EXPERIMENT NO. 1

**PART A**

**A.1 AIM: - To study Pandas and Numpy in Python**

**A.2 Prerequisite**

Python

**A.3 Outcome**

After successful completion of this experiment students will be able to

1. Understand and perform analysis using Numpy and pandas

**A.4 Theory**

NumPy is a Python package. It stands for 'Numerical Python'. It is a library consisting of multidimensional array objects and a collection of routines for processing of array.

Numeric, the ancestor of NumPy, was developed by Jim Hugunin. Another package Numarray was also developed, having some additional functionalities. In 2005, Travis Oliphant created NumPy package by incorporating the features of Numarray into Numeric package. There are many contributors to this open source project.

Operations using NumPy

Using NumPy, a developer can perform the following operations −

Mathematical and logical operations on arrays.

Fourier transforms and routines for shape manipulation.

Operations related to linear algebra. NumPy has in-built functions for linear algebra and random number generation.

NumPy – A Replacement for MatLab

NumPy is often used along with packages like SciPy (Scientific Python) and Mat−plotlib (plotting library). This combination is widely used as a replacement for MatLab, a popular platform for technical computing. It is open source, which is an added advantage of NumPy.

pip install numpy

## **Example 1**

import numpy as np

a = np.array([1,2,3])

print a

The output is as follows −

[1, 2, 3]

## **Example 2**

# more than one dimensions

import numpy as np

a = np.array([[1, 2], [3, 4]])

print a

The output is as follows −

[[1, 2]

[3, 4]]

## **Example 3**

# minimum dimensions

import numpy as np

a = np.array([1, 2, 3,4,5], ndmin = 2)

print a

The output is as follows −

[[1, 2, 3, 4, 5]]

## **Example 4**

# dtype parameter

import numpy as np

a = np.array([1, 2, 3], dtype = complex)

print a

The output is as follows −

[ 1.+0.j, 2.+0.j, 3.+0.j]

## **ndarray.shape**

This array attribute returns a tuple consisting of array dimensions. It can also be used to resize the array.

### **Example 1**

### import numpy as np

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

print a.shape

The output is as follows −

(2, 3)

### **Example 2**

# this resizes the ndarray

import numpy as np

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

a.shape = (3,2)

print a

The output is as follows −

[[1, 2]

[3, 4]

[5, 6]]

### **Example 3**

NumPy also provides a reshape function to resize an array.

import numpy as np

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

b = a.reshape(3,2)

print b

The output is as follows −

[[1, 2]

[3, 4]

[5, 6]]

## **ndarray.ndim**

This array attribute returns the number of array dimensions.

### **Example 1**

# an array of evenly spaced numbers

import numpy as np

a = np.arange(24)

print a

The output is as follows −

[0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23]

### **Example 2**

# this is one dimensional array

import numpy as np

a = np.arange(24)

a.ndim

# now reshape it

b = a.reshape(2,4,3)

print b

# b is having three dimensions

The output is as follows −

[[[ 0, 1, 2]

[ 3, 4, 5]

[ 6, 7, 8]

[ 9, 10, 11]]

[[12, 13, 14]

[15, 16, 17]

[18, 19, 20]

[21, 22, 23]]]

## **numpy.itemsize**

This array attribute returns the length of each element of array in bytes.

### **Example 1**

# dtype of array is int8 (1 byte)

import numpy as np

x = np.array([1,2,3,4,5], dtype = np.int8)

print x.itemsize

The output is as follows −

1

### **Example 2**

# dtype of array is now float32 (4 bytes)

import numpy as np

x = np.array([1,2,3,4,5], dtype = np.float32)

print x.itemsize

The output is as follows −

4

**np.zeros**

### **Example 1**

# array of five zeros. Default dtype is float

import numpy as np

x = np.zeros(5)

print x

The output is as follows −

[ 0. 0. 0. 0. 0.]

**np.ones**

### **Example 1**

# array of five ones. Default dtype is float

import numpy as np

x = np.ones(5)

print x

The output is as follows −

[ 1. 1. 1. 1. 1.]

**PANDAS**

Pandas is an open-source Python Library providing high-performance data manipulation and analysis tool using its powerful data structures. The name Pandas is derived from the word Panel Data – an Econometrics from Multidimensional data.

In 2008, developer Wes McKinney started developing pandas when in need of high performance, flexible tool for analysis of data.

Prior to Pandas, Python was majorly used for data munging and preparation. It had very little contribution towards data analysis. Pandas solved this problem. Using Pandas, we can accomplish five typical steps in the processing and analysis of data, regardless of the origin of data — load, prepare, manipulate, model, and analyze.

Python with Pandas is used in a wide range of fields including academic and commercial domains including finance, economics, Statistics, analytics, etc.

**Key Features of Pandas**

Fast and efficient DataFrame object with default and customized indexing.

Tools for loading data into in-memory data objects from different file formats.

Data alignment and integrated handling of missing data.

Reshaping and pivoting of date sets.

Label-based slicing, indexing and subsetting of large data sets.

Columns from a data structure can be deleted or inserted.

Group by data for aggregation and transformations.

High performance merging and joining of data.

Time Series functionality.

pip install pandas

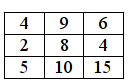
## **Dimension & Description**

The best way to think of these data structures is that the higher dimensional data structure is a container of its lower dimensional data structure. For example, DataFrame is a container of Series, Panel is a container of DataFrame.

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Dimensions** | **Description** |
| Series | 1 | 1D labeled homogeneous array, sizeimmutable. |
| Data Frames | 2 | General 2D labeled, size-mutable tabular structure with potentially heterogeneously typed columns. |
| Panel | 3 | General 3D labeled, size-mutable array. |

Refer the notes for examples

Find the appropriate answers :

1. Package that deals with dataframe is
2. numpy
3. dataframe
4. pandas
5. math
6. Create the following Matrix “Y” in Python and answer following questions   
   
7. Find the determinant of the matrix “Y” rounded off to the zeroth decimal place.
8. Find the inverse of matrix “Y” rounded off to the second place
9. Find the column sum of YT

Tasks:

Consider the link for the data set ‘mtcars.csv’ :

<https://drive.google.com/file/d/1Ua21bZfbtN4DUw4fK9XCF3AJmcIqSn4w/view>

1. From the given dataset ‘mtcars.csv’, plot a histogram to check the frequency distribution of the variable ‘mpg’ (Miles per gallon). The interval having the highest frequency is: -
2. 10 to 15
3. 15 to 20
4. 25 to 30
5. 30 to 35
6. Which of the following can be inferred from scatter plot of ‘mpg’ (Miles per gallon) vs ‘wt’ (Weight of car) from the dataset mtcars.csv?
7. As weight of the car increases, the mpg decreases
8. As weight of the car increases, the mpg increases
9. There is no ration between weight of the car and mpg
10. When weight increases, mpg increases exponentially
11. Read the given data “churn.csv” and save it as a dataframe called churn, and answer the questions below

<https://drive.google.com/file/d/1JSYGIIkIZr4jyheDEH0X1_TMSnQ2CzXc/view>

The no. of duplicate records in the churn dataframe based on the cutomerID column are

1. 7
2. 8
3. 22
4. No duplicates
5. In the churn dataframe, what are the total no. of missing values for the variable TotalCharges?
6. 10
7. 23
8. 15
9. 5
10. From the churn dataframe, what is the average monthly charge paid by a customer for the services he/she has signed up for?
11. 72.56
12. 62.47
13. 68.86
14. 78.58
15. In the churn dataframe, under the variable Dependents how many records have “1@#" ?
16. 9
17. 6
18. 2
19. 8
20. The data type of the variable tenure from the churn dataframe is
21. int64
22. float64
23. long
24. object
25. The command to replace ‘Four’ by 4 and ‘One’ by 1 under the variable “tenure” of the churn dataframe: -
26. pandas.Dataframe.where()
27. pandas.Dataframe.replace()
28. numpy.where()
29. all the above

**PART B**

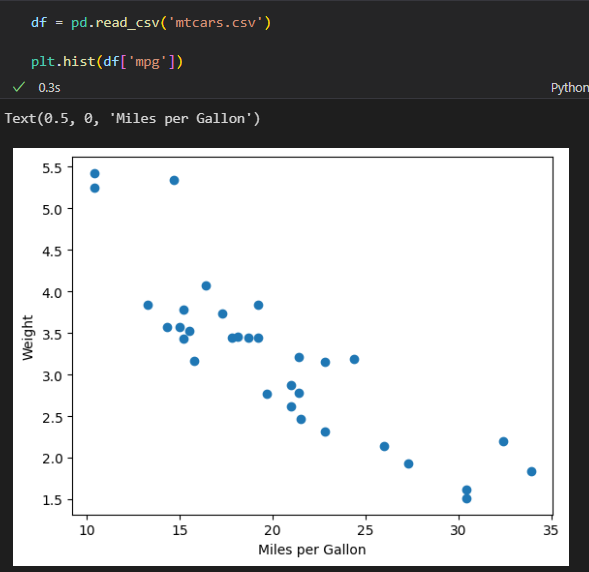
(PART B: TO BE COMPLETED BY STUDENTS)

**(Students must submit the soft copy as per following segments within two hours of the practical. The soft copy must be uploaded on the Blackboard or emailed to the concerned lab in charge faculties at the end of the practical in case the there is no Black board access available)**

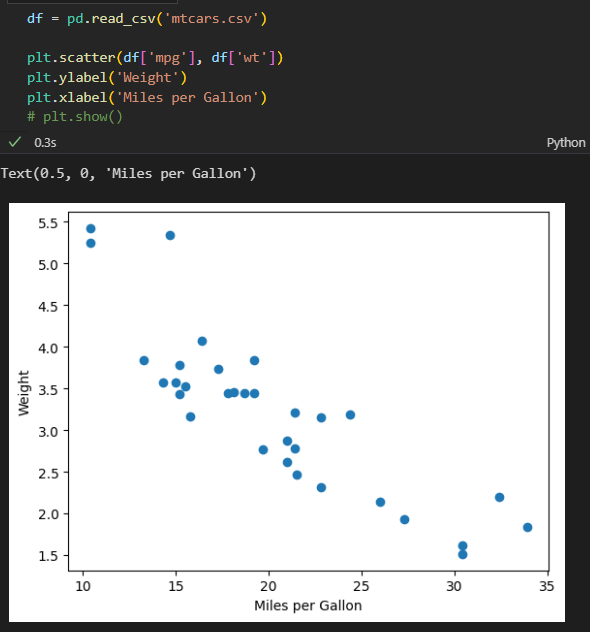
|  |  |
| --- | --- |
| Roll No.: C026 | Name: Anirbaan Ghatak |
| Program: BTI | Division: B |
| Semester: B1 | Batch: B1 |
| Date of Experiment: | Date of Submission: |
| Grade : |  |

B.1 Software Code written by student:

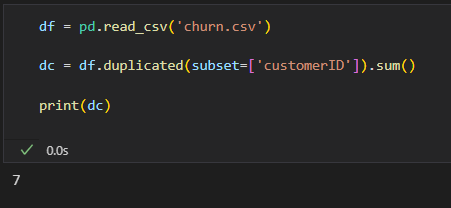
**1**



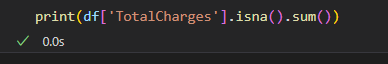
**2**



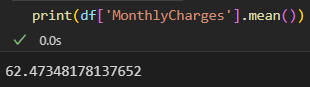
**3**



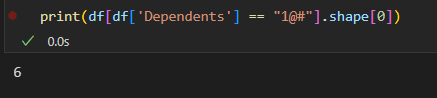
4.



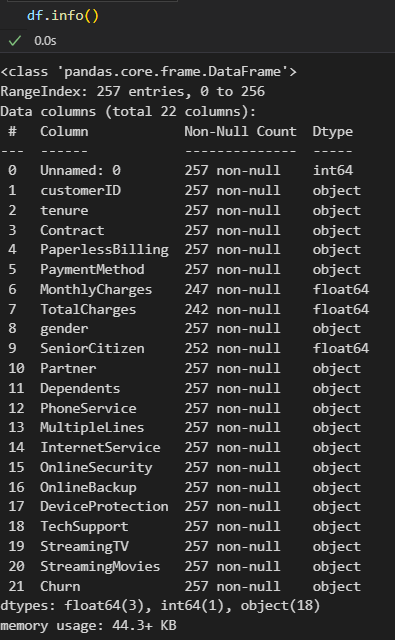
5.



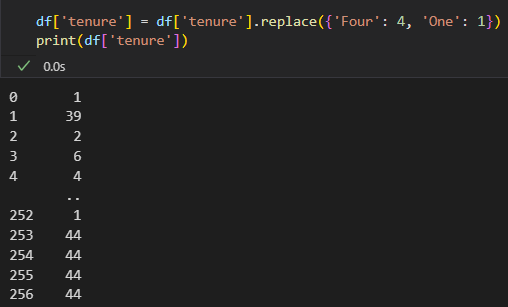
6.



7.



8.



B.2 Input and Output:

***(Paste your program input and output in following format. If there is error then paste the specific error in the output part. In case of error with due permission of the faculty extension can be given to submit the error free code with output in due course of time. Students will be graded accordingly.)***

B.3 Conclusion:

*Learned the basics of using pandas and matplotlib to manipulate data.*