NATIONAL UNIVERSITY OF SCIENCE AND TECHNOLOGY

DEPARTMENT OF COMPUTER SCIENCE

ARTIFICIAL INTELLIGENCE LAB

NAME	Ayesha Imran
Class	CS-A
Lab	01
Course	Artificial Intelligence
Date	17-September-25
Submitted To	Lec. Ijlal Haider

IN LAB TASKS

Task 1

Write a Python program to extract all odd numbers from a given list of integers. Use list comprehension to achieve this and print the result. For example, given the list [1, 2, 3, 4, 5, 6,7, 8, 9, 10], the output should be [1, 3, 5, 7, 9].

Solution:

```
lists =[1,2,3,4,5,6,7,8,9,10];
for num in lists :
    if num%2!=0:
      print(num);
```

Output:

```
1
3
5
7
9
PS C:\Users\hp\Desktop\Python practice>
```

Task 2

```
Create a numpy array of size 5-by-5 containing all
random values. Determine the transpose  and inverse
of this matrix.
```

Solution:

```
import numpy as np;
matrix=np.random.rand(5,5);
transpose = matrix.T
inverse = np.linalg.inv(matrix)
print('Original matrix \n ', matrix);
print('\n Transpose: \n', transpose);
print('Inverse:\n', inverse);
```

Output:

```
Original matrix
  [[0.88441701 0.66206118 0.30126241 0.12479478 0.49620185]
 [0.54828477 0.11426042 0.98070535 0.82793853 0.59006235]
 [0.90276557 0.86402656 0.1562652  0.99164445 0.11934238]
 [0.63139619 0.24943068 0.6135702  0.56745394 0.92815516]
 [0.07675162 0.87678887 0.45734872 0.85197392 0.70812016]]

 Transpose:
 [[0.88441701 0.54828477 0.90276557 0.63139619 0.07675162]
 [0.66206118 0.11426042 0.86402656 0.24943068 0.87678887]
 [0.30126241 0.98070535 0.1562652  0.6135702  0.45734872]
 [0.12479478 0.82793853 0.99164445 0.56745394 0.85197392]
 [0.49620185 0.59006235 0.11934238 0.92815516 0.70812016]]
Inverse:
 [[ 0.36517483 -0.06971369  0.52944808  0.48854699 -0.92738217]
 [ 1.09345087  0.10303403 -0.27549385 -1.34330072  0.95506495]
 [ 1.10651237  1.86702431 -0.8958368  -1.93336109  0.35397401]
 [-1.40999466 -0.18820465  0.93996389  0.80244702 -0.06535393]
 [-0.41170506 -1.09942383 -0.26859983  1.89353547  0.18016439]]
```

Task 3:

Write a Python program to create a pandas DataFrame from a dictionary containing data for countries, their capitals, and populations. The keys of the dictionary should represent the columns, Country, Capital, and Population. The values shuld be lists representing the data for each column.

Solution :

```
import pandas as pd;
data = {
    'Country': ['Pakistan', 'India', 'China', 'USA', 'UK'],
    'Capital': ['Islamabad', 'New Delhi', 'Beijing', 'Washington', 'London'],
    'Population': [240_000_000, 1_400_000_000, 1_410_000_000, 331_000_000, 67_000_000]
}
df = pd.DataFrame(data)

print(df)
```

Output:

```
    Country     Capital  Population
0  Pakistan   Islamabad   240000000
1     India   New Delhi  1400000000
2     China     Beijing  1410000000
3       USA  Washington   331000000
4        UK      London    67000000
```

Task 4:

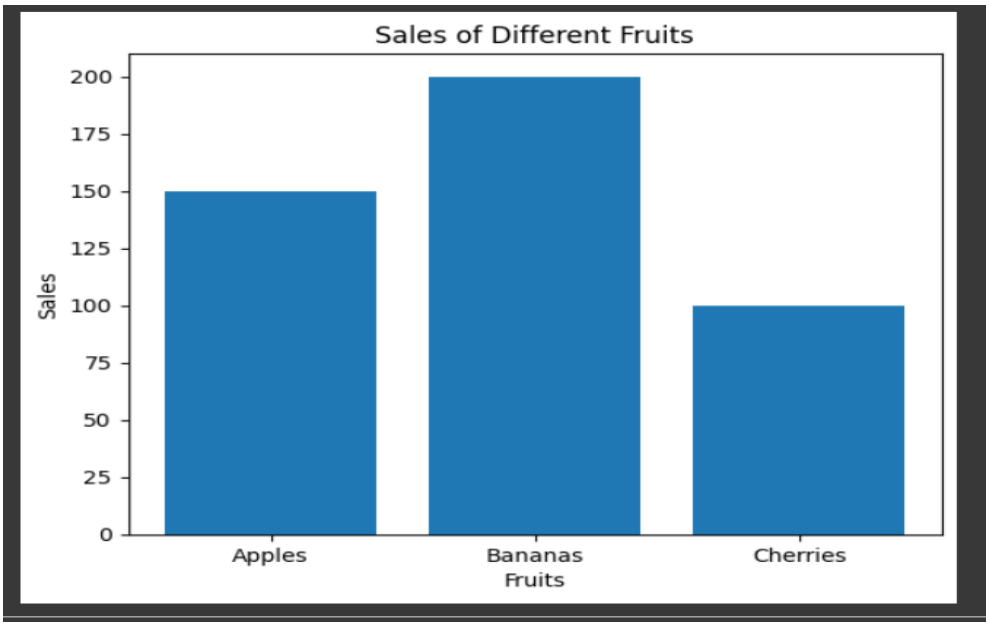
Create a bar chart showing the sales of different fruits. Use the following data:

- Fruits: [Apples, Bananas, Cherries]
- Sales: [150, 200, 100]

Solution :

```
 import matplotlib.pyplot as plt

fruits = ['Apples', 'Bananas', 'Cherries']
sales = [150, 200, 100]

plt.bar(fruits, sales)
plt.xlabel('Fruits')
plt.ylabel('Sales')
plt.title('Sales of Different Fruits')
plt.show()
```

Output:



POST LAB TASKS

Task 1:

Write a Python program using NumPy to count the number of non-zero values in a given NumPy array.

Solution :

```
import numpy as np

# Create a sample NumPy array
my_array = np.array([1, 0, 3, 0, 5, 0, 7])

# Count the number of non-zero values
non_zero_count = np.count_nonzero(my_array)

# Print the result
print(f"The number of non-zero values in the
array is: {non_zero_count}")
```

Output:

```
The number of non-zero values in the array is: 4
```

Task 2

Write a Python program using NumPy to create a 4x5 array, then find and print its transpose.The transpose should swap the rows and columns of the original array.

Solution:

```
import numpy as np

# Create a 4x5 NumPy array
my_array = np.arange(20).reshape(4, 5)
```

```
print("Original array:")
print(my_array)

# Find the transpose of the array
transpose_array = np.transpose(my_array)
# Alternatively, you can use: transpose_array =
my_array.T

# Print the transpose array
print("\nTranspose of the array:")
print(transpose_array)
```

Output :

```
Original array:
[[ 0  1  2  3  4]
 [ 5  6  7  8  9]
 [10 11 12 13 14]
 [15 16 17 18 19]]

Transpose of the array:
[[ 0  5 10 15]
 [ 1  6 11 16]
 [ 2  7 12 17]
 [ 3  8 13 18]
 [ 4  9 14 19]]
```

Task 3

Write a Python program to create a pandas DataFrame from a dictionary containing the following data:

- **Name: [Alice, Bob, Charlie]**
- **Age: [25, 30, 35]**
- **City: [New York , Los Angeles , Chicago]**

Solution:

```
import pandas as pd

# Create a dictionary with the data
data = {
    'Name': ['Alice', 'Bob', 'Charlie'],
    'Age': [25, 30, 35],
    'City': ['New York', 'Los Angeles',
'Chicago']
}

# Create a pandas DataFrame from the dictionary
df = pd.DataFrame(data)

# Print the DataFrame
print(df)
```

Output :

```
      Name  Age          City
0    Alice   25      New York
1      Bob   30   Los Angeles
2  Charlie   35       Chicago
```

Task 4:

Plot a scatter diagram to show the disadvantages of smoking with increasing age and weight.

The age vector: [10, 20, 30, 40, 50]

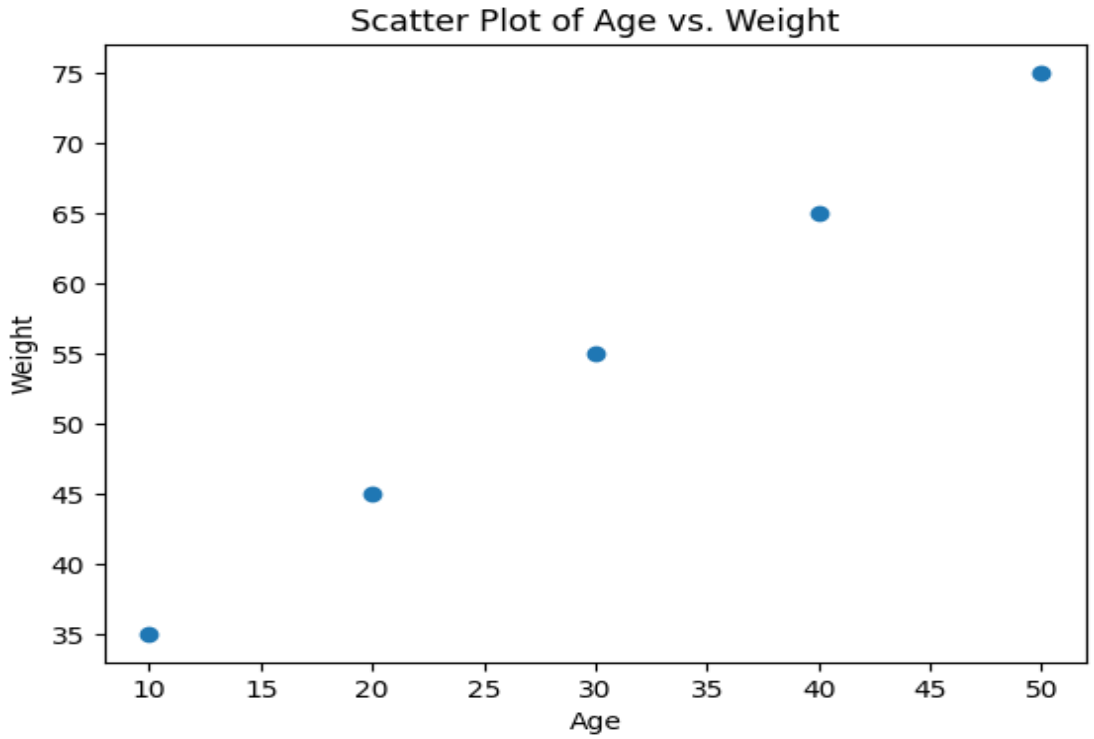
The weight vector: [35, 45, 55, 65, 75]

Solution:

```
import matplotlib.pyplot as plt
```

```
import numpy as np

# Age and weight data
age = np.array([10, 20, 30, 40, 50])
weight = np.array([35, 45, 55, 65, 75])

# Create the scatter plot
plt.scatter(age, weight)

# Add labels and title
plt.xlabel("Age")
plt.ylabel("Weight")
plt.title("Scatter Plot of Age vs. Weight")

# Show the plot
plt.show()
```

Output :



Task 5:

Given the 1D NumPy array arr = np.array([10, 20, 30, 40, 50, 60]), reshape it into a 2D array with 2 rows and 3 columns. What will the resulting array look like?

Solution:

```
import numpy as np

# Given 1D NumPy array
arr = np.array([10, 20, 30, 40, 50, 60])

# Reshape the array into a 2D array with 2 rows and 3 columns
reshaped_arr = arr.reshape(2, 3)

# Print the resulting array
print("Original array:", arr)
print("Reshaped array (2x3):")
print(reshaped_arr)
```

Output :

```
Original array: [10 20 30 40 50 60]
Reshaped array (2x3):
[[10 20 30]
 [40 50 60]]
```

Task 6:

Given a NumPy array arr = np.array([1, 2, 3, 4, 5, 6, 7, 8]), can you reshape it into a 3D array with dimensions (2, 2, 2)? Write the code and check if the reshaping is successful.What error (if any) would you encounter?

Solution:

```
import numpy as np

# Given 1D NumPy array
arr = np.array([1, 2, 3, 4, 5, 6, 7, 8])

# Attempt to reshape the array into a 3D array
with dimensions (2, 2, 2)
try:
    reshaped_arr_3d = arr.reshape(2, 2, 2)
    print("Original array:", arr)
    print("\nReshaped 3D array (2x2x2):")
    print(reshaped_arr_3d)
    print("\nReshaping was successful.")
except ValueError as e:
    print("Error during reshaping:", e)
    print("\nReshaping was not successful. The
error message indicates the reason.")
```

Output :

```
Original array: [1 2 3 4 5 6 7 8]
Reshaped 3D array (2x2x2):
[[[1 2]
[3 4]]
[[5 6]
 [7 8]]]
Reshaping was successful.
```