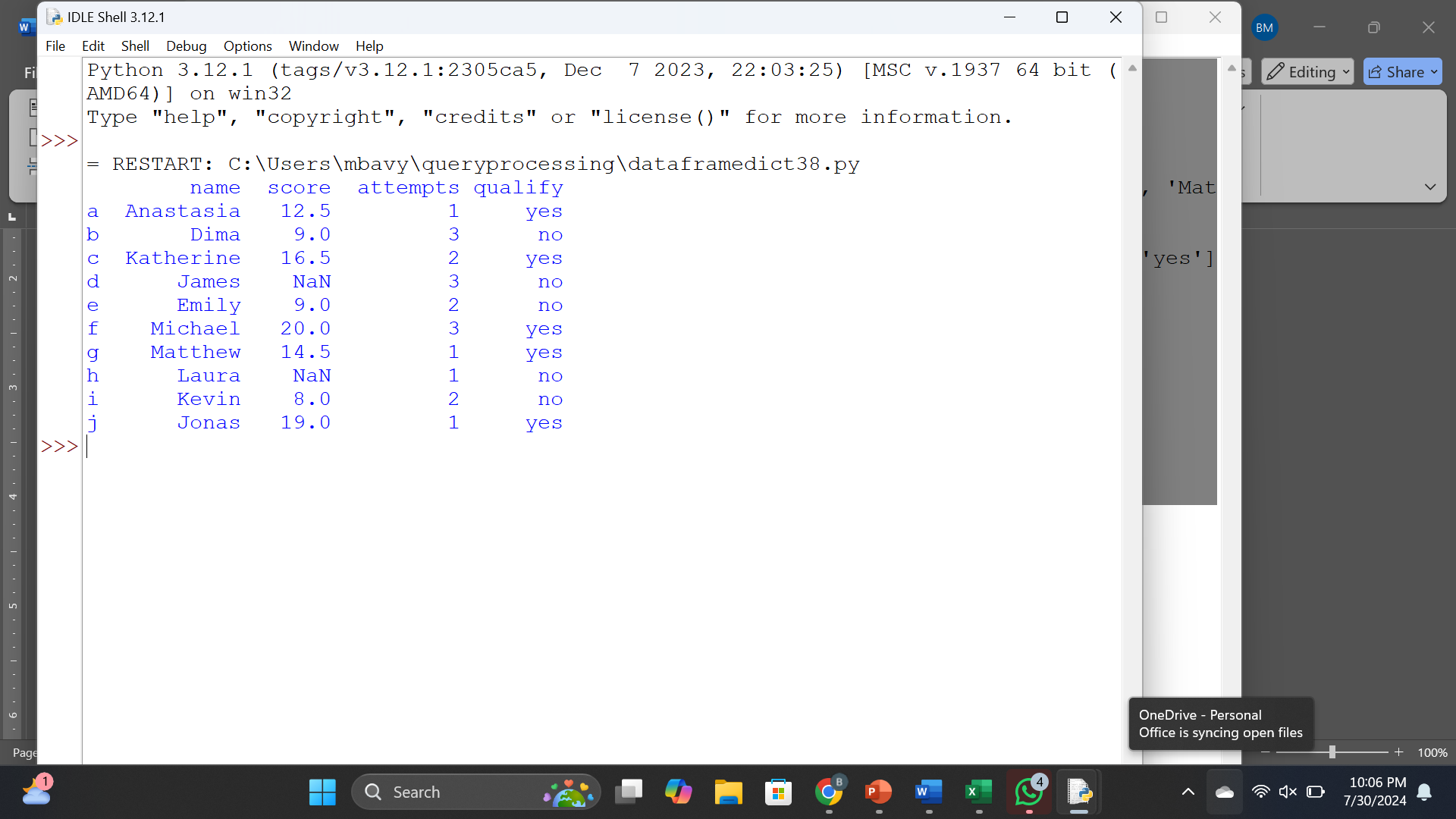
# Creating DataFrame

df = pd.DataFrame(exam\_data, index=labels)

# Display DataFrame

print(df)

Output:



Results:

Thus a Pandas program to create and display a DataFrame from a specified dictionary data which has the index labels.

Experiment 39

Aim:

To Write a Pandas program to get the first 3 rows of a given DataFrame.  
Sample Python dictionary data and list labels:

Code:

import pandas as pd

import numpy as np

# Dictionary data

exam\_data = {

'name': ['Anastasia', 'Dima', 'Katherine', 'James', 'Emily', 'Michael', 'Matthew', 'Laura', 'Kevin', 'Jonas'],

'score': [12.5, 9, 16.5, np.nan, 9, 20, 14.5, np.nan, 8, 19],

'attempts': [1, 3, 2, 3, 2, 3, 1, 1, 2, 1],

'qualify': ['yes', 'no', 'yes', 'no', 'no', 'yes', 'yes', 'no', 'no', 'yes']

}

# Index labels

labels = ['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j']

# Creating DataFrame

df = pd.DataFrame(exam\_data, index=labels)

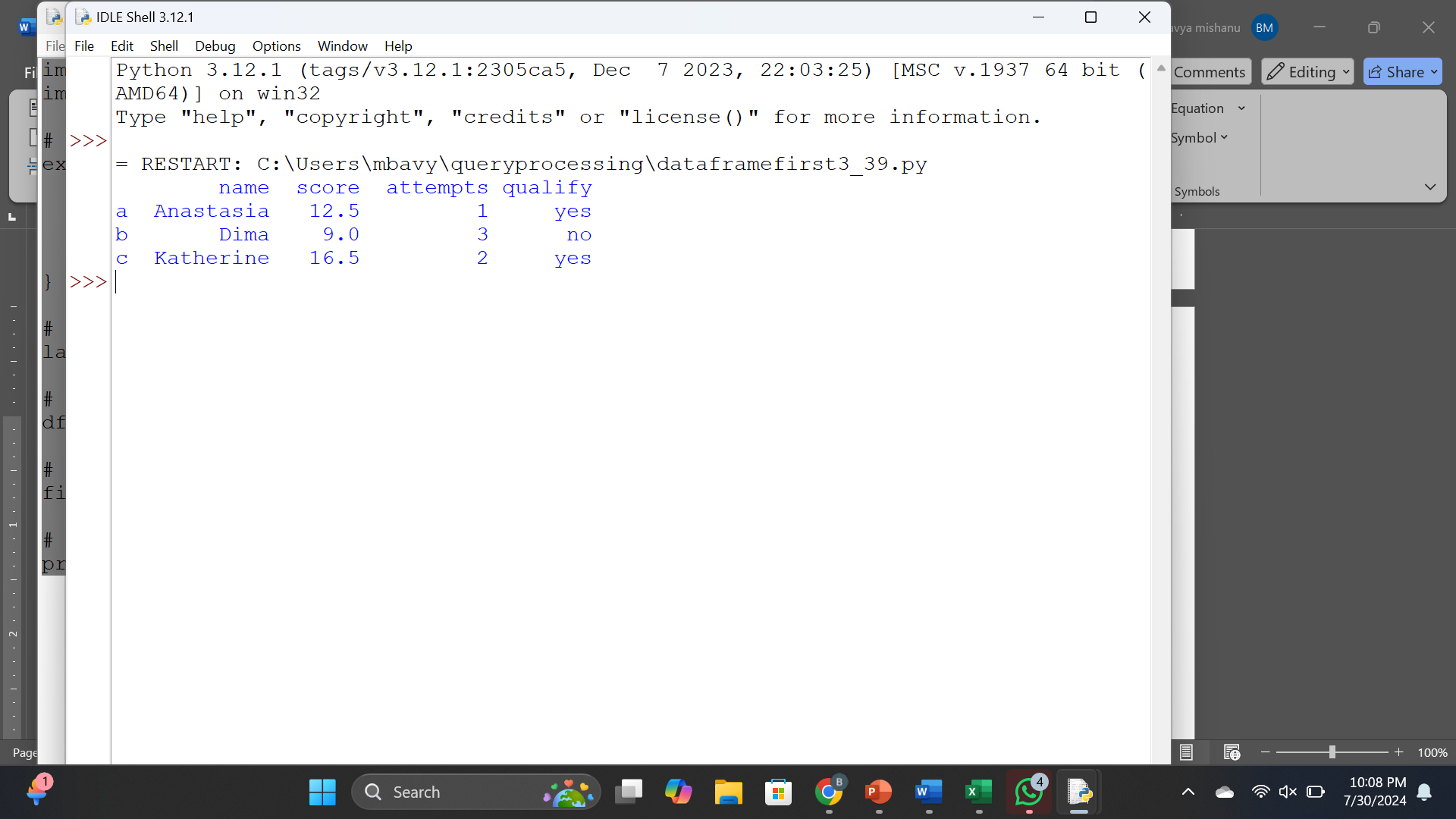
# Get the first 3 rows

first\_3\_rows = df.head(3)

# Display the result

print(first\_3\_rows)

Output:



Results:

Thus a Pandas program to get the first 3 rows of a given DataFrame.  
Sample Python dictionary data and list labels:

Experiment 40

Aim:

To Write a Pandas program to select the 'name' and 'score' columns from the following DataFrame.

Code:

import pandas as pd

import numpy as np

# Dictionary data

exam\_data = {

'name': ['Anastasia', 'Dima', 'Katherine', 'James', 'Emily', 'Michael', 'Matthew', 'Laura', 'Kevin', 'Jonas'],

'score': [12.5, 9, 16.5, np.nan, 9, 20, 14.5, np.nan, 8, 19],

'attempts': [1, 3, 2, 3, 2, 3, 1, 1, 2, 1],

'qualify': ['yes', 'no', 'yes', 'no', 'no', 'yes', 'yes', 'no', 'no', 'yes']

}

# Index labels

labels = ['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j']

# Creating DataFrame

df = pd.DataFrame(exam\_data, index=labels)

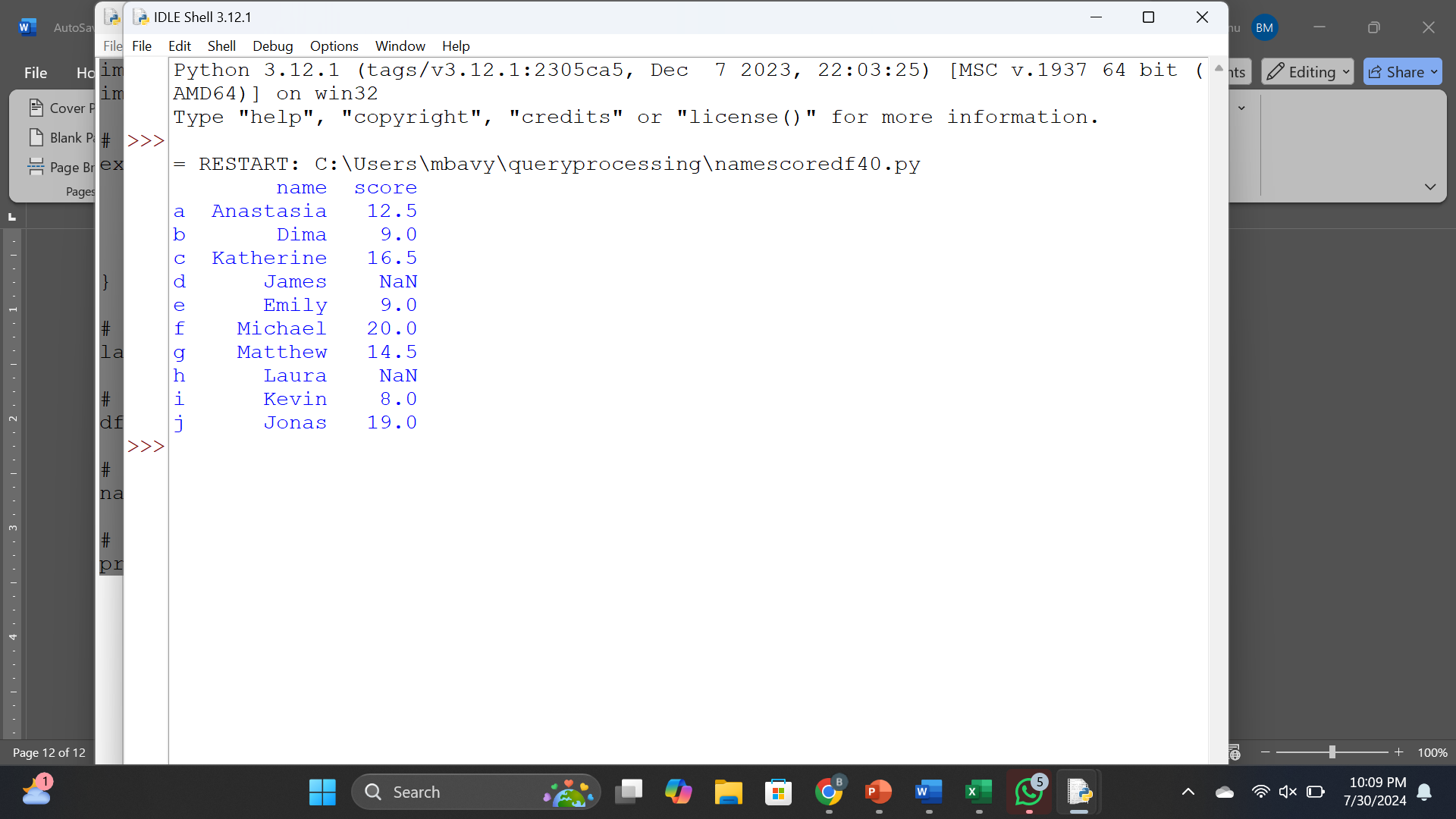
# Select 'name' and 'score' columns

name\_score = df[['name', 'score']]

# Display the result

print(name\_score)

Output



Results:

Thus a Pandas program to select the 'name' and 'score' columns from the following DataFrame.