Machine Learning Assignment-1

Naga Chetan Kumar Reddy-700743408

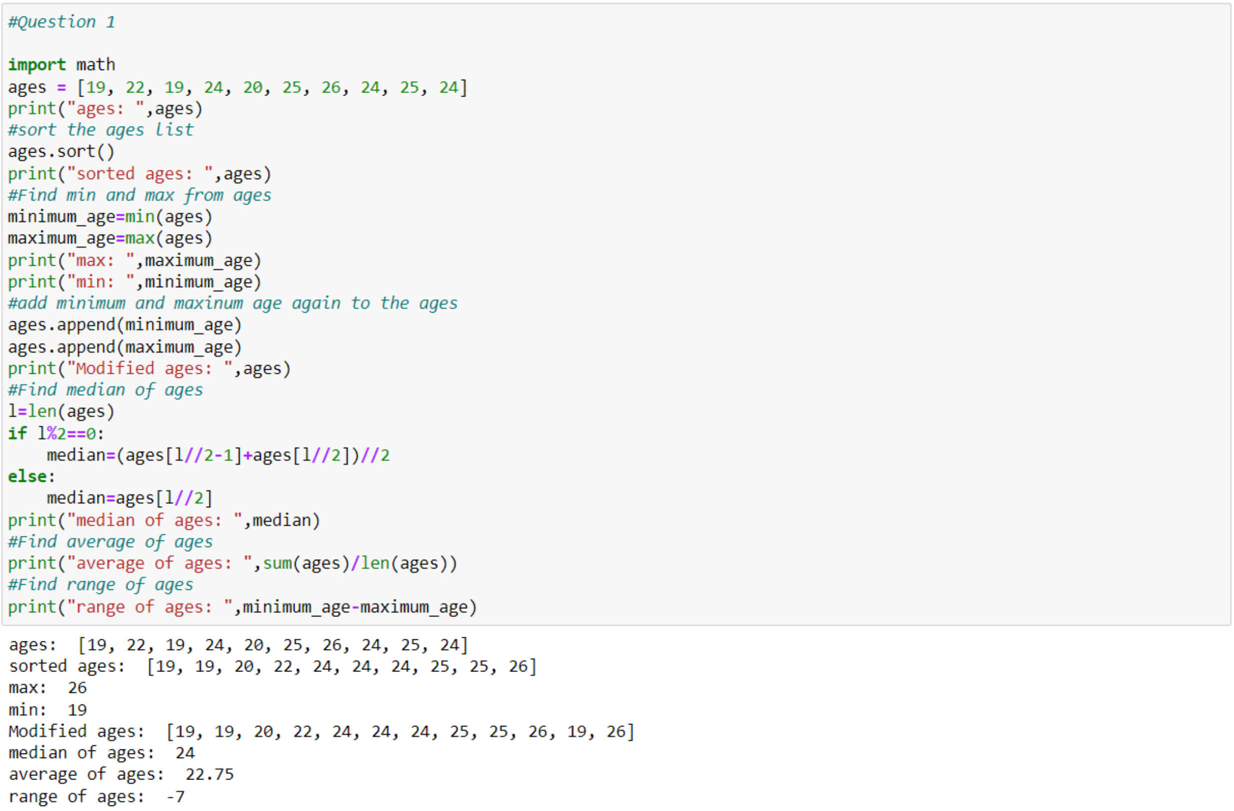
Question 1

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 Screenshot:



```
#Question 1

import math
ages = [19, 22, 19, 24, 20, 25, 26, 24, 25, 24]
print("ages: ",ages)
#sort the ages list
ages.sort()
print("sorted ages: ",ages)
#Find min and max from ages
minimum_age=min(ages)
maximum_age=max(ages)
print("max: ",maximum_age)
print("min: ",minimum_age)
#add minimum and maxinum age again to the ages
ages.append(minimum_age)
ages.append(maximum_age)
print("Modified ages: ",ages)
#Find median of ages
l=len(ages)
if l%2==0:
    median=(ages[l//2-1]+ages[l//2])//2
else:
    median=ages[l//2]
print("median of ages: ",median)
#Find average of ages
print("average of ages: ",sum(ages)/len(ages))
#Find range of ages
print("range of ages: ",minimum_age-maximum_age)
```

```
ages:  [19, 22, 19, 24, 20, 25, 26, 24, 25, 24]
sorted ages:  [19, 19, 20, 22, 24, 24, 24, 25, 25, 26]
max:  26
min:  19
Modified ages:  [19, 19, 20, 22, 24, 24, 24, 25, 25, 26, 19, 26]
median of ages:  24
average of ages:  22.75
range of ages:  -7
```

Description:

I have used sort(), min(), max(), append(), sum() and len() built in methods for sorting, finding minimum and maximum, adding items, summing the items and finding length of the list. Statistics package has been used to find the median of the ages list.

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**

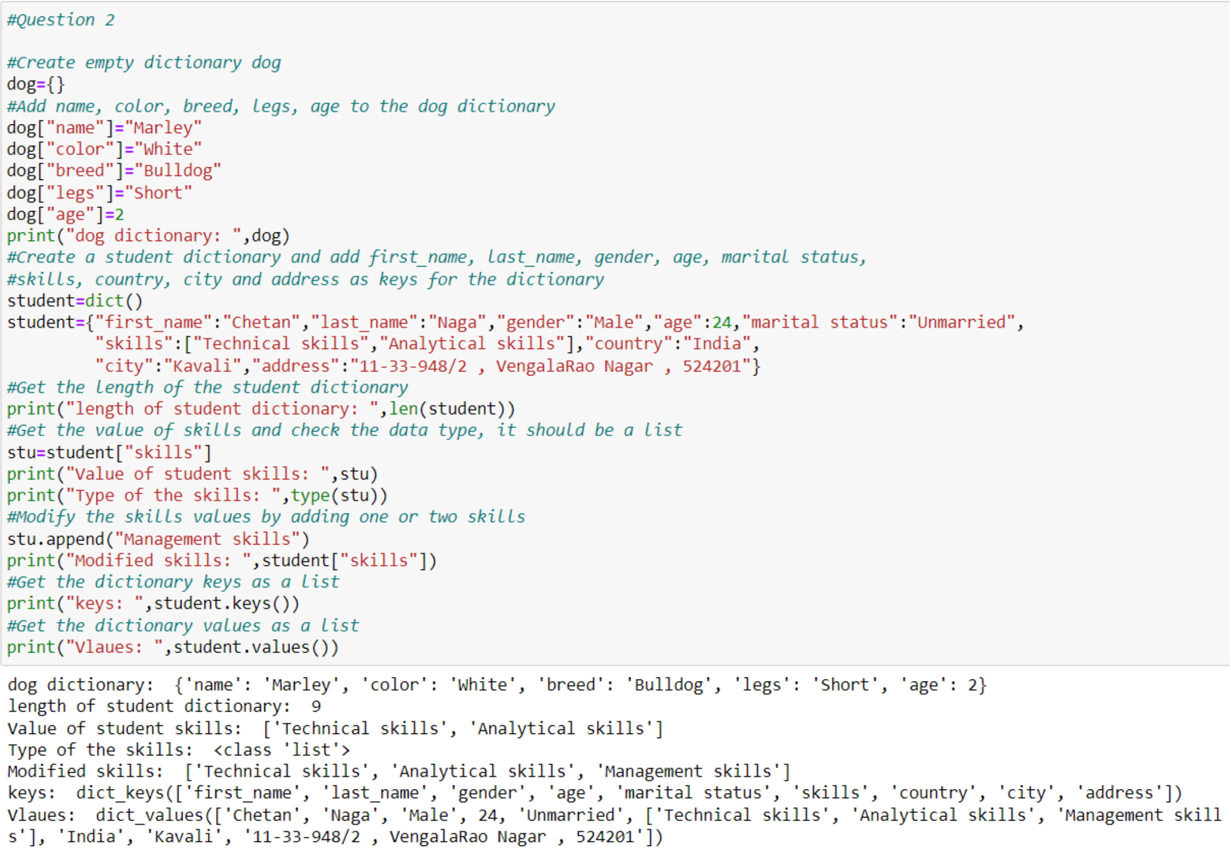
- **Get the value of skills and check the data type, it should be a list**

- **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 Screenshot:



```python
#Question 2

#Create empty dictionary dog
dog={}
#Add name, color, breed, legs, age to the dog dictionary
dog["name"]="Marley"
dog["color"]="White"
dog["breed"]="Bulldog"
dog["legs"]="Short"
dog["age"]=2
print("dog dictionary: ",dog)
#Create a student dictionary and add first_name, last_name, gender, age, marital status,
#skills, country, city and address as keys for the dictionary
student=dict()
student={"first_name":"Chetan","last_name":"Naga","gender":"Male","age":24,"marital status":"Unmarried",
         "skills":["Technical skills","Analytical skills"],"country":"India",
         "city":"Kavali","address":"11-33-948/2 , VengalaRao Nagar , 524201"}
#Get the length of the student dictionary
print("length of student dictionary: ",len(student))
#Get the value of skills and check the data type, it should be a list
stu=student["skills"]
print("Value of student skills: ",stu)
print("Type of the skills: ",type(stu))
#Modify the skills values by adding one or two skills
stu.append("Management skills")
print("Modified skills: ",student["skills"])
#Get the dictionary keys as a list
print("keys: ",student.keys())
#Get the dictionary values as a list
print("Vlaues: ",student.values())
```

```
dog dictionary:  {'name': 'Marley', 'color': 'White', 'breed': 'Bulldog', 'legs': 'Short', 'age': 2}
length of student dictionary:  9
Value of student skills:  ['Technical skills', 'Analytical skills']
Type of the skills:  <class 'list'>
Modified skills:  ['Technical skills', 'Analytical skills', 'Management skills']
keys:  dict_keys(['first_name', 'last_name', 'gender', 'age', 'marital status', 'skills', 'country', 'city', 'address'])
Vlaues:  dict_values(['Chetan', 'Naga', 'Male', 24, 'Unmarried', ['Technical skills', 'Analytical skills', 'Management skill
s'], 'India', 'Kavali', '11-33-948/2 , VengalaRao Nagar , 524201'])
```

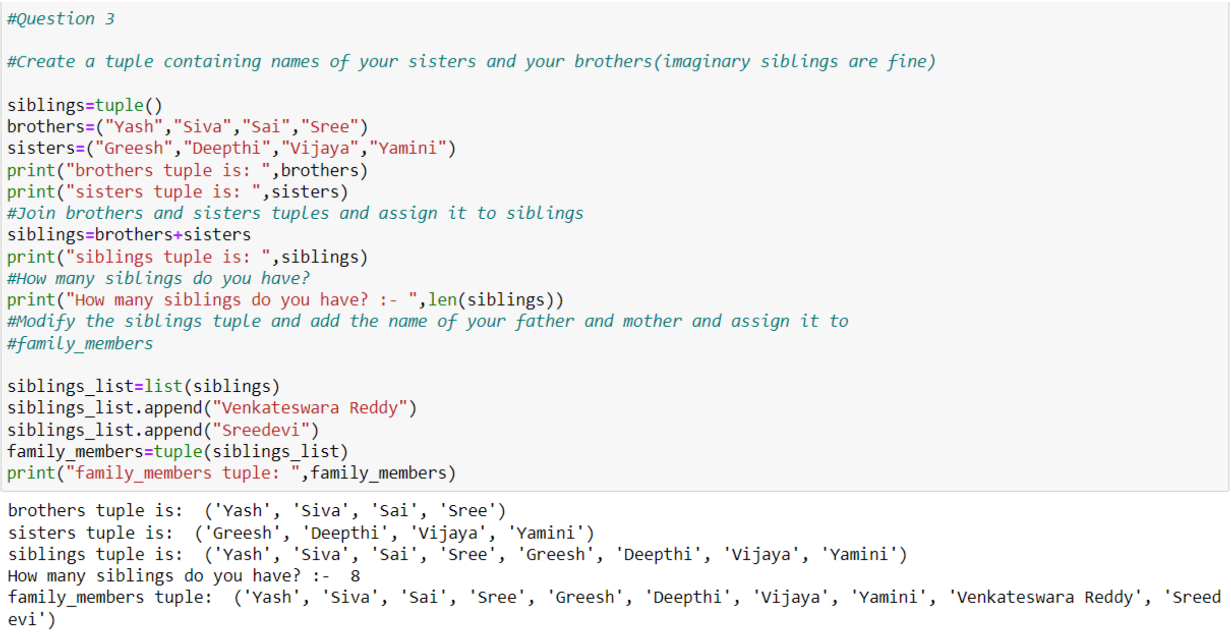
Description:

I created a dictionary using dict() method and added key-value pairs. I accessed the required values from the dictionary using their respective keys. I used keys() and values() built-in methods to access all the keys and values separately as a list.

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 Screenshot:



```
#Question 3

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

siblings=tuple()
brothers=("Yash","Siva","Sai","Sree")
sisters=("Greesh","Deepthi","Vijaya","Yamini")
print("brothers tuple is: ",brothers)
print("sisters tuple is: ",sisters)
#Join brothers and sisters tuples and assign it to siblings
siblings=brothers+sisters
print("siblings tuple is: ",siblings)
#How many siblings do you have?
print("How many siblings do you have? :- ",len(siblings))
#Modify the siblings tuple and add the name of your father and mother and assign it to
#family_members

siblings_list=list(siblings)
siblings_list.append("Venkateswara Reddy")
siblings_list.append("Sreedevi")
family_members=tuple(siblings_list)
print("family_members tuple: ",family_members)
```

```
brothers tuple is:  ('Yash', 'Siva', 'Sai', 'Sree')
sisters tuple is:  ('Greesh', 'Deepthi', 'Vijaya', 'Yamini')
siblings tuple is:  ('Yash', 'Siva', 'Sai', 'Sree', 'Greesh', 'Deepthi', 'Vijaya', 'Yamini')
How many siblings do you have? :-  8
family_members tuple:  ('Yash', 'Siva', 'Sai', 'Sree', 'Greesh', 'Deepthi', 'Vijaya', 'Yamini', 'Venkateswara Reddy', 'Sreed
evi')
```

Description:

I have created tuple using tuple() method and joined two tuples using ‘+’ symbol. As tuples are immutable I have converted the siblings tuple into list using list() method and modified it as lists can be changeable.

Question 4:

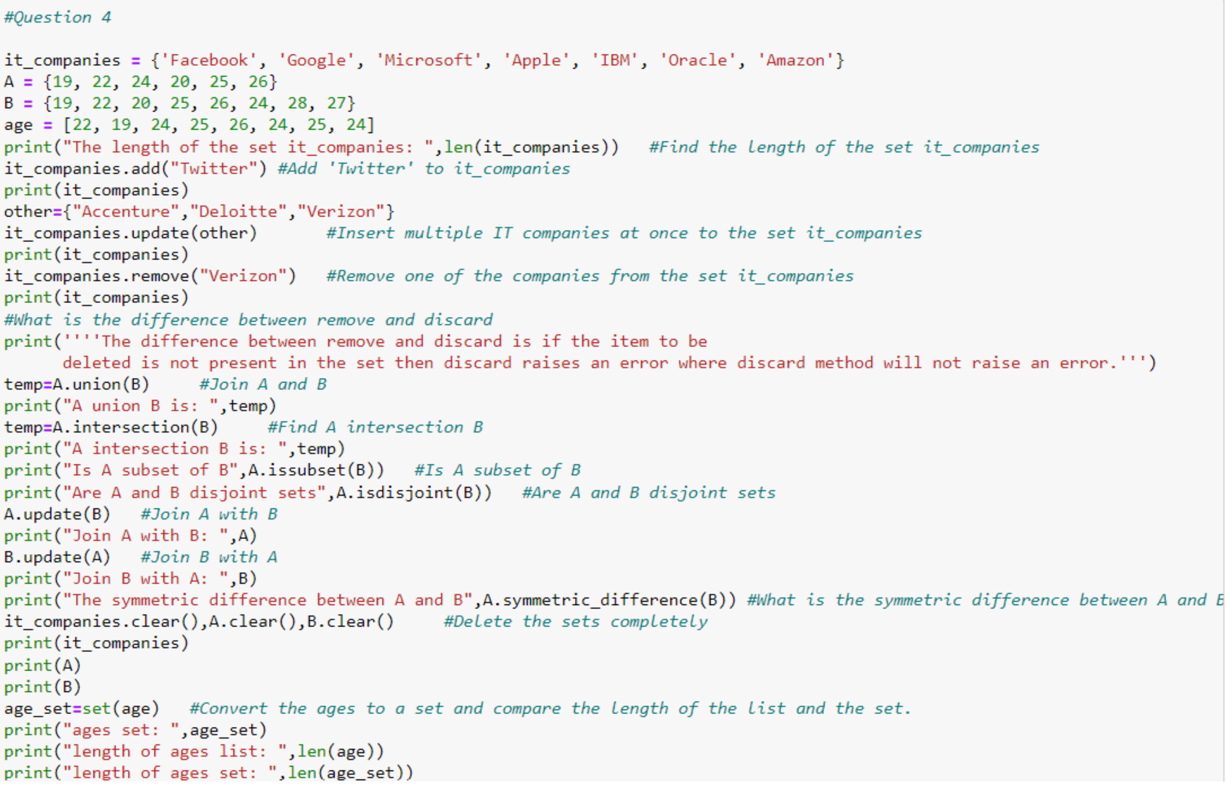
it_companies = {'Facebook', 'Google', 'Microsoft', 'Apple', 'IBM', 'Oracle', 'Amazon'}

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

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

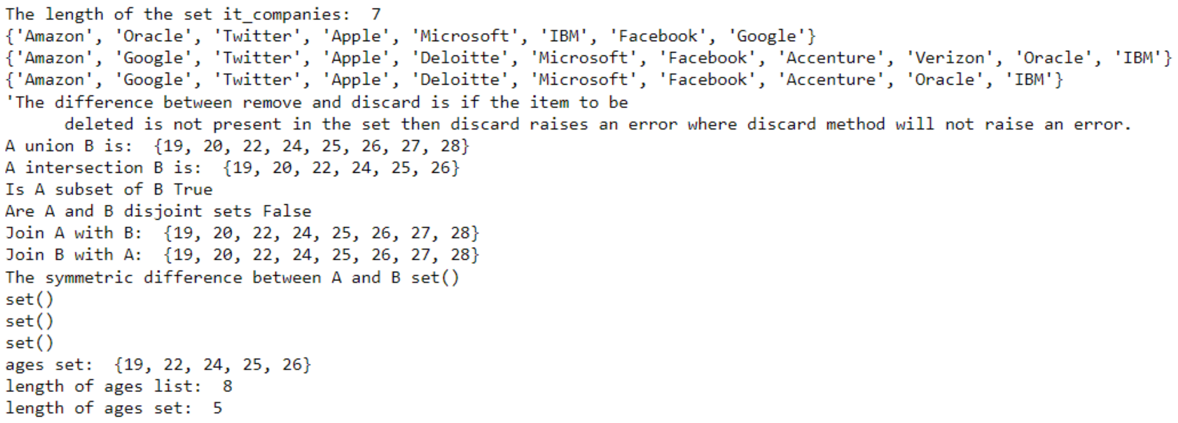
age = [22, 19, 24, 25, 26, 24, 25, 24]

- **Find the length of the set it_companies**
- **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.**



```
#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]
print("The length of the set it_companies: ",len(it_companies))   #Find the length of the set it_companies
it_companies.add("Twitter") #Add 'Twitter' to it_companies
print(it_companies)
other={"Accenture","Deloitte","Verizon"}
it_companies.update(other)       #Insert multiple IT companies at once to the set it_companies
print(it_companies)
it_companies.remove("Verizon")   #Remove one of the companies from the set it_companies
print(it_companies)
#What is the difference between remove and discard
print('''The difference between remove and discard is if the item to be
      deleted is not present in the set then discard raises an error where discard method will not raise an error.''')
temp=A.union(B)     #Join A and B
print("A union B is: ",temp)
temp=A.intersection(B)     #Find A intersection B
print("A intersection B is: ",temp)
print("Is A subset of B",A.issubset(B))   #Is A subset of B
print("Are A and B disjoint sets",A.isdisjoint(B))   #Are A and B disjoint sets
A.update(B)   #Join A with B
print("Join A with B: ",A)
B.update(A)   #Join B with A
print("Join B with A: ",B)
print("The symmetric difference between A and B",A.symmetric_difference(B)) #What is the symmetric difference between A and B
it_companies.clear(),A.clear(),B.clear()     #Delete the sets completely
print(it_companies)
print(A)
print(B)
age_set=set(age)   #Convert the ages to a set and compare the length of the list and the set.
print("ages set: ",age_set)
print("length of ages list: ",len(age))
print("length of ages set: ",len(age_set))
```



```
The length of the set it_companies:  7
{'Amazon', 'Oracle', 'Twitter', 'Apple', 'Microsoft', 'IBM', 'Facebook', 'Google'}
{'Amazon', 'Google', 'Twitter', 'Apple', 'Deloitte', 'Microsoft', 'Facebook', 'Accenture', 'Verizon', 'Oracle', 'IBM'}
{'Amazon', 'Google', 'Twitter', 'Apple', 'Deloitte', 'Microsoft', 'Facebook', 'Accenture', 'Oracle', 'IBM'}
'The difference between remove and discard is if the item to be
      deleted is not present in the set then discard raises an error where discard method will not raise an error.
A union B is:  {19, 20, 22, 24, 25, 26, 27, 28}
A intersection B is:  {19, 20, 22, 24, 25, 26}
Is A subset of B True
Are A and B disjoint sets False
Join A with B:  {19, 20, 22, 24, 25, 26, 27, 28}
Join B with A:  {19, 20, 22, 24, 25, 26, 27, 28}
The symmetric difference between A and B set()
set()
set()
set()
ages set:  {19, 22, 24, 25, 26}
length of ages list:  8
length of ages set:  5
```

Description:

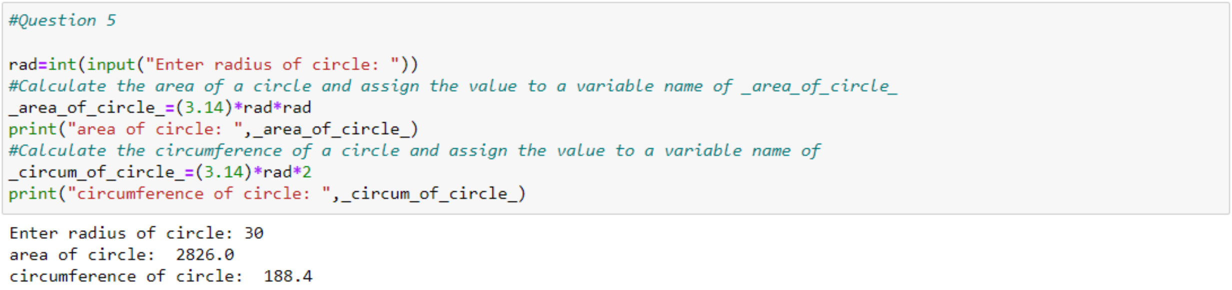
I added an item into the set using add() method and multiple items using the update() method. I also used the following set methods union(), intersection(), subset(), disjoint(), and symmetric_difference() in order to perform their specific operations on given sets. I deleted the entire set using the clear() method.

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 Screenshot:



```
#Question 5

rad=int(input("Enter radius of circle: "))
#Calculate the area of a circle and assign the value to a variable name of _area_of_circle_
_area_of_circle_=(3.14)*rad*rad
print("area of circle: ",_area_of_circle_)
#Calculate the circumference of a circle and assign the value to a variable name of
_circum_of_circle_=(3.14)*rad*2
print("circumference of circle: ",_circum_of_circle_)
```

```
Enter radius of circle: 30
area of circle:  2826.0
circumference of circle:  188.4
```

Description:

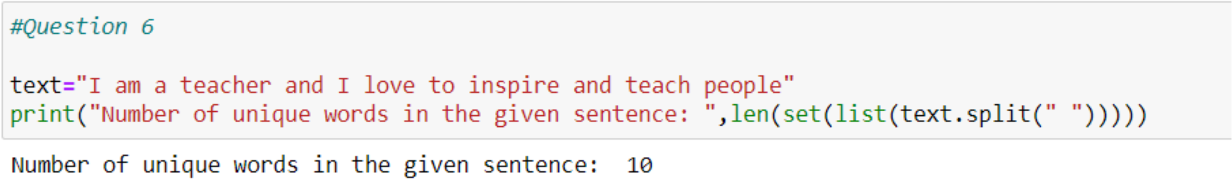
We can take input from the user using the input() method and then convert the stringformatted number to an integer using the int() method. I calculated the area and circumference of the circle using formulas πr^2 and $2\pi r$.

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

Solution Screenshot:



```
#Question 6

text="I am a teacher and I love to inspire and teach people"
print("Number of unique words in the given sentence: ",len(set(list(text.split(" ")))))
```

```
Number of unique words in the given sentence:  10
```

Description:

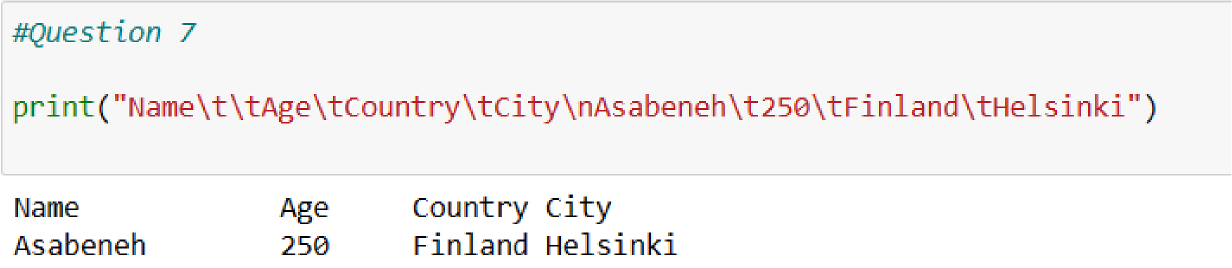
To split the sentence split() method is used which splits using white space by default. Then I converted the list of words to a set to get unique words as a set cannot containduplicated items in it.

Question 7:

Use a tab escape sequence to get the following lines.

Name	**Age**	**Country City**
Asabeneh	**250**	**Finland Helsinki**

Solution Screenshot:



```
#Question 7

print("Name\t\tAge\tCountry\tCity\nAsabeneh\t250\tFinland\tHelsinki")
```

```
Name            Age     Country City
Asabeneh        250     Finland Helsinki
```

Description:

I used ‘\t’ escape sequence to create a tab space in between the text and ‘\n’ to movethe text to the next line.

Question 8:

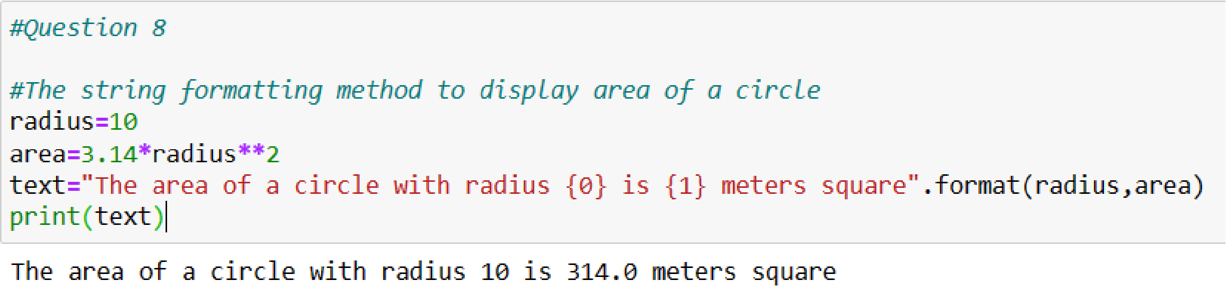
Use the string formatting method to display the following:

radius = 10

area = 3.14 * radius ** 2

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

Solution Screenshot:



```
#Question 8

#The string formatting method to display area of a circle
radius=10
area=3.14*radius**2
text="The area of a circle with radius {0} is {1} meters square".format(radius,area)
print(text)
```

```
The area of a circle with radius 10 is 314.0 meters square
```

Description:

I used πr^2 formula to get the area of a circle and used format() method to format the text.

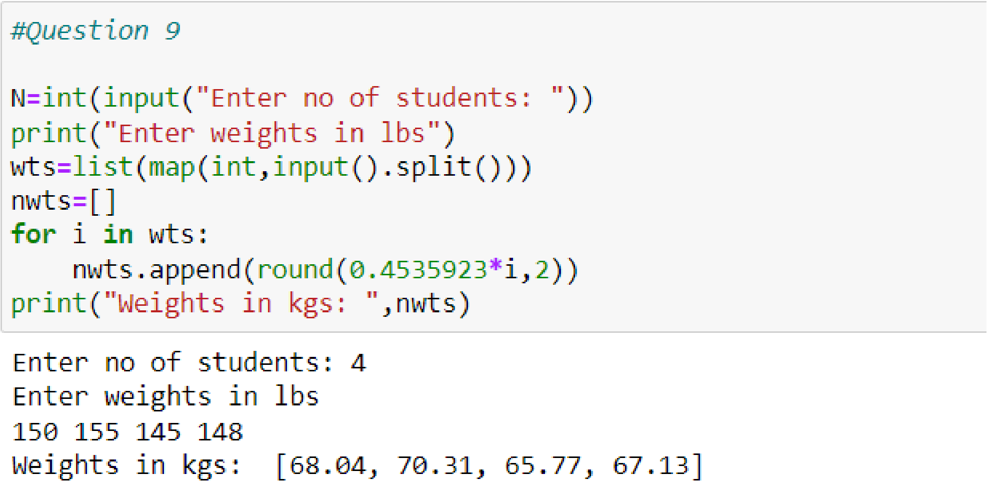
Question 9:

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 Screenshot:



```
#Question 9

N=int(input("Enter no of students: "))
print("Enter weights in lbs")
wts=list(map(int,input().split()))
nwts=[]
for i in wts:
    nwts.append(round(0.4535923*i,2))
print("Weights in kgs: ",nwts)
```

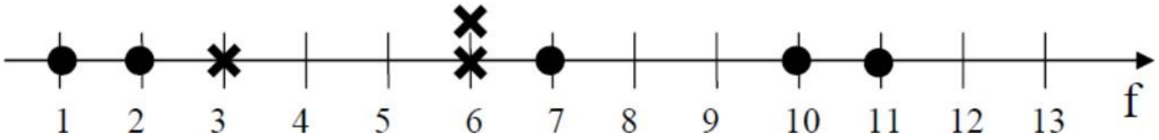
```
Enter no of students: 4
Enter weights in lbs
150 155 145 148
Weights in kgs:  [68.04, 70.31, 65.77, 67.13]
```

Description:

Using the append() method I added the converted weights to a new list by loopingthrough for.

Question 10

The diagram below shows a dataset with 2 classes and 8 data points, each with only one feature value, labeled f. Note that there are two data points with the same feature value of 6. These are shown as two x's one above the other.



1. Divide this data equally into two parts. Use first part as training and second part as testing. Using KNN classifier, for K=3, what would be the predicted outputs for the test samples? Show how you arrived at your answer.

2. Compute the confusion matrix for this and calculate accuracy, sensitivity, and specificity values.

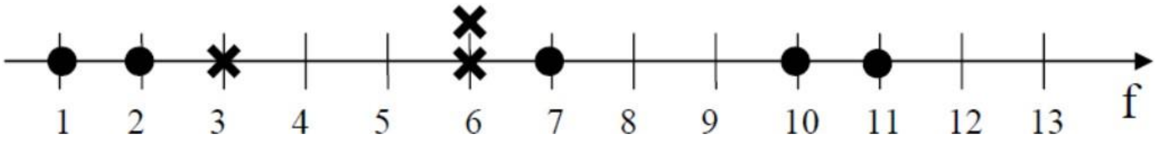
Solution:

1) The following are the data points from the above diagram given.

Data Point	**Class**
[1,0]	●
[2,0]	●
[3,0]	X
[6,0]	X
[6,0]	X
[7,0]	●
[10,0]	●
[11,0]	●

Randomly divide the above data points into two equal parts one is the training dataset and the another is the test dataset.

Training Dataset		**Test Dataset**	
Data point	**Class**	**Data point**	**Class**
[1,0]	●	[11,0]	●
[7,0]	●	[3,0]	X
[6,0]	X	[6,0]	X
[2,0]	●	[10,0]	●



Now, using the KNN classifier, for K=3 let us predict the class for the test dataset using the training dataset. For K=3 we must consider the three nearest data points for the taken sample point. By calculating the Euclidian distance between the points, we have to predict the class of the sample point.Euclidian Distance between two points (x1, x2) and (y1,y2) is $\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$

For [10,0] :- to this point the nearest three test points are

Data point	**Class**
[7,0]	●
[6,0]	X
[2,0]	●

For [11,0] :- to this point the nearest three test points are

Data point	Class
[7,0]	●
[6,0]	X
[2,0]	●

For [6,0] :- to this point the nearest three test points are

Data point	Class
[6,0]	X
[7,0]	●
[2,0]	●

For [3,0] :- to this point the nearest three test points are

Data point	Class
[2,0]	●
[1,0]	●
[6,0]	X

- Finally, the predicted classes for test dataset are

Data point	Actual Class	Predicted Class
[11,0]	●	●
[3,0]	X	●
[6,0]	X	●
[10,0]	●	●

2) The confusion matrix for the above data

	Actual values		
Predicted values		**X**	●
	X	**TP=0**	**FP=2**
	●	**FN=0**	**TN=2**

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN}$$

= 0+2/0+2+0+2

= 2/4

Accuracy = 0.5

$$Sensitivity = \frac{TP}{TP + FN}$$

= 0/0+0

Sensitivity = 0

$$Specificity = \frac{TN}{TN + FP}$$

= 2/2+2

= 2/4

Specificity – 0.5

Github Link: https://github.com/ChetanNaga/Machine-Learning