**Mahesh Kumar Uppu**

**700741747**

**Question 1**

The following is a list of 10 students ages:

ages = [19, 22, 19, 24, 20, 25, 26, 24, 25, 24]

• Sort the list and find the min and max age

• Add the min age and the max age again to the list

• Find the median age (one middle item or two middle items divided by two)

• Find the average age (sum of all items divided by their number)

• Find the range of the ages (max minus min)

**Solution**

import statistics # Importing statistics Library to use median and mean

ages = [19, 22, 19, 24, 20, 25, 26, 24, 25, 24]

sortedAges=sorted(ages) #sorting and storing in new list

leastAge=min(ages) #finding min age

highestAge=max(ages) #finding max age

newAges = ages + [leastAge,highestAge] #adding min and max age again back to the list

medianAge = statistics.median(newAges) #finding median age

aveargeAge= statistics.mean(newAges) #finding average age

ageRange=highestAge-leastAge #finding age ranges of the list

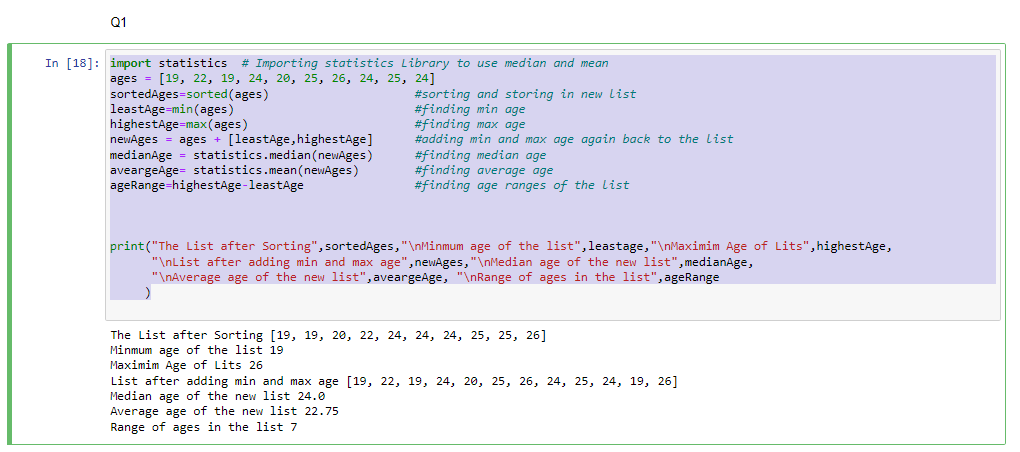
print("The List after Sorting",sortedAges,"\nMinmum age of the list",leastage,"\nMaximim Age of Lits",highestAge,

"\nList after adding min and max age",newAges,"\nMedian age of the new list",medianAge,

"\nAverage age of the new list",aveargeAge, "\nRange of ages in the list",ageRange

)

**Output**



**Question 2**

• Create an empty dictionary called dog

• Add name, color, breed, legs, age to the dog dictionary

• Create a student dictionary and add first\_name, last\_name, gender, age, marital status, skills, country, city and address as keys for the dictionary

• Get the length of the student dictionary

• Get the value of skills and check the data type, it should be a list

• Modify the skills values by adding one or two skills

• Get the dictionary keys as a list

• Get the dictionary values as a list

**Solution**

dog ={} #empty dictionary named dog

#adding data in the form of key value pairs to the dictionary

dog["name"] = "Bull"

dog["color"] = "Black"

dog["breed"] = "BullDog"

dog["legs"] = 4

dog["age"] = 10

#creating student dictionary with all the data at once

student = {

"first\_name": "Mahesh",

"last\_name": "Uppu",

"gender": "male",

"age": 24,

"skills": ['python','java','salesforce'],

"country": "USA",

"city": "Kansas",

"address": "OverlandPark"

}

studentDictionaryLength=len(student) #finding length of student dictionary

studentSkills=student.get('skills') #getting skills value of the dictionary

student["skills"]=['c','python','salesforce','java'] #modifying skills value of the dictionary

studentkeys = list(student.keys()) #getting keys of the dictionary

studentValues=list(student.values()) #getting values of the dictionary

print("Length of Student Dictionary",studentDictionaryLength)

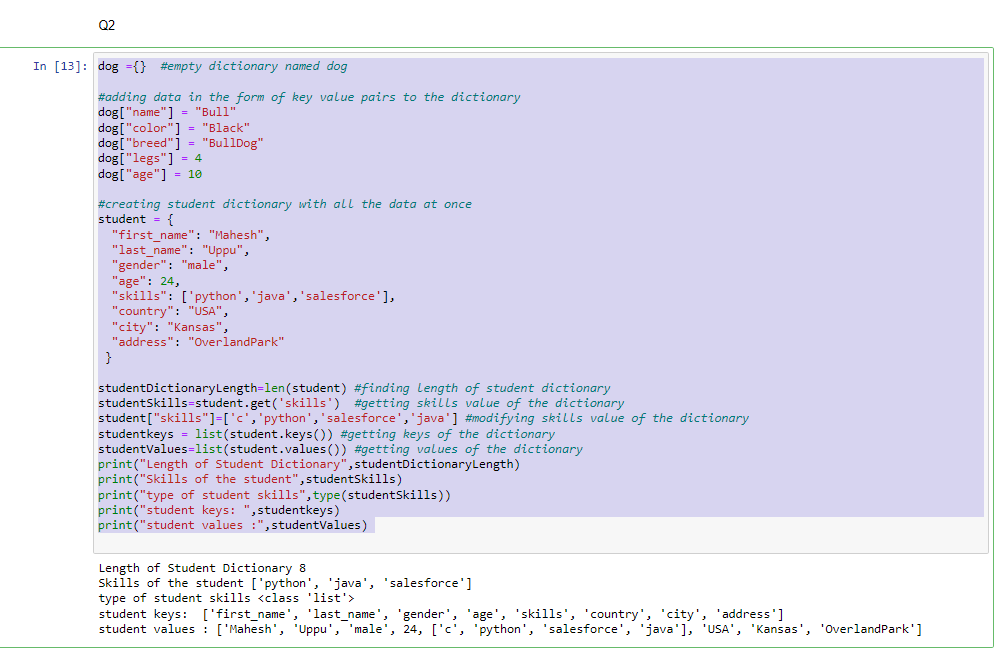
print("Skills of the student",studentSkills)

print("type of student skills",type(studentSkills))

print("student keys: ",studentkeys)

print("student values :",studentValues)

**Output**



**Question 3**

• Create a tuple containing names of your sisters and your brothers (imaginary siblings are fine)

• Join brothers and sisters tuples and assign it to siblings

• How many siblings do you have?

• Modify the siblings tuple and add the name of your father and mother and assign it to family\_members

**Solution**

brothers = ("pavan", "suman", "raman") #brothers tuple creation

sisters = ("laksmi", "simran", "deepika") #sisters tuple creation

siblings = brothers+sisters #creating sibiling by combining both brothers and sisters

print("brothers tuple",brothers)

print("sisters tuple",sisters)

print("siblings before adding father and mother",siblings)

siblingcount=len(siblings) #finding siblings count before adding father and mother

tempconversion = list(siblings)

tempconversion.append("Ramana") #adding bother father and mothers names individually after coverting to list

tempconversion.append("Kumari")

siblings=tuple(tempconversion) # converting back to tuple

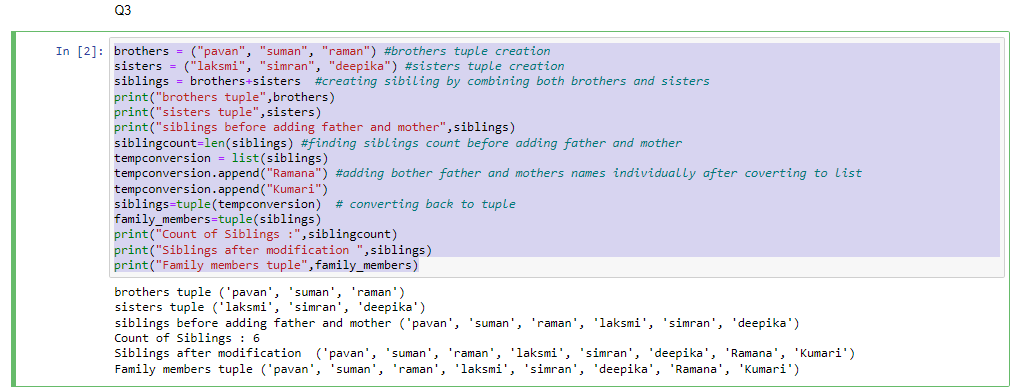
family\_members=tuple(siblings)

print("Count of Siblings :",siblingcount)

print("Siblings after modification ",siblings)

print("Family members tuple",family\_members)

**Output**



**Question 4**

it\_companies = {'Facebook', 'Google', 'Microsoft', 'Apple', 'IBM', 'Oracle', 'Amazon'}

A = {19, 22, 24, 20, 25, 26}

B = {19, 22, 20, 25, 26, 24, 28, 27}

age = [22, 19, 24, 25, 26, 24, 25, 24]

• Find the length of the set it\_companies

• Add 'Twitter' to it\_companies

• Insert multiple IT companies at once to the set it\_companies

• Remove one of the companies from the set it\_companies

• What is the difference between remove and discard

• Join A and B

• Find A intersection B

• Is A subset of B

• Are A and B disjoint sets

• Join A with B and B with A

• What is the symmetric difference between A and B

• Delete the sets completely

• Convert the ages to a set and compare the length of the list and the set

**Solution**

it\_companies = {'Facebook', 'Google', 'Microsoft', 'Apple', 'IBM', 'Oracle', 'Amazon'}

A = {19, 22, 24, 20, 25, 26}

B = {19, 22, 20, 25, 26, 24, 28, 27}

age = [22, 19, 24, 25, 26, 24, 25, 24]

print("Original IT Companies",it\_companies)

itCompaniesLength = len(it\_companies) #getting length of it\_companies

it\_companies.add("Twitter")

new\_companies = ["Salesfore","TATA","TCS"]

it\_companies.update(new\_companies) #adding multiple companies to the set

print("IT Companies after adding new companies",it\_companies)

it\_companies.remove("Facebook") #removing one item from IT Companies

#Difference between discard and remove

'''

discard() in Python, removes the element from the set only if the element is present in the set

If the element is not present in the set, then no error or exception is raised and the original set is printed.

remove() in Python, removes the element from the set only if the element is present in the set,

just as the discard() method does but If the element is not present in the set, then an error or exception is raised.

'''

jionedSets=A.union(B) #jioning A&B

intersectedset= A.intersection(B) #intersection of A and B

subset\_check= A.issubset(B) #checkAing A for subset condition with B

disjoint\_check= A.isdisjoint(B) #checking for disjiont

aWithBset=A.union(B) #jioning a with B

bWithAset= B.union(A) #jiioning bwith a

print("joining of A and B: ", jionedSets)

print("intersection of A and B: ", intersectedset)

print("Is A subset of B?", subset\_check)

print("Are A and B disjoint sets? ", disjoint\_check)

print("After joining A With B: ",aWithBset, "and joining B with A: ",bWithAset)

#What is the symmetric difference between A and B

aSymmerticDifference = A.symmetric\_difference(B)

print("Symmetric difference between A and B:", aSymmerticDifference)

del A,B #deleting the sets

I = set(age) #Converting the ages to a set and comparing the length of the list and the set

print("Is length of list and set is same :", len(I)==len(age))

**Output:**



**Question 5**

The radius of a circle is 30 meters.

• Calculate the area of a circle and assign the value to a variable name of \_area\_of\_circle\_

• Calculate the circumference of a circle and assign the value to a variable name of \_circum\_of\_circle\_

• Take radius as user input and calculate the area.

**Solution**

radius = 30

pi = 3.14

\_area\_of\_circle\_= pi\*radius\*radius #caluclating area ciecle

print("Area of the default circle is: ", \_area\_of\_circle\_)

\_circum\_of\_circle\_= 2\*pi\*radius #caluclating the circumference

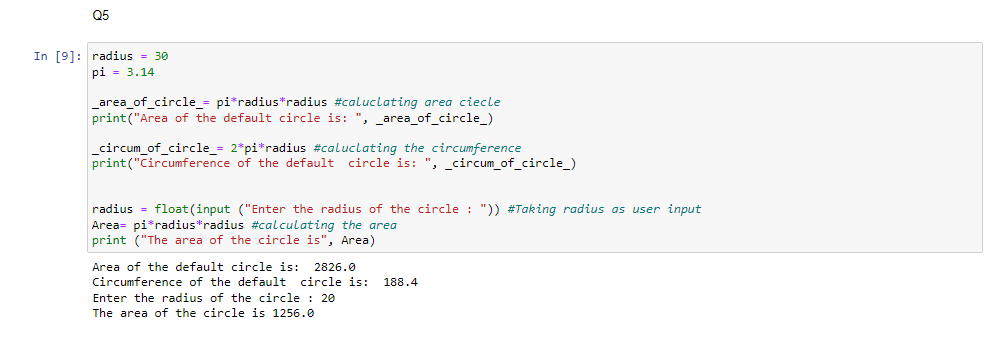
print("Circumference of the default circle is: ", \_circum\_of\_circle\_)

radius = float(input ("Enter the radius of the circle : ")) #Taking radius as user input

Area= pi\*radius\*radius #calculating the area

print ("The area of the circle is", Area)

**Output**



**Question 6**

“I am a teacher and I love to inspire and teach people”

• How many unique words have been used in the sentence?

Use the split methods and set to get the unique words.

**Solution**

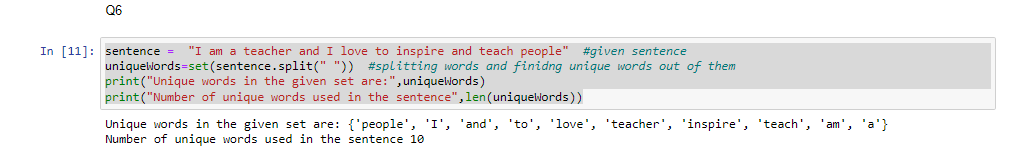
sentence = "I am a teacher and I love to inspire and teach people" #given sentence

uniqueWords=set(sentence.split(" ")) #splitting words and finidng unique words out of them

print("Unique words in the given set are:",uniqueWords)

print("Number of unique words used in the sentence",len(uniqueWords))

**Output**



**Question 7**

Use a tab escape sequence to get the following lines.

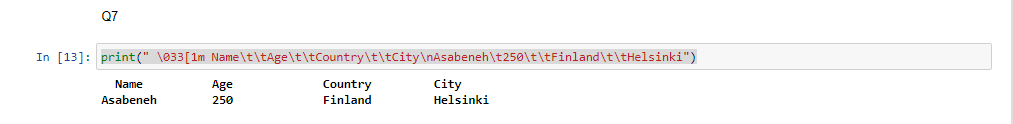
Name Age Country City

Asabeneh 250 Finland Helsinki

**Solution**

print(" \033[1m Name\t\tAge\t\tCountry\t\tCity\nAsabeneh\t250\t\tFinland\t\tHelsinki")

**Output**



**Question 8**

Use the string formatting method to display the following:

radius = 10 area = 3.14 \* radius \*\* 2

“The area of a circle with radius 10 is 314 meters square.”

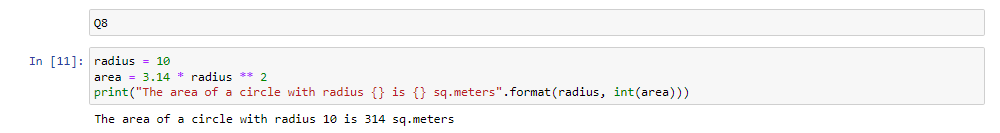
**Solution**

radius = 10

area = 3.14 \* radius \*\* 2

print("The area of a circle with radius {} is {} sq.meters".format(radius, int(area)))

**Output**



**Question 9**

Write a program, which reads weights (lbs.) of N students into a list and convert these weights to kilograms in a separate list using Loop. N: No of students (Read input from user)

Ex: L1: [150, 155, 145, 148]

Output: [68.03, 70.3, 65.77, 67.13]

**Solution**

lb\_kg\_conversion= 0.45

enteredValues=int(input('enter the number of students'))

caluclatedValues=[]

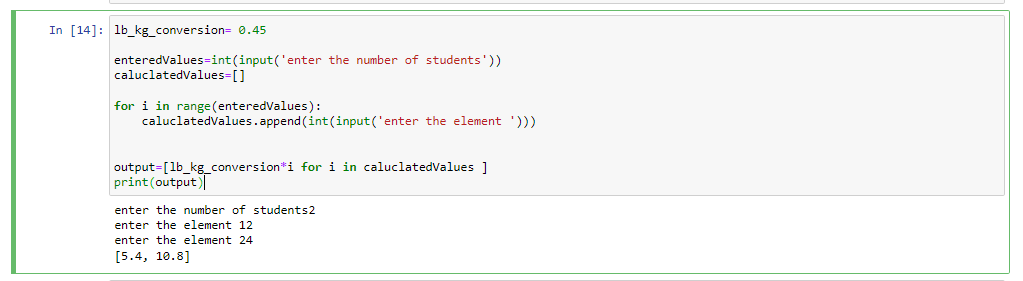
for i in range(enteredValues):

caluclatedValues.append(int(input('enter the element ')))

output=[lb\_kg\_conversion\*i for i in caluclatedValues ]

print(output)

**Output**



**Question 10**

