ASSIGNMENT 2

CONTENTS

- What is dataset
- How to upload dataset
- Numpy libraries
- Pandas libraries

WHAT IS DATASET

- A dataset is a collection of data.
- In the case of tabular data, a data set corresponds to one or more database tables where every column of a table represents a particular variable.
- Each row corresponds to a given record of the data set in question.
- Each value is known as a datum.

IMPORT OF DATASET

- Datasets are generally imported as a raw data files.
- Datasets are generally in the form of CSV, JSON or XML data format.
- For the purpose of this tutorial, CSV is used in the accompanying examples.

IMPORTING THE DATASET - PYTHON

```
import pandas as pd                          # use pandas library for data frames

dataset = pd.read_csv( ‘data.csv’ )          # read CSV file into a data frame
```

CSV data converted to data frame.

Function to read a CSV file

pathname to raw data file

Data Frame adds these indices

Example Data (CSV File):

```
Age, Gender, Income, Spending
22,M,18000,6000
25,F,30000,8000
31,F,35000,12000
35,M,40000,18000
```

Generated Data Frame:

	Age	Gender	Income	Spending
0	22	M	18000	6000
1	25	F	30000	8000
2	31	F	35000	12000
3	35	M	40000	18000

CATEGORICAL VARIABLES

Independent Variables (Features)

Dependent Variables (Label)

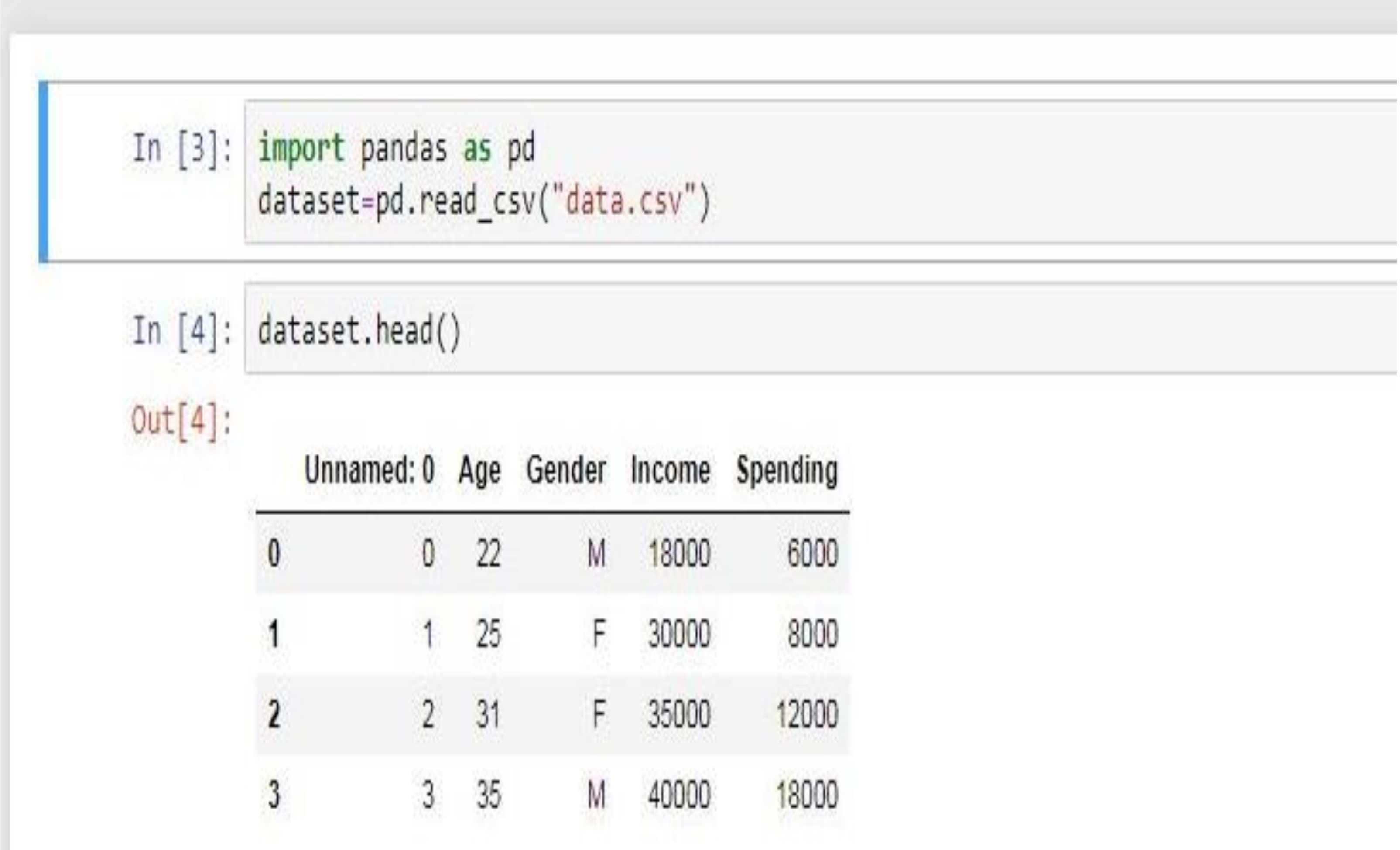
Age	Gender	Income
25	Male	25000
26	Female	22000
30	Male	45000
24	Female	26000

Real Values

Value to Predict

Categorical Values

HOW TO UPLOAD DATASET IN JUPYTER NOTEBOOK



```
In [3]: import pandas as pd
        dataset=pd.read_csv("data.csv")
```

```
In [4]: dataset.head()
```

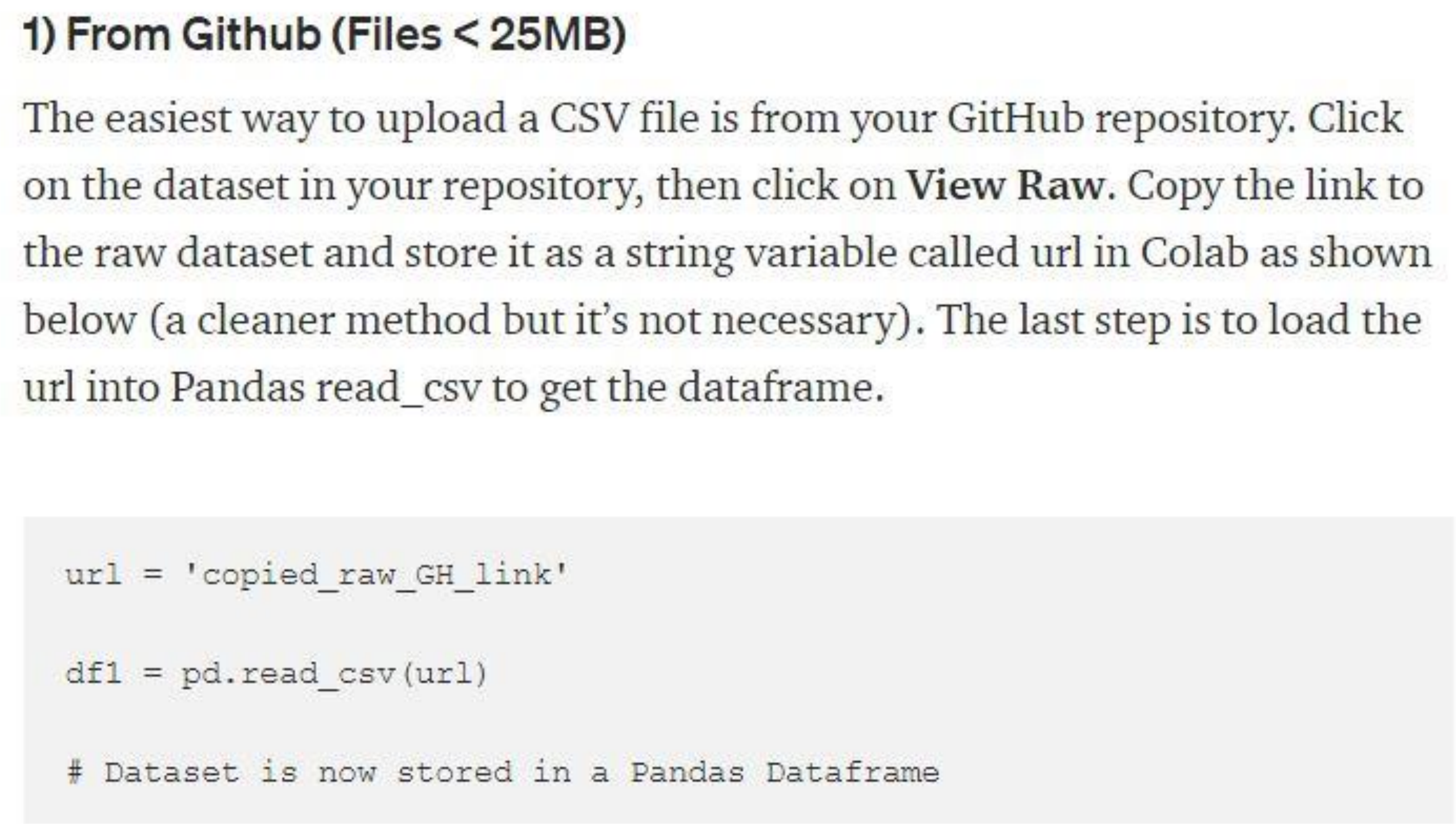
Out[4]:

	Unnamed: 0	Age	Gender	Income	Spending
0	0	22	M	18000	6000
1	1	25	F	30000	8000
2	2	31	F	35000	12000
3	3	35	M	40000	18000

WHAT IS GOOGLE COLAB

- Colab (short for Colaboratory) is a free platform from Google that allows users to code in Python.
- Colab is essentially the Google Suite version of a Jupyter Notebook.
- Some of the advantages of Colab over Jupyter include an easier installation of packages and sharing of documents.
- When loading files like CSV files, it requires some extra coding.
- I will show you three ways to load a CSV file into Colab and insert it into a Pandas dataframe..

WAY TO UPLOAD DATASET IN COLAB



1) From Github (Files < 25MB)

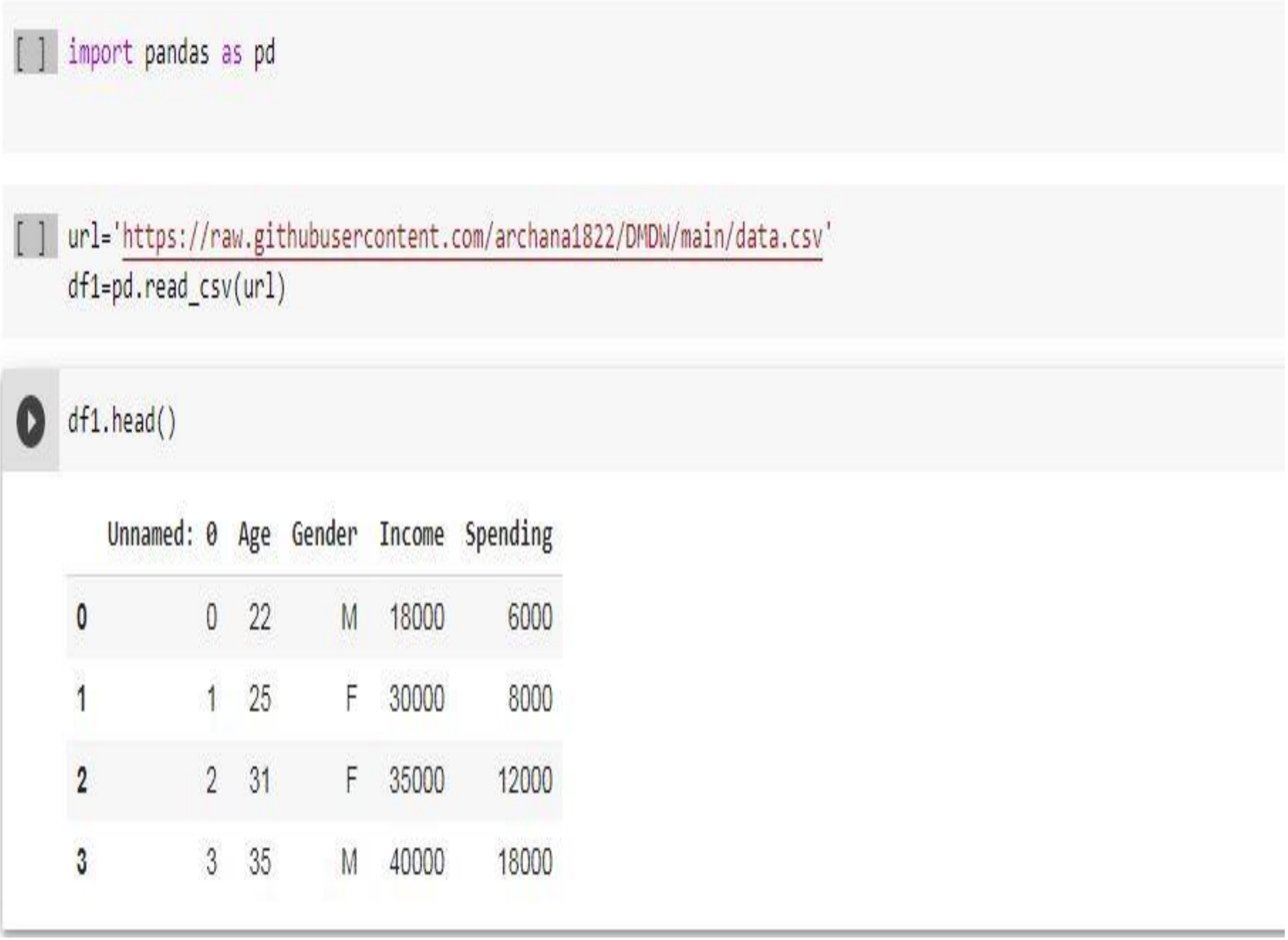
The easiest way to upload a CSV file is from your GitHub repository. Click on the dataset in your repository, then click on **View Raw**. Copy the link to the raw dataset and store it as a string variable called url in Colab as shown below (a cleaner method but it's not necessary). The last step is to load the url into Pandas read_csv to get the dataframe.

```
url = 'copied_raw_GH_link'

df1 = pd.read_csv(url)

# Dataset is now stored in a Pandas Dataframe
```

FROM GITHUB



```
[ ] import pandas as pd
```

```
[ ] url='https://raw.githubusercontent.com/archana1822/DMDW/main/data.csv'
    df1=pd.read_csv(url)
```

```
df1.head()
```

```
   Unnamed: 0  Age  Gender  Income  Spending
0           0   22       M   18000      6000
1           1   25       F   30000      8000
2           2   31       F   35000     12000
3           3   35       M   40000     18000
```

FROM LOCAL DRIVE

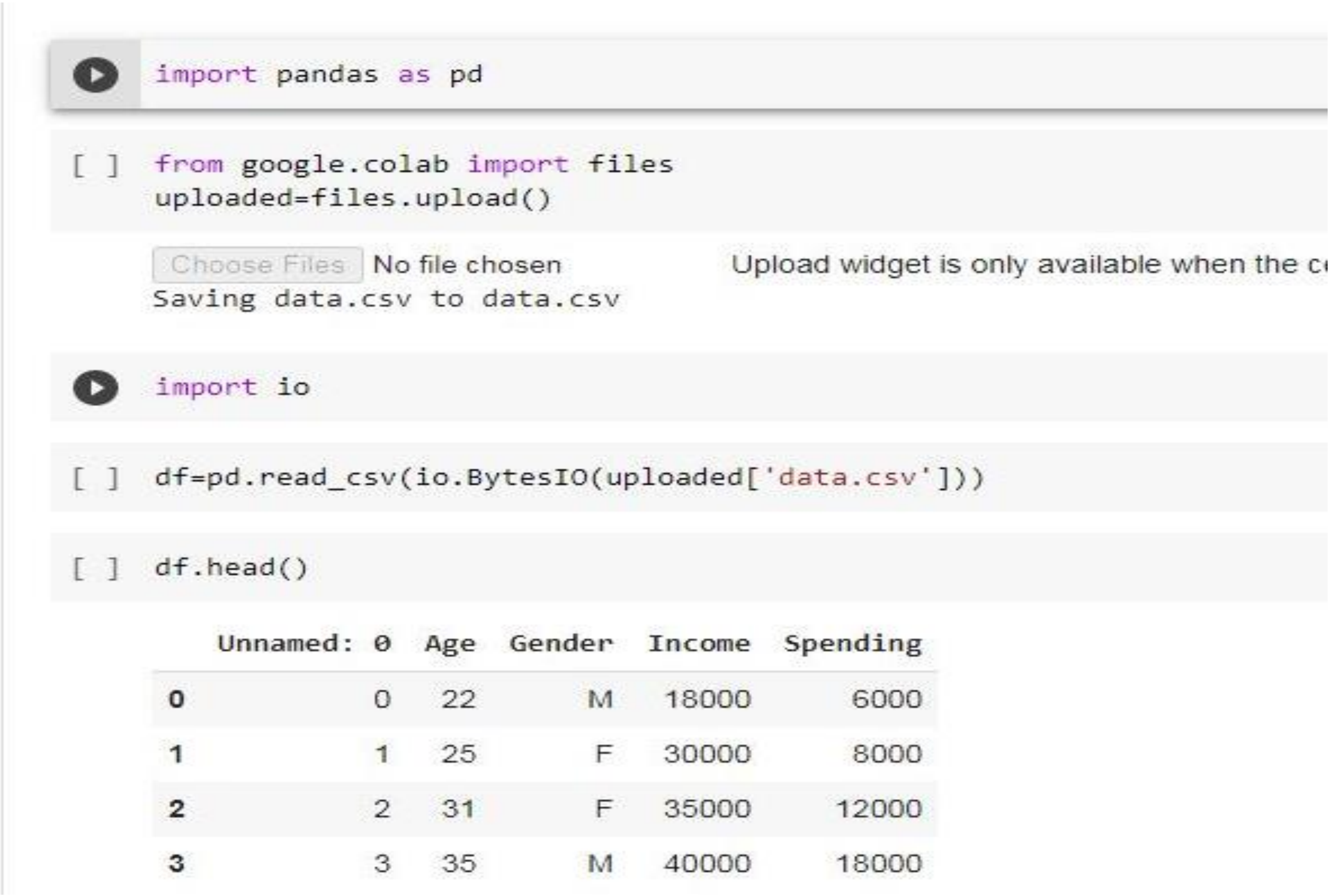
2. From a local drive to upload from your local drive, start with the following code:

```
from google.colab import files
uploaded = files.upload()
```

- It will prompt you to select a file. Click on “Choose Files” then select and upload the file. Wait for the file to be 100% uploaded. You should see the name of the file once Colab has uploaded it.
- Finally, type in the following code to import it into a dataframe

```
import io
df1 = pd.read_csv(io.BytesIO(uploaded['Filename.csv']))
```

FROM LOCAL DRIVE



```
import pandas as pd
```

```
[ ] from google.colab import files
    uploaded=files.upload()
```

```
Saving data.csv to data.csv
```

```
import io
```

```
[ ] df=pd.read_csv(io.BytesIO(uploaded['data.csv']))
```

```
[ ] df.head()
```

```
   Unnamed: 0  Age  Gender  Income  Spending
0           0   22       M   18000      6000
1           1   25       F   30000      8000
2           2   31       F   35000     12000
3           3   35       M   40000     18000
```

NUMPY LIBRARY

- NumPy is a python library used for working with arrays.
- NumPy stands for Numerical Python.
- It also has functions for working in domain of linear algebra, fourier transform, and matrices.
- NumPy was created in 2005 by Travis Oliphant. It is an open source project and you can use it freely.

INSTALLATION OF NUMPY LIBRARY

- Syntax:- pip install numpy

- Use of numpy in program :-
  import numpy as np

EXAMPLE NUMPY LIBRARY

1.
   ```
   import numpy
    arr = numpy.array([1, 2, 3, 4, 5])
    print(arr)
   ```

2.
   ```
   import numpy as np
   arr = np.array([1, 2, 3, 4, 5])
   print(arr)
   ```

3.
   ```
   import numpy as np
   print(np.__version__)
   ```

4.
   ```
   import numpy as np
   arr = np.array([1, 2, 3, 4, 5])
   print(arr)
   print(type(arr))
   ```

EXAMPLE NUMPY LIBRARY

5. # Create 2 new lists height and weight

```
    height = [1.87,  1.87, 1.82, 1.91, 1.90, 1.85]
   weight = [81.65, 97.52, 95.25, 92.98, 86.18, 88.45]
    # Import the numpy package as np
      import numpy as np
  # Create 2 numpy arrays from height and weight
     np_height = np.array(height)
     np_weight = np.array(weight)
```

EXAMPLE NUMPY LIBRARY

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

x = np.arange(1,11)
y = 2 * x + 5
plt.title("Matplotlib demo")
plt.xlabel("x axis caption")
plt.ylabel("y axis caption")
plt.plot(x,y,"ob")
plt.show()
```

EXAMPLE NUMPY LIBRARY

```
7.import numpy as np
a = np.array(42)
b = np.array([1, 2, 3, 4, 5])
c = np.array([[1, 2, 3], [4, 5, 6]])
d = np.array([[[1, 2, 3], [4, 5, 6]], [[1, 2, 3], [4, 5, 6]]])
print(a.ndim)
print(b.ndim)
print(c.ndim)
print(d.ndim)
```

PANDAS LIBRARY

- Pandas is an open source, BSD-licensed library providing high-performance, easy-to-use data structures and data analysis tools for the Python programming language.
- Pandas data analysis and modeling features enable users to carry out their entire data analysis workflow in Python.

INSTALLATION OF PANDAS LIBRARY

- Syntax:- pip install pandas

- Use of numpy in program :-
  import pandas as pd

EXAMPLE OF PANDAS LIBRARY

Ex-1)

```
import pandas as pd
data = [1,2,3,4,5]
df = pd.DataFrame(data)
print(df)
```

EXAMPLE OF PANDAS LIBRARY

Ex-2)

```
import pandas as pd
data = [['Alex',10],['Bob',12],['Clarke',13]]
df = pd.DataFrame(data,columns=['Name','Age'])
print(df)
```

EXAMPLE OF PANDAS LIBRARY

Ex-3)

```
import pandas as pd
d = {'one' : pd.Series([1, 2, 3], index=['a', 'b', 'c']),
   'two' : pd.Series([1, 2, 3, 4], index=['a', 'b', 'c', 'd'])}
df = pd.DataFrame(d)
print ("Adding a new column by passing as Series:")
df['three']=pd.Series([10,20,30],index=['a','b','c'])
print(df)
print ("Adding a new column using the existing columns in Data")
df['four']=df['one']+df['three']
print(df)
```

EXAMPLE OF PANDAS LIBRARY

Ex-4)

```
import pandas as pd
data = [['Alex',10],['Bob',12],['Clarke',13]]
df = pd.DataFrame(data,columns=['Name','Age'],dtype=float)
print(df)
```

EXAMPLE OF PANDAS LIBRARY

Ex-5)

```
import pandas as pd
data = {'Name':['Tom', 'Jack', 'Steve',
   'Ricky'],'Age':[28,34,29,42]}
df = pd.DataFrame(data)
print(df)
```

ASSIGNMENT QUESTION

1. Download a dataset from kaggle and upload it in jupyter notebook or google colab. Perform any random operation on the dataset.
2. Practice 10 python programs using numpy libraries.
3. Practice 10 python programs using pandas libraries.

THANK YOU