

```
In [134... import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import numpy.ma as ma
```

Q.1 Convert a 1D array to a 2D array with 2 rows

```
In [6]: a=np.array(['1','2','3','5'])
b=a.reshape(2,2)
b
```

```
Out[6]: array([[ '1', '2'],
               [ '3', '5']], dtype='<U1')
```

Q.2 Get the common items between a and b

```
In [9]: a = np.array([1,2,3,2,3,4,3,4,5,6])
b = np.array([7,2,10,2,7,4,9,4,9,8])
c=np.intersect1d(a,b)
c
```

```
Out[9]: array([2, 4])
```

Q.3 Get all items between 5 and 10 from a

```
In [34]: a = np.array([2, 6, 1, 9, 10, 3, 27])
c=np.where((a>5)&(a<10))
print(a[c])
```

```
[6 9]
```

4 Limit the number of items printed in python NumPy array a to a maximum of 6 elements.

```
In [141... arr = np.array([25, 32, 38, 47, 53, 66, 73, 79, 88, 95, 108])
print("Array...", arr)

maskArr = ma.masked_array(arr, mask =[0, 0, 0, 1, 1, 1, 1, 1, 0, 0, 0])
print("Our Masked Array...", maskArr)
```

```
Array... [ 25  32  38  47  53  66  73  79  88  95 108]
Our Masked Array... [25 32 38 -- -- -- -- -- 88 95 108]
```

pandas

1 Compute the minimum, 25th percentile, median, 75th, and maximum of of a given series

```
In [64]: num_state = np.random.RandomState(100)
num_series = pd.Series(num_state.normal(10, 4, 20))
print("Original Series:")
print(num_series)
result = np.percentile(num_series, q=[0, 25, 50, 75, 100])
print("\nMinimum, 25th percentile, median, 75th, and maximum of a given series:")
print(result)
```

Original Series:

```
0      3.000938
1     11.370722
2     14.612143
3      8.990256
4     13.925283
5     12.056875
6     10.884719
7      5.719827
8      9.242017
9     11.020006
10     8.167892
11     11.740654
12     7.665620
13     13.267388
14     12.690883
15     9.582355
16     7.874878
17     14.118931
18     8.247458
19     5.526727
dtype: float64
```

Minimum, 25th percentile, median, 75th, and maximum of a given series:
[3.00093811 8.09463867 10.23353705 12.21537733 14.61214321]

1. Creating A Pandas Data Frame and Using Sample Data Sets

```
In [65]: data = [11,45,36,11,99,100]
df = pd.DataFrame(data, columns=['Numbers'])
df
```

```
Out[65]:
```

	Numbers
0	11
1	45
2	36
3	11
4	99
5	100

3.Using NumPy, create a Pandas Data Frame with five rows and three columns.

```
In [71]: data = [['a', 10, 'm'], ['b', 15, 'm'], ['c', 14, 'f'], ['d', 15, 'm'], ['e', 11, 'f']]
df = pd.DataFrame(data, columns=['Name', 'Age', 'gender'])
df
```

Out[71]:

	Name	Age	gender
0	a	10	m
1	b	15	m
2	c	14	f
3	d	15	m
4	e	11	f

1. For a Pandas Data Frame created from a NumPy array, what is the default behavior for the labels for the columns? For the rows?

```
In [72]: numpyArray = np.array([[15, 22, 43],
                                [33, 24, 56]])
panda_df = pd.DataFrame(data = numpyArray,
                        index = ["Row_1", "Row_2"],
                        columns = ["Column_1",
                                "Column_2", "Column_3"])

panda_df
```

Out[72]:

	Column_1	Column_2	Column_3
Row_1	15	22	43
Row_2	33	24	56

5. take csv file contains at least 10,000 rows and 12 columns which numerical and text values according to that continue following steps.

```
In [88]: df=pd.read_csv('Downloads/Salaries.csv')
df
```

Out[88]:

	Id	EmployeeName	JobTitle	BasePay	OvertimePay	OtherPay	Benefits	T
0	1	NATHANIEL FORD	GENERAL MANAGER-METROPOLITAN TRANSIT AUTHORITY	167411.18	0.00	400184.25	NaN	56
1	2	GARY JIMENEZ	CAPTAIN III (POLICE DEPARTMENT)	155966.02	245131.88	137811.38	NaN	53
2	3	ALBERT PARDINI	CAPTAIN III (POLICE DEPARTMENT)	212739.13	106088.18	16452.60	NaN	33
3	4	CHRISTOPHER CHONG	WIRE ROPE CABLE MAINTENANCE MECHANIC	77916.00	56120.71	198306.90	NaN	33
4	5	PATRICK GARDNER	DEPUTY CHIEF OF DEPARTMENT, (FIRE DEPARTMENT)	134401.60	9737.00	182234.59	NaN	32
...
148649	148650	Roy I Tillery	Custodian	0.00	0.00	0.00	0.0	
148650	148651	Not provided	Not provided	NaN	NaN	NaN	NaN	
148651	148652	Not provided	Not provided	NaN	NaN	NaN	NaN	
148652	148653	Not provided	Not provided	NaN	NaN	NaN	NaN	
148653	148654	Joe Lopez	Counselor, Log Cabin Ranch	0.00	0.00	-618.13	0.0	

148654 rows × 13 columns

1. Write the code to show the number of rows and columns in data frame.

```
In [94]: data = [11,45,36,11,99,100]
df = pd.DataFrame(data, columns=['Numbers'])
print(df)
rows = df.shape[0]
cols = df.shape[1]
print('rows',rows)
print('column',cols)
```

```

Numbers
0      11
1      45
2      36
3      11
4      99
5     100
rows 6
column 1

```

1. How might you show the first few rows of data frame?

```

In [98]: data = [11,45,36,11,99,100]
df = pd.DataFrame(data, columns=['Numbers'])
df1=df.head(2)
print(df1)

```

```

Numbers
0      11
1      45

```

1. If you select a single column from the diamonds Data Frame, what will be the type of the return value?

```

In [105]: data = [['a', 10,'m'], ['b', 15,'m'], ['c', 14,'f'],['d',15,'m'],['e',11,'f']]
df = pd.DataFrame(data, columns=['Name', 'Age', 'gender'])
col=df.shape[1]
type(col)

```

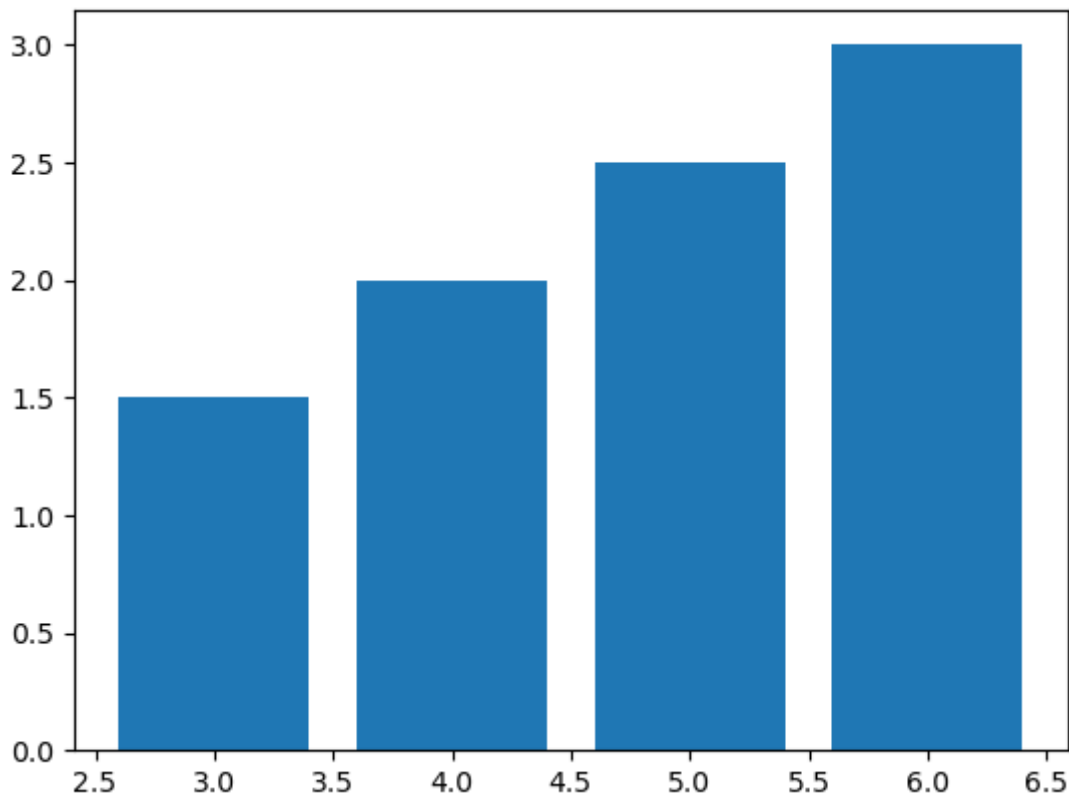
Out[105]: int

1. Create a line plot using the x and y values provided below. Label the y-axis as "Y" and label the x-axis as "X".

```

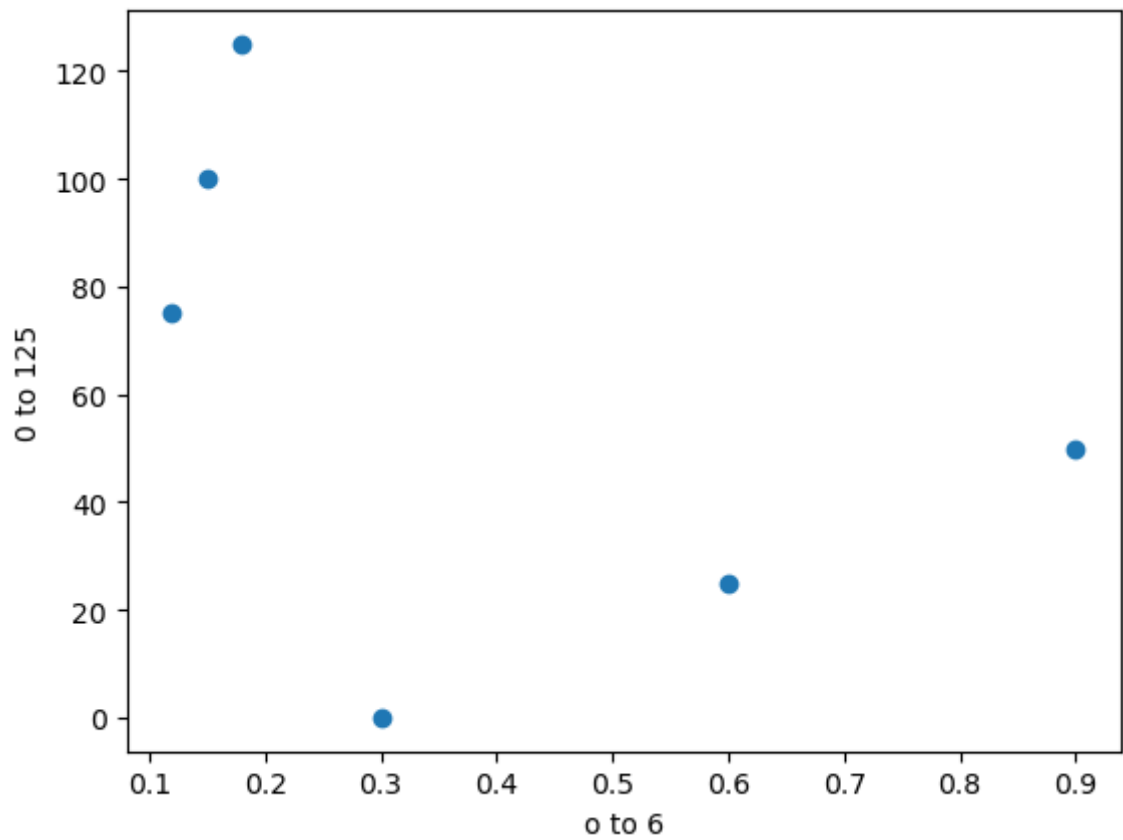
In [114]: x = [3, 4, 5, 6]
y = [1.5, 2, 2.5, 3]
plt.bar(x,y)
plt.show()

```



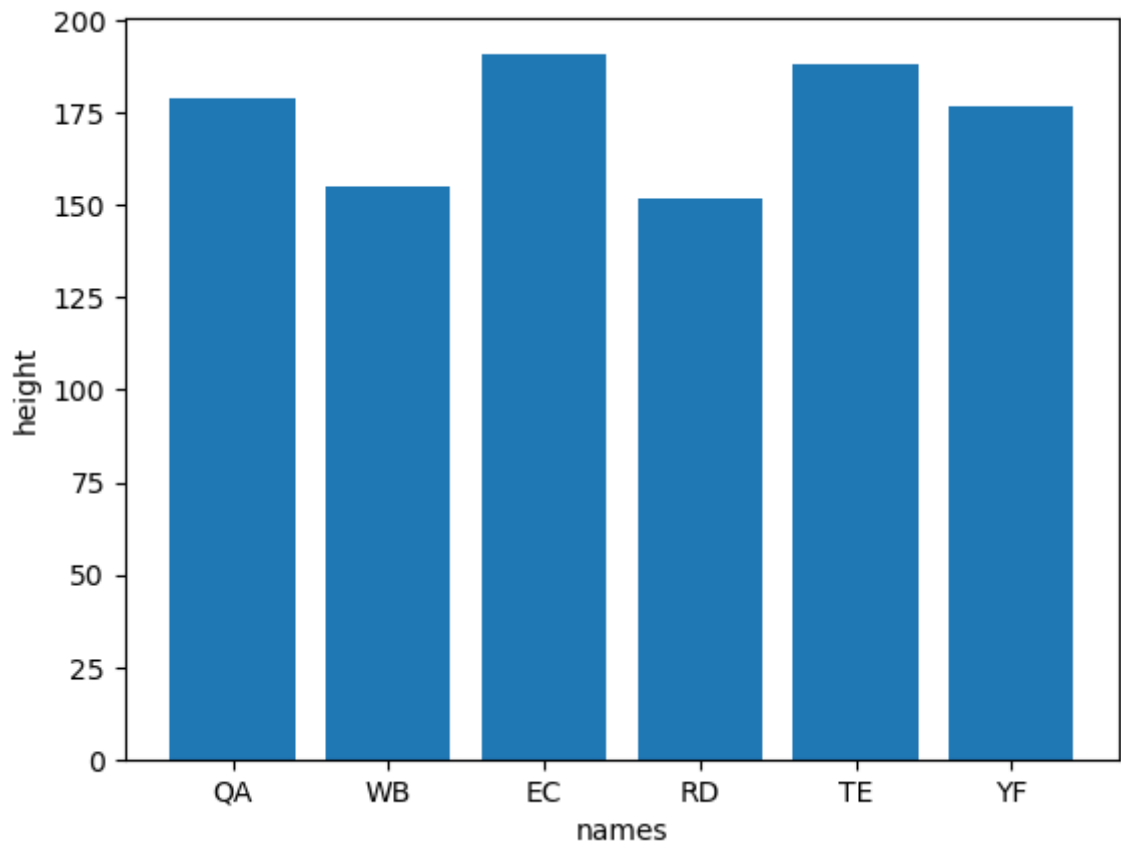
1. Create an array of numbers between 0 and 6 with increments of 0.3 and name its "x". Then on the same plot, plot x , x^2 , x^3 , and x^4 . For consistency, use the following style lines respectively, "ro", "bs", "g", and ":". Lastly, make sure that the x-axis covers 0 to 6, while the y-axis spans from 0 to 125. Do not worry if you are not familiar with the style lines — you will recognize them as soon as you see the plot.

```
In [122... a=np.array([0.3,0.6,0.9,0.12,0.15,0.18])
b=np.array([0,25,50,75,100,125])
plt.scatter(a,b)
plt.xlabel('0 to 6')
plt.ylabel('0 to 125')
plt.show()
```



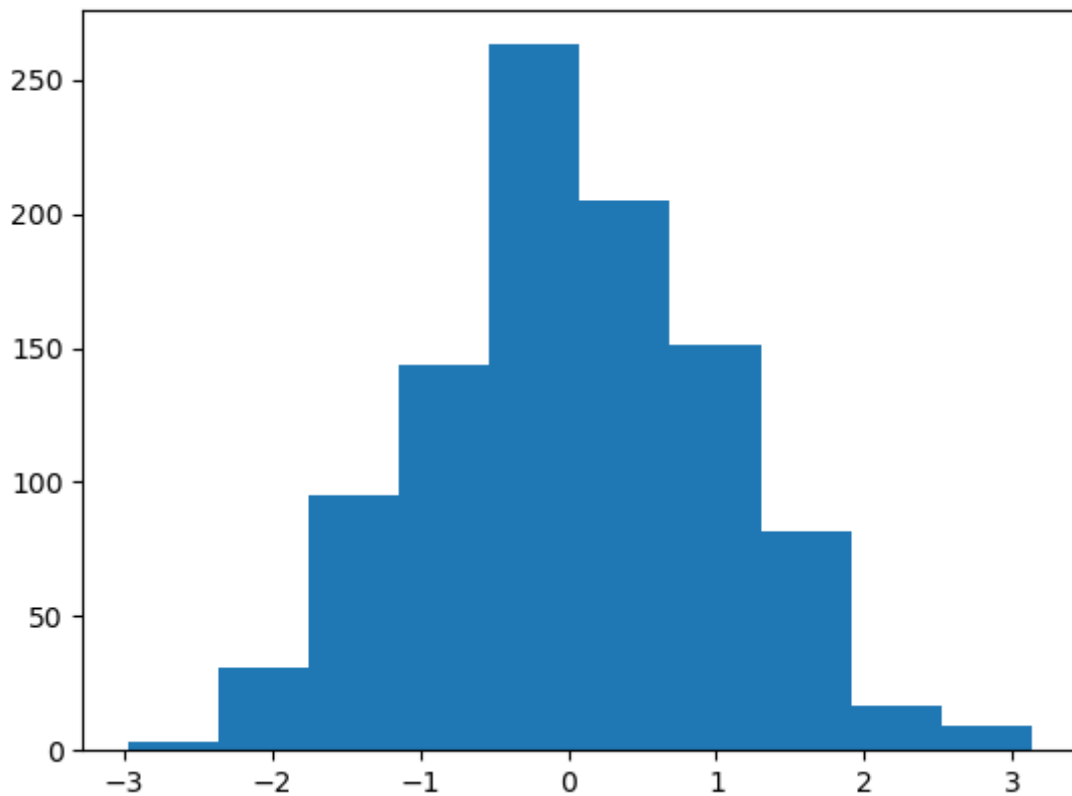
1. Heights and initials of a group of individuals are provided below. Create a bar plot titled "Height Comparison" to compare the heights among this group.

```
In [117... height = [179, 155, 191, 152, 188, 177]
names = ['QA', 'WB', 'EC', 'RD', 'TE', 'YF']
plt.bar(names,height)
plt.xlabel('names')
plt.ylabel('height')
plt.show()
```



1. Plot a histogram of x , where x consists of 100,000 randomly-selected points with a normal distribution (hint: you can use `numpy.random.randn()` to generate the random points). The histogram should have 10 bins. Look at how the histogram changes when we try 20 and 50 bins.

```
In [120... data = np.random.normal(size=1000)
plt.hist(data, bins=10)
plt.show()
```

1. What are the features of TensorFlow?

1. Good support of deep neural networks and machine learning concepts
2. Efficiently works with mathematical expressions involving multi-dimensional arrays
3. GPU/CPU computing where the same code can be executed on both architectures
4. High scalability of computation across machines and huge data sets

1. List a few limitations of TensorFlow.

- 1 Has the GPU memory conflicted with Theano if imported in the same scope?
- 2 It has dependencies with other libraries.
- 3 Requires prior knowledge of advanced calculus and linear algebra along with the pretty good understanding of machine learning.

1. What do you know about supervised and unsupervised machine learning?

- 1) supervised learning is a learning that pre input is given by user and output is predicted by machine algorithm
- 2)unsupervised learning is a learning only input data is given and output is predict by machine algorithm