

Assignment: Assignment 2 : Module 2 : Array

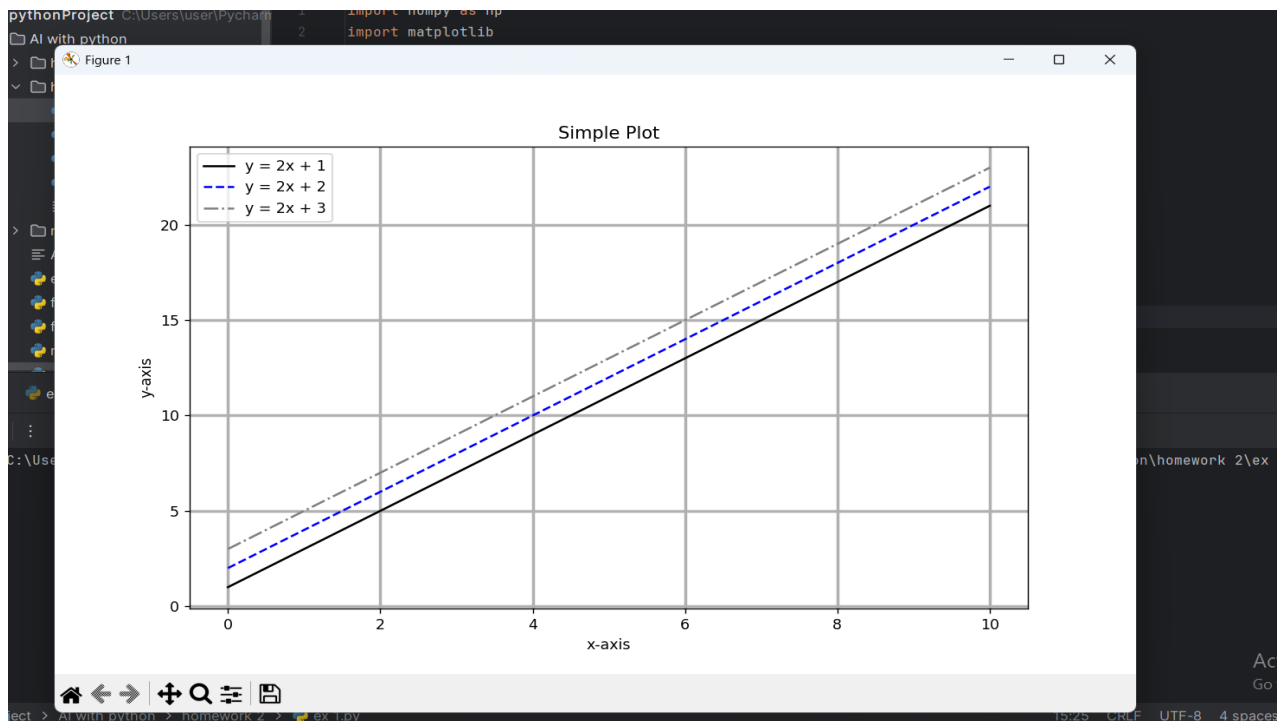
1. Here,

```
import numpy as np
import matplotlib
import matplotlib.pyplot as plt
matplotlib.use('TkAgg')
x = np.linspace(0, 10, 100)
y1 = 2 * x + 1
y2 = 2 * x + 2
y3 = 2 * x + 3

plt.figure(figsize = (10, 6))
plt.plot(x, y1, label = "y = 2x + 1", linestyle="-", color="black")
plt.plot(x, y2, label = "y = 2x + 2", linestyle="--", color="blue")
plt.plot(x, y3, label = "y = 2x + 3", linestyle="-.", color="grey")

plt.title("Simple Plot")
plt.xlabel("x-axis")
plt.ylabel("y-axis")

plt.grid(True, linestyle = '-', linewidth=2)
plt.legend()
plt.show()
```



2. Here,

```
import numpy as np
import matplotlib
import matplotlib.pyplot as plt
```

```

matplotlib.use('TkAgg')

x = np.array([1, 2, 3, 4, 5, 6, 7, 8, 9])
y = np.array([-0.57, -2.57, -4.80, -7.36, -8.78, -10.52, -12.85, -14.69, -16.78])

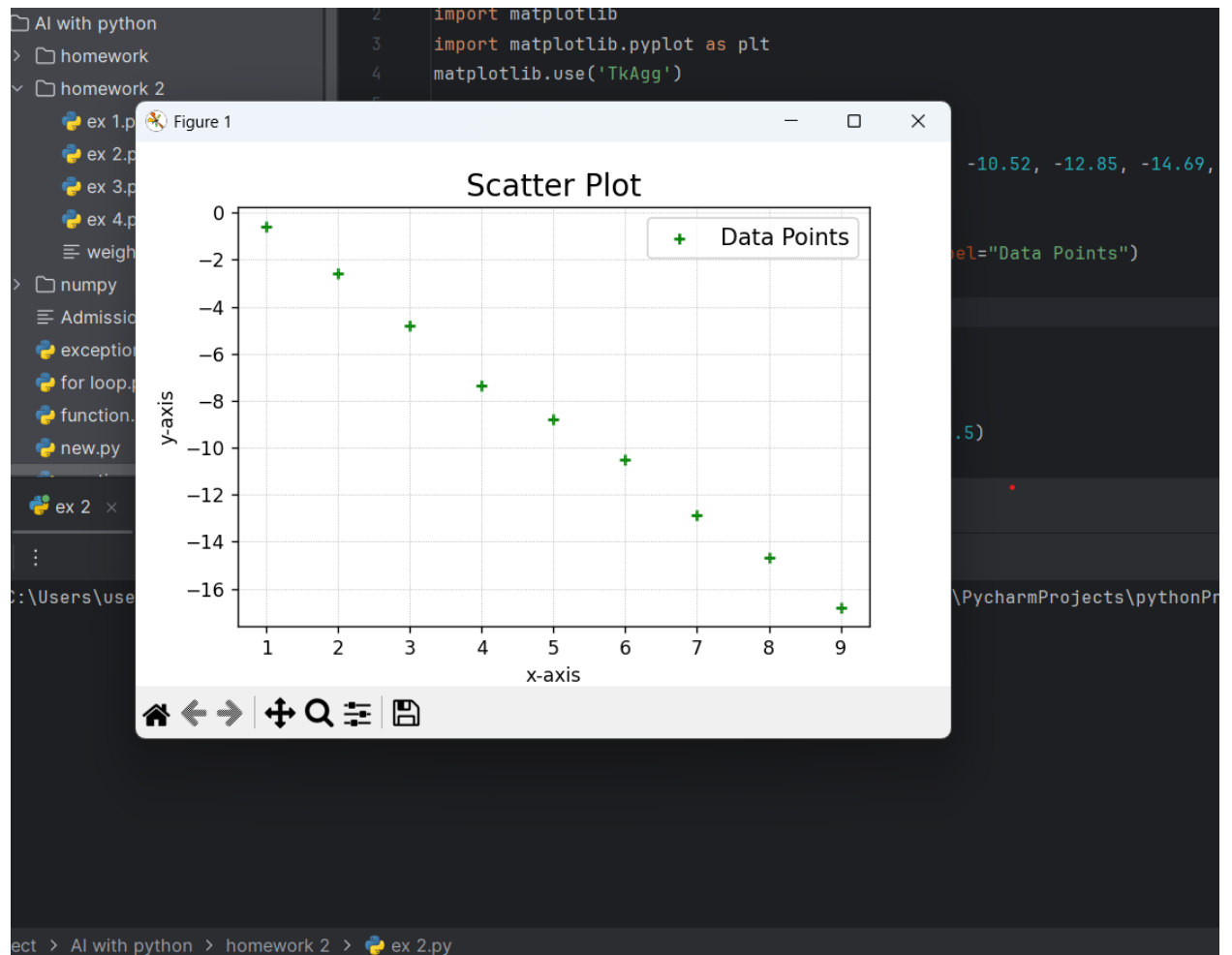
plt.figure(figsize=(6,4))
plt.scatter(x, y, marker='+', color='green', label="Data Points")

plt.title("Scatter Plot", fontsize=16)
plt.xlabel("x-axis", fontsize=10)
plt.ylabel("y-axis", fontsize=10)

plt.grid(True, linestyle=':', linewidth=0.5)
plt.legend(fontsize=12)

plt.show()

```



3. Here,

```

import pandas as pd
import numpy as np

```

```
import matplotlib
import matplotlib.pyplot as plt
matplotlib.use('TkAgg')

df = pd.read_csv('weight-height.csv')
length_in_inches = df['Height'].values
weight_in_pounds = df['Weight'].values

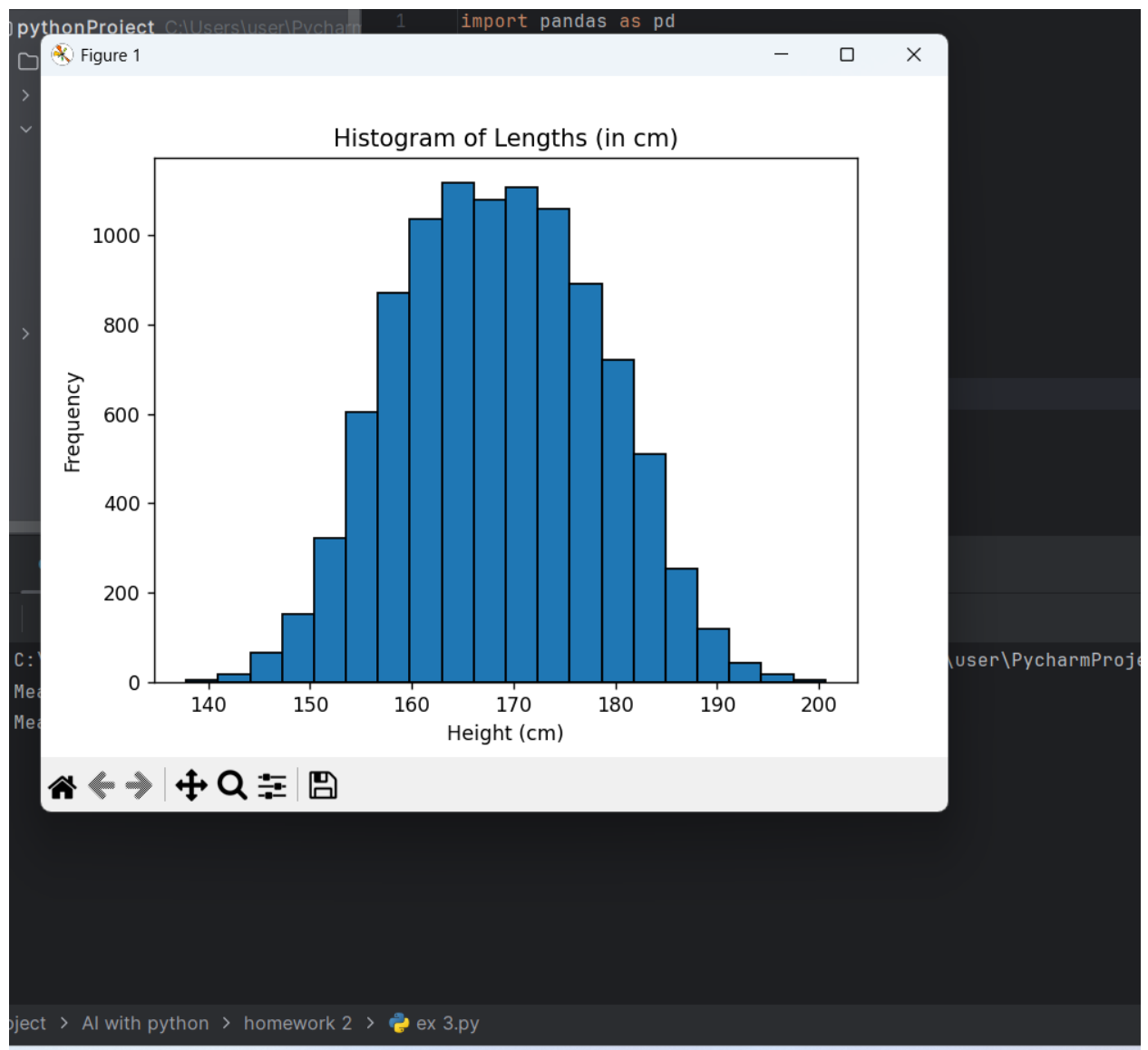
length_in_cm = length_in_inches * 2.54

weight_in_kg = weight_in_pounds * 0.453592

mean_length = np.mean(length_in_cm)
mean_weight = np.mean(weight_in_kg)

print(f"Mean Height (cm): {mean_length}")
print(f"Mean weight (kg): {mean_weight}")

plt.hist(length_in_cm, bins=20, edgecolor='black')
plt.title('Histogram of Lengths (in cm)')
plt.xlabel('Height (cm)')
plt.ylabel('Frequency')
plt.show()
```



4. Here,

```
import numpy as np

A = np.array([[1, 2, 3],
              [0, 1, 4],
              [5, 6, 0]])

A_inv = np.linalg.inv(A)

product_A_A_inv = np.dot(A, A_inv)
product_A_inv_A = np.dot(A_inv, A)

print("Inverse of A:")
print(A_inv)
```

```

print("\nA * A_inv (close to identity matrix):")
print(product_A_A_inv)

print("\nA_inv * A (close to identity matrix):")
print(product_A_inv_A)

```

The screenshot shows a PyCharm IDE with a project named 'pythonProject'. The file explorer on the left shows a directory structure: 'AI with python' > 'homework' > 'homework 2', containing files 'ex 1.py', 'ex 2.py', 'ex 3.py', and 'ex 4.py'. The 'Run' tab is active, showing the execution of 'ex 4.py'. The code in the editor is as follows:

```

10 product_A_inv_A = np.dot(A_inv, A)
11
12 print("Inverse of A:")
13 print(A_inv)
14
15 print("\nA * A_inv (close to identity matrix):")
16 print(product_A_A_inv)
17
18 print("\nA_inv * A (close to identity matrix):")
19 print(product_A_inv_A)

```

The Run console output is as follows:

```

C:\Users\User\PycharmProjects\pythonProjectTest\.venv\Scripts\python.exe "C:\Users\User\PycharmProjects\pythonProject\AI with python\homework 2\ex 4.py"
Inverse of A:
[[-24.  18.   5.]
 [ 20. -15.  -4.]
 [-5.   4.   1.]]

A * A_inv (close to identity matrix):
[[ 1.00000000e+00 -2.66453526e-15  0.00000000e+00]
 [ 0.00000000e+00  1.00000000e+00  0.00000000e+00]
 [ 0.00000000e+00 -7.18542736e-15  1.00000000e+00]]

A_inv * A (close to identity matrix):
[[ 1.00000000e+00  0.00000000e+00  0.00000000e+00]
 [ 0.00000000e+00  1.00000000e+00  0.00000000e+00]
 [ 0.00000000e+00 -8.817842e-16  1.00000000e+00]]

Process finished with exit code 0

```

At the bottom right of the Run console, there is a watermark: 'Activate Windows Go to Settings to activate W'.