Practical No. 2

Title:Introduction to Python Programming and python libraries

Aim: Understanding basics of python programming and python libraries like numpy, pandas and matplotlib

Introduction:

What is Python :-

Python is a popular programming language. It was created by Guido van Rossum, and released in 1991. It was designed with an emphasis on code readability, and its syntax allows programmers to express their concepts in fewer lines of code.

Python is a programming language that lets you work quickly and integrate systems more efficiently.

Python libraries for Machine Learning

Machine Learning, as the name suggests, is the science of programming a computer by which they are able to learn from different kinds of data. A more general definition given by Arthur Samuel is – "Machine Learning is the field of study that gives computers the ability to learn without being explicitly programmed." They are typically used to solve various types of life problems.

In the older days, people used to perform Machine Learning tasks by manually coding all the algorithms and mathematical and statistical formulas. This made the processing time consuming, tedious and inefficient. But in the modern days, it has become very much easy and efficient compared to the olden days with various python libraries, frameworks, and modules. Today, Python is one of the most popular programming languages for this task and it has replaced many languages in the industry, one of the reasons is its vast collection of libraries. Python libraries that used in Machine Learning are:

Numpy

NumPy is a very popular python library for large multi-dimensional array and matrix processing, with the help of a large collection of high-level mathematical functions. It is very useful for fundamental scientific computations in Machine Learning. It is particularly useful for linear algebra, Fourier transform, and random number capabilities. High-end libraries like TensorFlow use NumPy internally for manipulation of Tensors.

Pandas

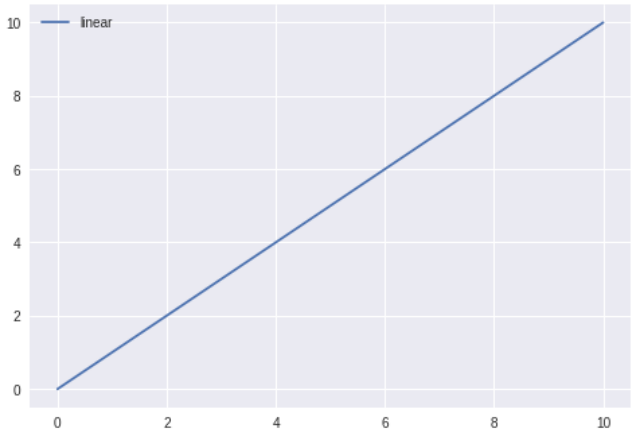
Pandas is a popular Python library for data analysis. It is not directly related to Machine Learning. As we know that the dataset must be prepared before training. In this case, Pandas comes handy as it was developed specifically for data extraction and preparation. It provides high-level data structures and a wide variety of tools for data analysis. It provides many inbuilt methods for grouping, combining and filtering data.

Matplotlib

Matplotlib is a very popular Python library for data visualization. Like Pandas, it is not directly related to Machine Learning. It particularly comes in handy when a programmer wants to visualize the patterns in the data. It is a 2D plotting library used for creating 2D graphs and plots. A module named pyplot makes it easy for programmers for plotting as it provides features to control line styles, font properties, formatting axes, etc. It provides various kinds of graphs and plots for data visualization, viz., histogram, error charts, bar charts, etc,

Example : -

```
import matplotlib.pyplot as plt

import numpy as np

x = np.linspace(0, 10, 100)

plt.plot(x, x, label ='linear')

plt.legend()

plt.show()
```



Exercise -

1. **Create Pandas Dataframe to enter employee database which includes Sr No, Name, Mobile No, City**
2. **Use Iris Dataset from Github and perform all basic operations on Iris Dataset**
3. **Use matplotlib to plot bar chart using dictionary**
4. **Use matplotlib to scatter graph and histogram**

Implementation:
Program:

```
#Understanding basics of python programming
a1=True
type(a1)
```

```
bool
```

```
print("Hello this is first python prgm")
```

```
Hello this is first python prgm
```

```
student="Amey"
student
```

```
'Amey'
```

```
a1=10
a2=20
a1*a2
```

```
200
```

```
str3='''
amey is
MCA
student
from batch 2025
'''
str3
```

```
'\namey is\nMCA\nstudent\nfrom batch 2025\n'
```

```
student="Akshay"
len(student)
```

```
6
```

```
student[3:]
```

```
'hay'
```

```
student="Akshay"
student.replace('A','m')
```

```
'mkshay'
```

```
student="Akshay Akshay Akshay Akshay"
student.count('Akshay')
```

```
4
```

```
#Python Data Structures Tupple
tup1=(1,"b",True,2.2)
tup1
```

```
(1, 'b', True, 2.2)
```

```
tup1[:3]
```

```
(1, 'b', True)
```

```
len(tup1)
```

```
4
```

```
#concat operation
tup2=(3,"c",False,4)
tup1+tup2
```

```
(1, 'b', True, 2.2, 3, 'c', False, 4)
```

```
tup1*2+tup2
```

```
(1, 'b', True, 2.2, 1, 'b', True, 2.2, 3, 'c', False, 4)
```

```
tup2=(3,"c",False,4,3)
tup2
```

```
(3, 'c', False, 4, 3)
```

```
#Python Data Structures : List
list1=[1,2,True,2.2]
list1
```

```
[1, 2, True, 2.2]
```

```
list1[0]=2
list1
```

```
[2, 2, True, 2.2]
```

```
list1[:3]
```

```
[2, 2, True]
```

```
list1.append("amey")
list1
```

```
[2, 2, True, 2.2, 'amey']
```

```
list1.pop()
list1
```

```
[2, 2, True, 2.2]
```

```
list1.insert(3,"three")
list1
```

```
[2, 2, True, 'three', 2.2]
```

```
def msg():
  print("this is function")
```

```
msg()
```

```
this is function
```

```
def oddeven(X):
  if X%2==0:
    print("this is even number")
  else:
    print("this is odd number")
```

```
oddeven(100)
```

```
this is even number
```

```
g=lambda x:x*x*x

g(2)
```

```
8
```

```
#numpy library
#numpy array

import numpy as np
n1=np.array([1,2,3])
n1
```

```
array([1, 2, 3])
```

```
n2=np.array([[1,2,3],[4,5,6],[6,7,8]])
n2
```

```
array([[1, 2, 3],
       [4, 5, 6],
       [6, 7, 8]])
```

```
n2.shape
```

```
(3, 3)
```

```
n3=np.zeros((5,5))
n3
```

```
array([[0., 0., 0., 0., 0.],
       [0., 0., 0., 0., 0.],
       [0., 0., 0., 0., 0.],
       [0., 0., 0., 0., 0.],
       [0., 0., 0., 0., 0.]])
```

```
n3=np.full((4,4),4)
n3
```

```
array([[4, 4, 4, 4],
       [4, 4, 4, 4],
       [4, 4, 4, 4],
       [4, 4, 4, 4]])
```

```
n3=np.arange(10,200)
n3
```

```
array([ 10,  11,  12,  13,  14,  15,  16,  17,  18,  19,  20,  21,  22,
        23,  24,  25,  26,  27,  28,  29,  30,  31,  32,  33,  34,  35,
        36,  37,  38,  39,  40,  41,  42,  43,  44,  45,  46,  47,  48,
        49,  50,  51,  52,  53,  54,  55,  56,  57,  58,  59,  60,  61,
        62,  63,  64,  65,  66,  67,  68,  69,  70,  71,  72,  73,  74,
        75,  76,  77,  78,  79,  80,  81,  82,  83,  84,  85,  86,  87,
        88,  89,  90,  91,  92,  93,  94,  95,  96,  97,  98,  99, 100,
       101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113,
       114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126,
       127, 128, 129, 130, 131, 132, 133, 134, 135, 136, 137, 138, 139,
       140, 141, 142, 143, 144, 145, 146, 147, 148, 149, 150, 151, 152,
       153, 154, 155, 156, 157, 158, 159, 160, 161, 162, 163, 164, 165,
       166, 167, 168, 169, 170, 171, 172, 173, 174, 175, 176, 177, 178,
       179, 180, 181, 182, 183, 184, 185, 186, 187, 188, 189, 190, 191,
       192, 193, 194, 195, 196, 197, 198, 199])
```

```
n4=np.array([[1,2,3],[4,5,6]])
n4
```

```
array([[1, 2, 3],
       [4, 5, 6]])
```

```
n4.shape=(3,2)
n4
```

```
array([[1, 2],
       [3, 4],
```

```
       [5, 6]])
```

```
n1=np.array([1,2,3])
n2=np.array([4,5,6])
np.vstack((n1,n2))
```

```
array([[1, 2, 3],
       [4, 5, 6]])
```

```
np.hstack((n1,n2))
```

```
array([1, 2, 3, 4, 5, 6])
```

```
np.column_stack((n1,n2))
```

```
array([[1, 4],
       [2, 5],
       [3, 6]])
```

```
n1=np.array([1,2,3,4,5])
n2=np.array([4,5,6,7,8])
np.intersect1d(n1,n2)
```

```
array([4, 5])
```

```
np.setdiff1d(n2,n1)
```

```
array([6, 7, 8])
```

```
import numpy as np

n1=np.array([1,2,3])
n2=np.array([4,5,4])
np.sum([n1,n2])
```

```
np.int64(19)
```

```
n1=np.array([1,2,3])
np.save('myarr',n1)

n2=np.load('myarr.npy')
n2
```

```
array([1, 2, 3])
```

```
#Python Pandas lib #series Object

import pandas as pd
s1=pd.Series([1,2,3,4])
s1
```

	0
0	1
1	2
2	3
3	4

dtype: int64

```
s1=pd.Series([1,2,3,4],index=['a','b','c','d'])
s1
```

	0
a	1
b	2
c	3
d	4

dtype: int64

```
s2=pd.Series({'a':1,'b':2,'c':3})
s2
```

	0
a	1
b	2
c	3

dtype: int64

```
#Pandas DataFrame
pd.DataFrame({'Name':['Amey','Akshay','Anil'],
              'Marks':[80,90,95],
              'MobileNo':[7972221200,7972221201,7972221202]})
```

	Name	Marks	MobileNo
0	Amey	80	7972221200
1	Akshay	90	7972221201
2	Anil	95	7972221202

```
url='https://gist.githubusercontent.com/curran/a08a1080b88344b0c8a7/raw/0e7a9b0a5d22642a06d3d5b9
iris=pd.read_csv(url)

iris.head()
```

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3.0	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
3	4.6	3.1	1.5	0.2	setosa
4	5.0	3.6	1.4	0.2	setosa

```
iris.tail()
```

	sepal_length	sepal_width	petal_length	petal_width	species
145	6.7	3.0	5.2	2.3	virginica
146	6.3	2.5	5.0	1.9	virginica
147	6.5	3.0	5.2	2.0	virginica
148	6.2	3.4	5.4	2.3	virginica
149	5.9	3.0	5.1	1.8	virginica

```
iris.shape
```

```
(150, 5)
```

```
iris.describe()
```

	sepal_length	sepal_width	petal_length	petal_width
count	150.000000	150.000000	150.000000	150.000000
mean	5.843333	3.054000	3.758667	1.198667
std	0.828066	0.433594	1.764420	0.763161
min	4.300000	2.000000	1.000000	0.100000
25%	5.100000	2.800000	1.600000	0.300000
50%	5.800000	3.000000	4.350000	1.300000
75%	6.400000	3.300000	5.100000	1.800000
max	7.900000	4.400000	6.900000	2.500000

```
iris.loc[75:77,"petal_length"]
```

	petal_length
75	4.4
76	4.8
77	5.0

dtype: float64

```
iris.drop([0,1,2],axis=0)
```

	sepal_length	sepal_width	petal_length	petal_width	species
3	4.6	3.1	1.5	0.2	setosa
4	5.0	3.6	1.4	0.2	setosa
5	5.4	3.9	1.7	0.4	setosa
6	4.6	3.4	1.4	0.3	setosa
7	5.0	3.4	1.5	0.2	setosa
...	...	...	...	...	...
145	6.7	3.0	5.2	2.3	virginica
146	6.3	2.5	5.0	1.9	virginica
147	6.5	3.0	5.2	2.0	virginica
148	6.2	3.4	5.4	2.3	virginica
149	5.9	3.0	5.1	1.8	virginica

147 rows × 5 columns

```
iris.head()
```

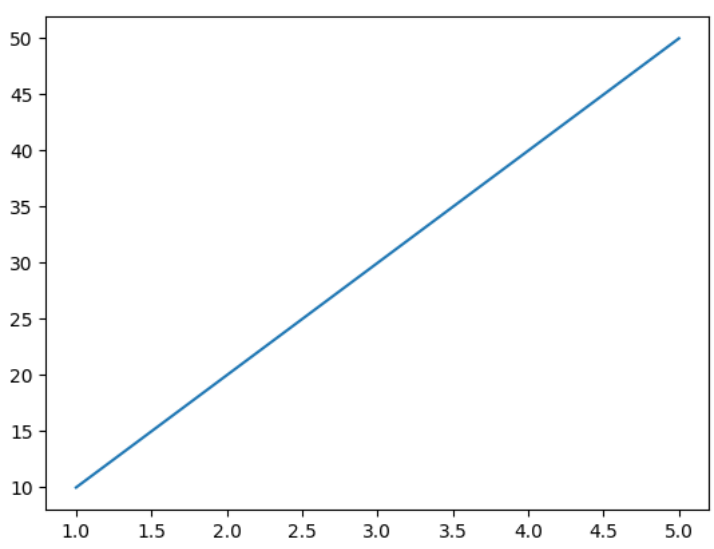
	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3.0	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
3	4.6	3.1	1.5	0.2	setosa
4	5.0	3.6	1.4	0.2	setosa

```
iris.sort_values(by='sepal_length')
```

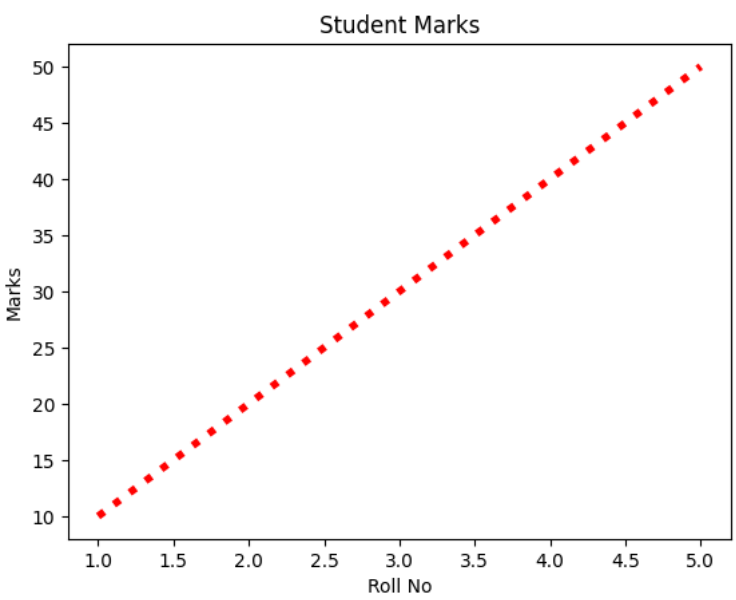
	sepal_length	sepal_width	petal_length	petal_width	species
13	4.3	3.0	1.1	0.1	setosa
8	4.4	2.9	1.4	0.2	setosa
42	4.4	3.2	1.3	0.2	setosa
38	4.4	3.0	1.3	0.2	setosa
41	4.5	2.3	1.3	0.3	setosa
...	...	...	...	...	...
122	7.7	2.8	6.7	2.0	virginica
117	7.7	3.8	6.7	2.2	virginica
118	7.7	2.6	6.9	2.3	virginica
135	7.7	3.0	6.1	2.3	virginica
131	7.9	3.8	6.4	2.0	virginica

150 rows × 5 columns

```
#python library matplotlib

import numpy as np
from matplotlib import pyplot as plt
x=np.array([1,2,3,4,5])
y=np.array([10,20,30,40,50])

plt.plot(x,y)
plt.show()
```

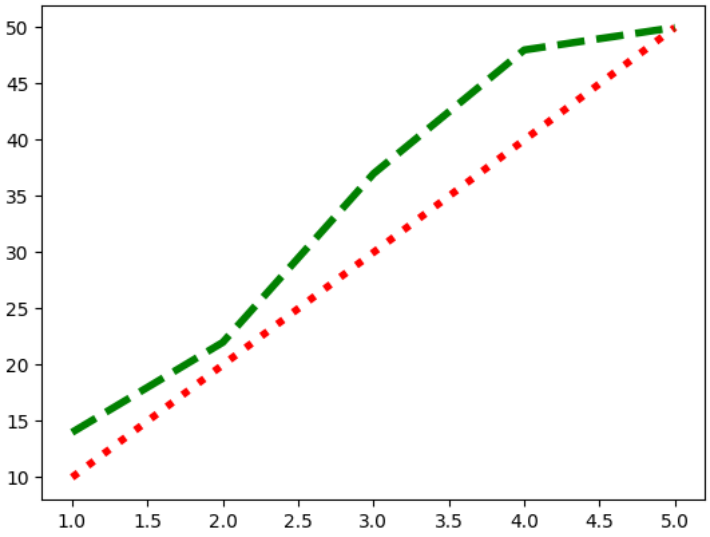



```
plt.plot(x,y,color="r",linestyle=":",linewidth=4)
plt.title("Student Marks")
plt.xlabel("Roll No")
plt.ylabel("Marks")
plt.show()
```

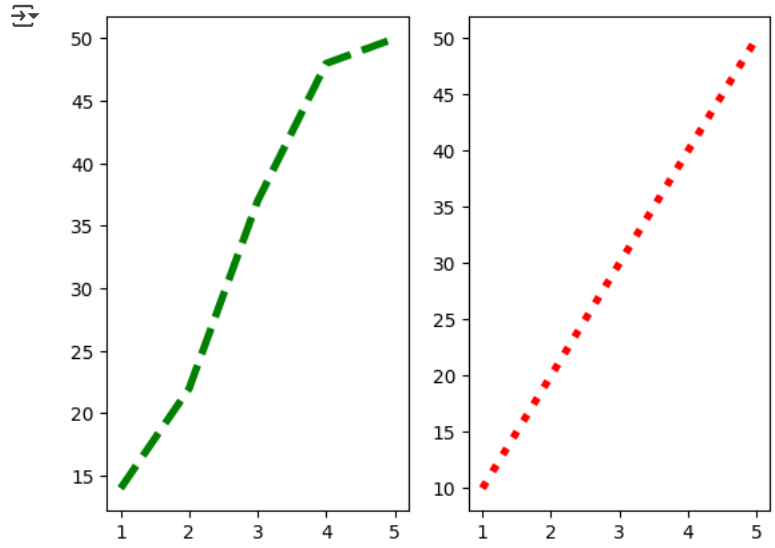



```
y2=np.array([14,22,37,48,50])
plt.plot(x,y,color="r",linestyle=":",linewidth=4)
plt.plot(x,y2,color="g",linestyle="--",linewidth=4)
```

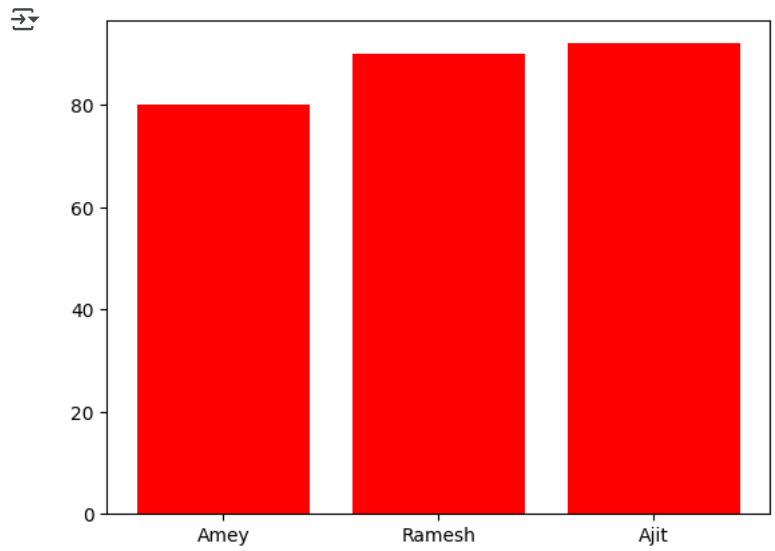
```
[<matplotlib.lines.Line2D at 0x7f7e42ef9d10>]
```



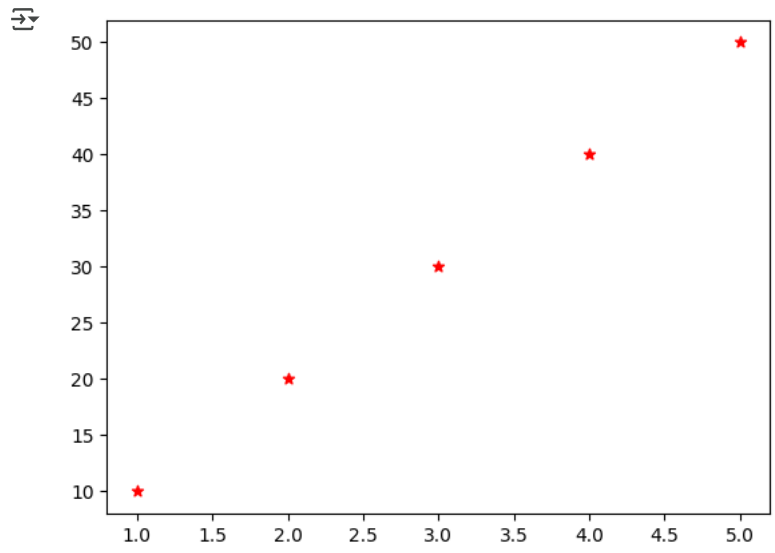

```
plt.subplot(1,2,2)
plt.plot(x,y,color="r",linestyle=":",linewidth=4)

plt.subplot(1,2,1)
plt.plot(x,y2,color="g",linestyle="--",linewidth=4)
plt.show()
```




```
student={"Amey":80,"Ramesh":90,"Ajit":92}
name=list(student.keys())
marks=list(student.values())

plt.bar(name,marks,color="r")
plt.show()
```

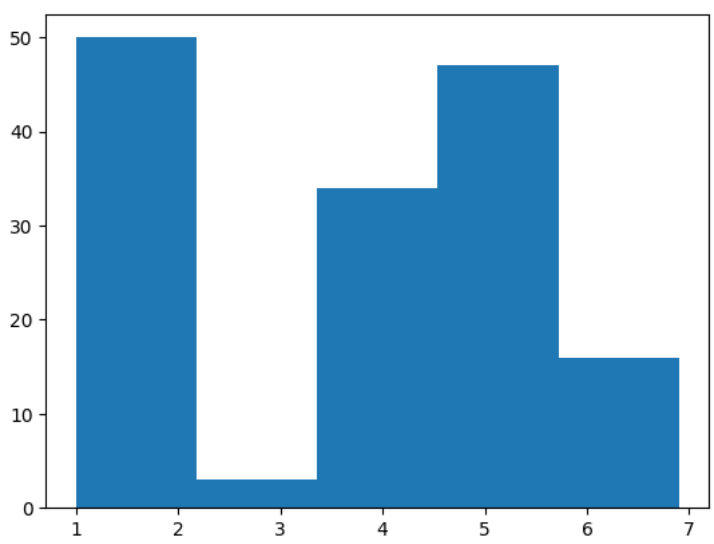



```
plt.scatter(x,y,color="r",marker="*")
plt.show()
```



```
x=[1,2,3,4,5]
y=[4,5,6,7,8]
z=[7,8,9,10,11]
```

```
plt.hist(iris['petal_length'],bins=5)
plt.show()
```




```
x=[1,2,3,4,5]
```