```
#
# Name : Aravind Kumar kaspe
# Banned ID : 001291145
#
# Description : Python script reads data from the text file ‘5930 – MP4 Data.txt’.
#               The data file contains information about students in a class who
#               have taken 3 exams. The script has 6 functions, the details of
#               the functions are in the documentation and the script performs
#               below operations :
#               (1) Print the student’s information out in the original form
#               (2) Sort the data by the student’s last name
#               (3) Print the information, again
#               (4) Sort the data by the student’s test average
#               (5) Print the information 1 more time.
#

def getScores():
#
# Opens the data file of names and scores... firstName, lastName, score1,
# score2, score3... reads each line of data as a str, divides the line into
# the 5 values... str, str, int, int, int... puts those values in a list,
# and returns a list of those lists.
#
# There are no parameters.
#
# Returns a list of lists... each list contains a str, str, int, int, int.
#
    with open(filename) as file:
        data = file.read()
        source = data.split('\n')
        source.pop()
        get_Scores_list =[]
        for i in source:
            b = []
            k=i.split()
            b.append(k[0])
            b.append(k[1])
            b.append(int(k[2]))
            b.append(int(k[3]))
            b.append(int(k[4]))
            get_Scores_list.append(b)
        return get_Scores_list


def addTestAverage(studentScores):
#
# Finds the average of each student's test scores, and then appends that
# average onto the end of that student's list. So, each student list now
# contains str, str, int, int, int, float.
#
```

```
# studentScores A list of lists, each list contains a str, str, int,
# int, int which are firstName, lastName, test1, test2,
# test3.
#
# There is no return value.
#
    for i in studentScores:
        test_average = round((i[2]+i[3]+i[4])/3,2)
        i.append(test_average)


def calcTotals(studentScores):
#
# Finds the average of test1, test2, test3, and the total average. Returns
# those 4 values in a list.
#
# studentScores A list of lists, each list contains a str, str, int,
# int, int, float which are firstName, lastName, test1,
# test2, test3, average.
#
# Returns a list with 4 values... float, float, float, float... which are
# test1 avg, test2 avg, test3 avg, total avg.
#
     calcTotals = []
     test1 = 0
     test2 = 0
     test3 = 0
     totalAvg = 0
     count = 0

     for i in studentScores:
         count += 1
         test1 += i[2]
         test2 += i[3]
         test3 += i[4]
         totalAvg += i[5]
     calcTotals.append(test1/count)
     calcTotals.append(test2/count)
     calcTotals.append(test3/count)
     calcTotals.append(totalAvg/count)

     return calcTotals


def printScores(studentScores, totals):
#
# Prints out the entire list including firstName, lastName, score1, score2,
# score3, average. There is a header for each column. The totals are
# printed at the end.
#
```

```
# studentScores A list of lists, each list contains a str, str, int,
# int, int, float which are firstName, lastName, test1,
# test2, test3, average.
# totals A list of 4 float values... the averages for test1,
# test2, test3, and totalAverage.
#
# There is no return value.
#
     print(f"{'Name':<22} {'Exam1':<7} {'Exam2':<7} {'Exam3':<7} {'Avg':>5}")
     for i in studentScores:
         print(f'{i[0]+" "+i[1]:<17} {i[2]:>10} {i[3]:>7} {i[4]:>7} {i[5]:>7.2f}')
     print(f'Total {totals[0]:>22.2f} {totals[1]:>7.2f} {totals[2]:>7.2f}
{totals[3]:>7.2f}')
     print()


def sortByName(studentScores):
#
# Sorts the list of student info by the student's last name. Uses the
# Bubble Sort algorithm.
#
# studentScores A list of lists, each list contains a str, str, int,
# int, int, float which are firstName, lastName, test1,
# test2, test3, average.
#
# There is no return value.
#
     for i in range(len(studentScores) - 1):
         for j in range(len(studentScores) - 1):
             if studentScores[j][1] > studentScores[j+1][1]:
                 temp = studentScores[j]
                 studentScores[j] = studentScores[j+1]
                 studentScores[j+1] = temp

def sortByAverage(studentScores):
#
# Sorts the list of student info by the test average. Uses the
# Bubble Sort algorithm.
#
# studentScores A list of lists, each list contains a str, str, int,
# int, int, float which are firstName, lastName, test1,
# test2, test3, average.
#
# There is no return value.
#
      for i in range(len(studentScores) - 1):
          for j in range(len(studentScores) - 1):
              if studentScores[j][5] < studentScores[j+1][5]:
                  temp = studentScores[j]
                  studentScores[j] = studentScores[j+1]
                  studentScores[j+1] = temp


filename = '5930 - MP4 Data.txt'
data = getScores()

addTestAverage(data)
c = calcTotals(data)
printScores(data,c)

sortByName(data)
printScores(data,c)

sortByAverage(data)
printScores(data,c)
```