**Name :- Shivam Dilipsingh Bhadoriya**

**Course:- BSC – IT**

**Python Assignmenmt :- Numpy and Pandas**

**Aim:-** **Create a 1D NumPy array with 10 integers. Print the array and its data type.**

**Code:-**

# Create a 1D NumPy array with 10 integers. Print the array and its data type.

import numpy as np

list = [1,2,3,4,5,6,7,8,9,10]

print(list)

print(type(list))

**Output:-**



**Aim:- Generate an array of 10 random integers between 1 and 100 using NumPy.**

**Code:-**

# Generate an array of 10 random integers between 1 and 100 using NumPy.

import numpy as np

x = np.random.randint(1,101, size=10)

print(x)

**Output:-**



**Aim:- Calculate the mean, median, mode, standard deviation, and variance for the array: [12,**

**15, 12, 18, 21, 24, 24, 24, 27, 30]**

**Code:-**

# Calculate the mean, median, mode, standard deviation, and variance for the array:

# [12, 15, 12, 18, 21, 24, 24, 24, 27, 30]

import numpy as np

import scipy

import statistics as stats

list = [12, 15, 12, 18, 21, 24, 24, 24, 27, 30]

print(list)

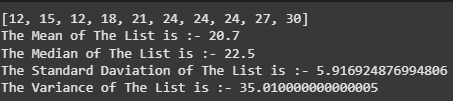
print(f"The Mean of The List is :- {np.mean(list)}")

print(f"The Median of The List is :- {np.median(list)}")

print(f"The Standard Daviation of The List is :- {np.std(list)}")

print(f"The Standard Daviation of The List is :- {np.var(list)}")

**Output:-**



**Aim:-** **Create a list of integers from 1 to 10. Convert it into a NumPy array and apply all five**

**statistical functions.**

**Code:-**

# Create a list of integers from 1 to 10. Convert it into a NumPy array and apply all five

# statistical functions.

import numpy as np

import scipy

import statistics as stats

list = [1,2,3,4,5,6,7,8,9,10]

print(list)

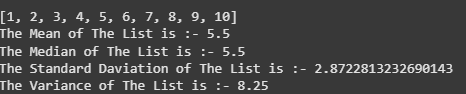
print(f"The Mean of The List is :- {np.mean(list)}")

print(f"The Median of The List is :- {np.median(list)}")

print(f"The Standard Daviation of The List is :- {np.std(list)}")

print(f"The Variance of The List is :- {np.var(list)}")

**Output:-**



**Aim:-** **Take a list of marks scored by 5 students in 3 subjects. Convert it into a 2D NumPy array**

**and:**

**- Calculate the average marks of each student.**

**- Find the subject with the highest average.**

**Code:-**

import numpy as np

Marks = np.array([[45,35,64],[34,64,36],[43,65,74],[46,46,74],[37,54,32]])

x = np.mean(Marks,axis=1)

y = np.mean(Marks,axis=0)

print("Average marks of each student: ",x)

print("subject with the highest average: ",np.max(y))

**Output:-**



**Aim:- Print the first and last rows of the DataFrame using .head() and .tail().**

**Code:-**

# Print the first and last rows of the DataFrame using .head() and .tail().

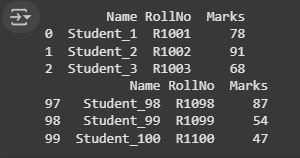
import pandas as pd

df = pd.read\_csv("data.csv")

print(df.head(3))

print(df.tail(3))

**Output:-**



**Aim:-** **Create a DataFrame using the dictionary: {&#39;Product&#39;: [&#39;Pen&#39;, &#39;Notebook&#39;, &#39;Eraser&#39;], &#39;Price&#39;:**

**[10, 25, 5], &#39;Stock&#39;: [100, 50, 200]}. Print the entire DataFrame.**

**Code:-**

# Create a DataFrame using the dictionary: {Product;: [Pen, Notebook, Eraser],Price:

# [10, 25, 5], Stock: [100, 50, 200]}. Print the entire DataFrame..

import pandas as pd

data = {'Product': ['Pen', 'Notebook', 'Eraser'],

        'Price': [10, 25, 5],

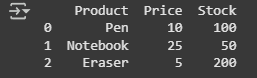
        'Stock': [100, 50, 200]

        }

df = pd.DataFrame(data)

print(df)

**Output:-**



**Aim:-** **Add a new column Discounted\_Price = Price - 10%.**

**Code:-**

# Create a DataFrame using the dictionary: {Product;: [Pen, Notebook, Eraser],Price:

# [10, 25, 5], Stock: [100, 50, 200]}. Print the entire DataFrame..

import pandas as pd

data = {'Product': ['Pen', 'Notebook', 'Eraser'],

        'Price': [10, 25, 5],

        'Stock': [100, 50, 200]

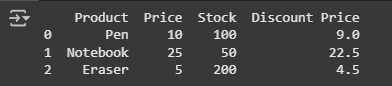
        }

df = pd.DataFrame(data)

df['Discount Price'] = df.Price \* 0.9

print(df)

**Output:-**



**Aim:- Create a sample data.csv file with student details (Name, RollNo, Marks) and:**

**- Read it using pd.read\_csv().**

**- Print the total number of students.**

**Code:-**

# Create a sample data.csv file with student details (Name, RollNo, Marks) and:

# - Read it using pd.read\_csv().

# - Print the total number of students.

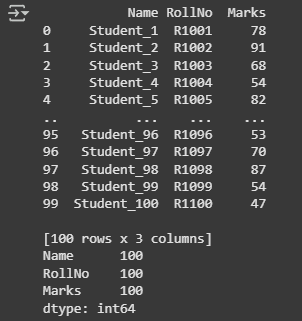
import pandas as pd

df = pd.read\_csv("data.csv")

print(df)

print(df.count())

**Output:-**



**Aim:- Load a CSV file and display only the names of students who scored more than 75 marks.**

**Code:-**

from os import sysconf\_names

# Load a CSV file and display only the names of students who scored more than 75 marks.

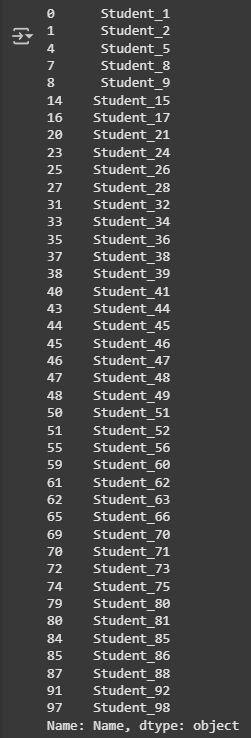
import pandas as pd

df = pd.read\_csv("data.csv")

s\_names = df[df["Marks"]>75]["Name"]

print(s\_names)

**Output:-**



**Aim:- Given a DataFrame of employees with Name, Dept, and Salary:**

**- Retrieve the first 3 rows.**

**- Retrieve only the Name and Salary columns.**

**- Display the salaries of employees in the HR department.**

**Code:-**

from os import sysconf\_names

# Load a CSV file and display only the names of students who scored more than 75 marks.

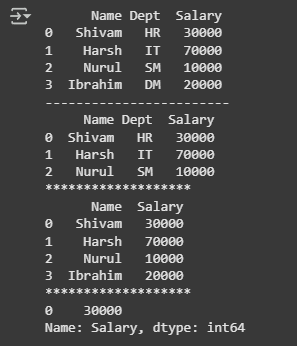
import pandas as pd

df = pd.read\_csv("data.csv")

s\_names = df[df["Marks"]>75]["Name"]

print(s\_names)

**Output:-**



**Aim:- Access the value at row index 2 and column index 1 using iloc.**

**Code:-**

# Access the value at row index 2 and column index 1 using iloc.

import numpy as np

emp =  { "Name" : ["Shivam","Nurul", "Jarir","Pratham","Harsh"] , "Dept":["IT", "HR","Sales","HR","Finance"], "Salary": [80000, 55000, 62000,64000,45000]}

df = pd.DataFrame(emp)

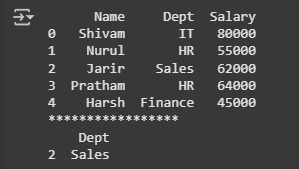
print(df)

print("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")

row = df.iloc[[2],[1]]

print(row)

**Output:-**



**Aim:- Use loc to get the rows where Salary &gt; 50000.**

**Code:-**

# Use loc to get the rows where Salary &gt; 50000.

import pandas as pd

emp =  { "Name" : ["Shivam","Nurul", "Jarir","Pratham","Abhii"] , "Dept":["IT", "HR","Sales","HR","Finance"], "Salary": [80000, 45000, 72000,64000,45000]}

df = pd.DataFrame(emp)

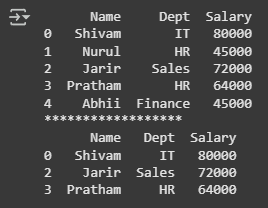
print(df)

print("--------------------")

row = df[df["Salary"]>50000]

print(row)

**Output:-**



**Aim:-** **Create a DataFrame with at least 5 rows and 3 columns.**

**- Retrieve the third row using iloc.**

**- Retrieve the values from the second and fourth rows, and the first two columns using**

**iloc.**

**Code:-**

# Create a DataFrame with at least 5 rows and 3 columns.

# - Retrieve the third row using iloc.

# - Retrieve the values from the second and fourth rows, and the first two columns using

# iloc.

emp =  { "Name" : ["Shivam","Nurul", "Jarir","Pratham","Abhii"] , "Dept":["IT", "HR","Sales","HR","Finance"], "Salary": [80000, 79999, 72000,64000,45000]}

df = pd.DataFrame(emp)

print(df)

print("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")

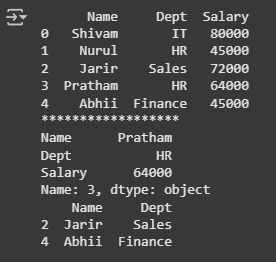
row = df.iloc[3,:]

print(row)

row = df.iloc[[2,4],[0,1]]

print(row)

**Output:-**



**Aim:-** **Using loc, display:**

**- All columns for the row with index 0.**

**- A slice of rows and selected columns using labels.**

**Code:-**

# Using loc, display:

# - All columns for the row with index 0.

# - A slice of rows and selected columns using labels.

emp =  { "Name" : ["Shivam","Nurul", "Jarir","Pratham","Abhii"] , "Dept":["IT", "HR","Sales","HR","Finance"], "Salary": [80000, 72000, 72000, 64000, 45000]}

df = pd.DataFrame(emp)

print(df)

print("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")

row = df.loc[:0]

print(row)

print("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")

col= df.loc[0:3,["Name","Dept"]]

print(col)

**Output:-**

