ASSIGNMENT 7 – NUMPY & PANDAS

1. Create a numpy array containing the numbers from 1 to 10, and then reshape it to a 2x5 matrix.

import numpy as np

arr = np.arange(1, 11).reshape(2, 5)

print("Array:\n", arr)

print("Array Shape:\n",arr.shape)

print("Array Dimension:\n",arr.ndim)

```
# Exercise 1: (Score : 1)
# Create a numpy array containing the numbers from 1 to 10, and then reshape it to a 2x5 matrix.
import numpy as np
arr = np.arange(1, 11).reshape(2, 5)
print("Array:\n", arr)
print("Array Shape:\n",arr.shape)
print("Array Dimension:\n",arr.ndim)

Array:
 [[ 1  2  3  4  5]
 [ 6  7  8  9 10]]
Array Shape:
 (2, 5)
Array Dimension:
 2
```

2. Create a numpy array containing the numbers from 1 to 20, and then extract the elements between the 5th and 15th index.

arr = np.arange(1, 21)

extracted_elements = arr[5:16]

print("Array:\n", arr)

print("Extracted elements (between 5th and 15th index): ", extracted_elements)

```
# Exercise 2: (Score : 1)
# Create a numpy array containing the numbers from 1 to 20, and then extract the elements between the 5th and 15th index.
arr = np.arange(1, 21)
extracted_elements = arr[5:16]
print("Array:\n", arr)
print("Extracted elements (between 5th and 15th index): ", extracted_elements)

Array:
 [ 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20]
Extracted elements (between 5th and 15th index):  [ 6  7  8  9 10 11 12 13 14 15 16]
```

3. Create a Pandas series with the following data: {'apples': 3, 'bananas': 2, 'oranges': 1}. Then, add a new item to the series with the key 'pears' and the value 4

import pandas as pd

fruits = pd.Series({'apples': 3, 'bananas': 2, 'oranges': 1})

print("Fruits List:\n",fruits)

fruits["pears"] = 4

print("\n Updated Fruits List:\n",fruits)

```
[3]: # Exercise 3: (Score : 2)
     # Create a Pandas series with the following data: {'apples': 3, 'bananas': 2, 'oranges': 1}.
     # Then, add a new item to the series with the key 'pears' and the value 4
     import pandas as pd
     fruits = pd.Series({'apples': 3, 'bananas': 2, 'oranges': 1})
     print("Fruits List:\n",fruits)
     fruits["pears"] = 4
     print("\n Updated Fruits List:\n",fruits)

     Fruits List:
      apples     3
     bananas    2
     oranges    1
     dtype: int64

      Updated Fruits List:
      apples     3
     bananas    2
     oranges    1
     pears      4
     dtype: int64
```

4. Create a dataframe with the following columns: name, age, and gender. The dataframe should have 10 rows of data.

df = pd.DataFrame({

   "name": ["Aadhya", "Arun", "Nithya", "Vivek", "Meera", "Sreeja", "Santhosh", "Anjali", "Ravi", "Anupama"],

   "age": [24, 30, 28, 32, 27, 29, 35, 22, 31, 26],

   "gender": ["Female", "Male", "Female", "Male", "Female", "Female", "Male", "Female", "Male", "Female"]

})

df

```
[ ]:  # Exercise 4: (Score : 2)
      # Create a dataframe with the following columns: name, age, and gender. The dataframe should have 10 rows of data.

      df = pd.DataFrame({
          "name": ["Aadhya", "Arun", "Nithya", "Vivek", "Meera", "Sreeja", "Santhosh", "Anjali", "Ravi", "Anupama"],
          "age": [24, 30, 28, 32, 27, 29, 35, 22, 31, 26],
          "gender": ["Female", "Male", "Female", "Male", "Female", "Female", "Male", "Female", "Male", "Female"]
      })

[9]:  df
```

[9]:

	name	age	gender
0	Aadhya	24	Female
1	Arun	30	Male
2	Nithya	28	Female
3	Vivek	32	Male
4	Meera	27	Female
5	Sreeja	29	Female
6	Santhosh	35	Male
7	Anjali	22	Female
8	Ravi	31	Male
9	Anupama	26	Female

5. Add a new column to the data frame created in question 1, called occupation. The values for this column should be Programmer, Manager, and Analyst, corresponding to the rows in the dataframe.

df["occupation"] = ["Programmer", "Manager", "Analyst", "Programmer", "Manager",

"Analyst", "Programmer", "Manager", "Analyst", "Programmer"]

df

```
[23]:  # Exercise 5: (Score : 1)
       # Add a new column to the data frame created in question 1, called occupation.
       # The values for this column should be Programmer, Manager, and Analyst, corresponding to the rows in the dataframe.
       df["occupation"] = ["Programmer", "Manager", "Analyst", "Programmer", "Manager",
                           "Analyst", "Programmer", "Manager", "Analyst", "Programmer"]
       df
```

[23]:

	name	age	gender	occupation
0	Aadhya	24	Female	Programmer
1	Arun	30	Male	Manager
2	Nithya	28	Female	Analyst
3	Vivek	32	Male	Programmer
4	Meera	27	Female	Manager
5	Sreeja	29	Female	Analyst
6	Santhosh	35	Male	Programmer
7	Anjali	22	Female	Manager
8	Ravi	31	Male	Analyst
9	Anupama	26	Female	Programmer

6. Select the rows of the dataframe where the age is greater than or equal to 30.

df_selected_age = df[df["age"] >= 30]

print(df_selected_age)

```
[27]:  # Exercise 6: (Score : 1)
       # Select the rows of the dataframe where the age is greater than or equal to 30.

       df_selected_age = df[df["age"] >= 30]
       print(df_selected_age)

            name  age gender   occupation
       1     Arun   30   Male      Manager
       3    Vivek   32   Male   Programmer
       6  Santhosh   35   Male   Programmer
       8     Ravi   31   Male      Analyst
```

7. Convert this dataframe to a csv file and read that csv file, finally display the contents.

df_selected_age.to_csv("selected_age_data.csv", index=False)

print("File saved as selected_age_data.csv \n")

df_from_csv = pd.read_csv("selected_age_data.csv")

print(df_from_csv)

```
[33]:  # Exercise 7: (Score : 2)
       # Convert this dataframe to a csv file and read that csv file, finally display the contents.

       df_selected_age.to_csv("selected_age_data.csv", index=False)
       print("File saved as selected_age_data.csv \n")
       df_from_csv = pd.read_csv("selected_age_data.csv")
       print(df_from_csv)

       File saved as selected_age_data.csv

            name  age gender   occupation
       0     Arun   30   Male      Manager
       1    Vivek   32   Male   Programmer
       2  Santhosh   35   Male   Programmer
       3     Ravi   31   Male      Analyst
```