```
# Q1

import numpy as np

url = "https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data"
iris_1d = np.genfromtxt(url, delimiter=',', dtype = "float,"*4 + "U20", names=True)
iris_2d = np.array([row.tolist()[:4] for row in iris_1d])
iris_class = np.array([row.tolist()[4] for row in iris_1d])

sepal_len = float(input("Enter the sepal length : "))
sepal_width = float(input("Enter the sepal width : "))
petal_len = float(input("Enter the petal length : "))
petal_width = float(input("Enter the petal width : "))
arr = np.array([sepal_len, sepal_width, petal_len, petal_width])
print(arr)

i = 0
dic = {}
for row in iris_2d:
  d = np.linalg.norm(row - arr)
  dic[i] = d
  i += 1
  #i=149
dic = dict(sorted(dic.items(), key=lambda item: item[1]))

cnt = 0
print("\nClass of closest 5 samples : ")
for i in dic.keys():
  cnt += 1
  print(iris_class[i])
  if cnt == 5:
    break

#np.linalg.norm() is called on an array-like input to compute the square root of the sum of squared elements
#humne x means key value pair diya for all the items kyuki apne aap d.items se ek loop chal rha hai on all elements
#and x[1] means value is picked up and then sorted and stored in sorted dictionary d
```

```
#dictionary ki key hai excel ka serial number and value hai distance jo aayi
#same k varible ko use krenge toh excel ka class aajeyga by output_column[k]
```

```
     Enter the sepal length : 2
     Enter the sepal width : 3
     Enter the petal length : 2
     Enter the petal width : 2

     Class of closest 5 samples :
     Iris-setosa
     Iris-setosa
     Iris-setosa
     Iris-setosa
     Iris-setosa
```

```
# Q2

import numpy as np

url = "https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data"
iris_1d = np.genfromtxt(url, delimiter=',', dtype = "float,"*4 + "U20", names=True)
iris_2d = np.array([row.tolist()[:4] for row in iris_1d])
iris_class = np.array([row.tolist()[4] for row in iris_1d])

sepal_len = float(input("Enter the sepal length : "))
sepal_width = float(input("Enter the sepal width : "))
petal_len = float(input("Enter the petal length : "))
petal_width = float(input("Enter the petal width : "))
arr = np.array([sepal_len, sepal_width, petal_len, petal_width])

from scipy.spatial import distance

i = 0
dic = {}
for row in iris_2d:
  d = distance.cityblock(row, arr)
  dic[i] = d
  i += 1
```

```
    i += 1
dic = dict(sorted(dic.items(), key=lambda item: item[1]))

cnt = 0
print("\nClass of closest 5 samples : ")
for i in dic.keys():
  cnt += 1
  print(iris_class[i])
  if cnt == 5:
    break

  #scipy.spatial.distance.cityblock computes the City Block (Manhattan) distance between vectors u and v
```

```
 Enter the sepal length : 2
 Enter the sepal width : 3
 Enter the petal length : 2
 Enter the petal width : 2

 Class of closest 5 samples :
 Iris-setosa
 Iris-setosa
 Iris-setosa
 Iris-setosa
 Iris-setosa
```

```
# Q3

import numpy as np

dic = {"best case" : [], "avg case" : [], "worst case" : []}
limit = 1

df = np.genfromtxt("student.csv", delimiter = ',', dtype = int, names = True)
rno = np.array([row.tolist()[0] for row in df])
marks = np.array([row.tolist()[1:] for row in df])

for i in range(len(rno)):
  print("Mean of marks of student with roll no ", rno[i], " : ", np.mean(marks[i]))
  if abs(np.mean(marks[i]) - np.mean(marks)) <= limit:
```

```
      dic["avg case"].append(rno[i])
    elif np.mean(marks[i]) > np.mean(marks):
      dic["best case"].append(rno[i])
    else:
      dic["worst case"].append(rno[i])

print("\nTotal mean of data stored : ", np.mean(marks))

print("\n", dic)
```

```
     Mean of marks of student with roll no  1  :  63.6
     Mean of marks of student with roll no  2  :  62.4
     Mean of marks of student with roll no  3  :  67.4
     Mean of marks of student with roll no  4  :  95.6
     Mean of marks of student with roll no  5  :  67.4
     Mean of marks of student with roll no  6  :  65.0
     Mean of marks of student with roll no  7  :  41.4
     Mean of marks of student with roll no  8  :  45.6
     Mean of marks of student with roll no  9  :  54.4
     Mean of marks of student with roll no  10  :  76.0

     Total mean of data stored :  63.88

      {'best case': [3, 4, 5, 6, 10], 'avg case': [1], 'worst case': [2, 7, 8, 9]}
```

```
# Q4

import numpy as np
import random as r

n= int(input("Enter the no of elements required in random sequence : "))

ran = []
for i in range(n):
  ran.append(r.randint(0, 149))
ran = np.array(ran)

url = "https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data"
```

```
iris_1d = np.genfromtxt(url, delimiter=',', dtype = "float,"*4 + "U20", names=True)
iris_2d = np.array([row.tolist()[:4] for row in iris_1d])

df = []
for i in ran:
  df.append(iris_2d[i])
df = np.array(df)

print("\nMean of randomly selected samples : ", np.mean(df))
print("Standard Deviation of randomly selected samples : ", np.std(df))
```

```
     Enter the no of elements required in random sequence : 6

     Mean of randomly selected samples :  3.775
     Standard Deviation of randomly selected samples :  2.002342378315956
```