1. # Ans : Bar charts are good for comparisons, while line charts work better for trends. Scatter plot charts are good

# for relationships and distributions, but pie charts should be used only for simple compositions — never for

# comparisons or distributions.

2. # Ans : Compound interest makes your money grow faster because interest is calculated on the accumulated interest

# over time as well as on your original principal. Compounding can create a snowball effect, as the original

# investments plus the income earned from those investments grow together.

3. # Ans : Histogram is a type of a plot which hold the total number of counts or the percentile value in y-axis w.r.t.

# corresponding x-variables. When we say the number of counts it means that for a particular range in x-axis

# what will be the number of counts of the data points. When we talk about percentile it shows the probability

# occurrence of the x-variable or what percentage of data are below that particular percetile value w.r.t.

# particular x-variable range. Numpy has a built-in numpy. histogram() function.

4. # Ans : We can use figure(figsize=(10,8)) function inside the matplot.pyplot library which we scale

# down or up the graph.

figure(figsize=(10,8))

5. # Ans :

import numpy as np

a1=np.array([[1,2,3],[4,5,6],[6,7,8]])

a2=np.array([[10,20,30],[40,50,60],[60,70,80]])

# suppose above are the two arrays of shape 3 x 3

print(a1)

print()

print(a2)

[[1 2 3]

[4 5 6]

[6 7 8]]

[[10 20 30]

[40 50 60]

[60 70 80]]

#standard multiplication

a1\*a2

array([[ 10, 40, 90],

[160, 250, 360],

[360, 490, 640]])

# When we do standard multiplication in that case values of the same indexes in the array will get multiply.

# Like in above example. A1(i,j) x A2(i,j)

# dot product

np.dot(a1,a2)

array([[ 270, 330, 390],

[ 600, 750, 900],

[ 820, 1030, 1240]])

# In case of dot product vector, multiplication will take place between row of first array and column of secodn array

# .respectively. Like first row of array a1 will be multiply by column value of array a2 one by one and then added.

#outer multiplication

np.outer(a1,a2)

array([[ 10, 20, 30, 40, 50, 60, 60, 70, 80],

[ 20, 40, 60, 80, 100, 120, 120, 140, 160],

[ 30, 60, 90, 120, 150, 180, 180, 210, 240],

[ 40, 80, 120, 160, 200, 240, 240, 280, 320],

[ 50, 100, 150, 200, 250, 300, 300, 350, 400],

[ 60, 120, 180, 240, 300, 360, 360, 420, 480],

[ 60, 120, 180, 240, 300, 360, 360, 420, 480],

[ 70, 140, 210, 280, 350, 420, 420, 490, 560],

[ 80, 160, 240, 320, 400, 480, 480, 560, 640]])

# In outer multiplication every element of first array a1 will be multiply by every element of other array a2 such

# such the number of columns will be equal to the number of element in another array a2.

6. # Ans : np.pmt(rate, nper, pv) function we will be using in order to calculate monthly mortgage payment before

# you purchase a house

# rate = The periodic interest rate

# nper = The number of payment periods

# pv = The total value of the mortgage loan

7. # Ans : Yes an array can store the string.The limitation which imposed on the string data is, whenever we store the

# data of string dtype then it should should keep in mind that the string which is having the maximum length is

# the limit.

# The dtype value is the maximum length of a string in such data. Meaning suppose if any new string we wanted

# to add, then we can add only that string which have the length either equal to that dtype value or less than

# that. If any string which are adding and if its length is more than the dtype value then it will only accpet

# to maximum length and rest of the characters will not be included.