Machine Learning - Assignment1

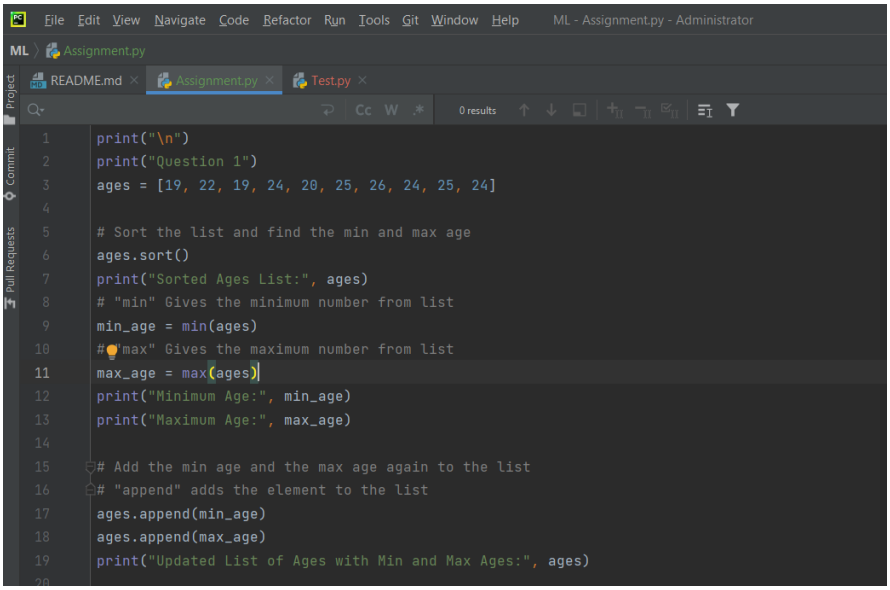
Name: **Idavalapati Vijay Taraka Ramarao**

ID: **700742485**

CRN: **13428**

Question1:

Given a list of “ages”, and perform sort operation, and get minimum and maximum element from the list and add the max and min back to the list.




```python
print("\n")
print("Question 1")
ages = [19, 22, 19, 24, 20, 25, 26, 24, 25, 24]

# Sort the list and find the min and max age
ages.sort()
print("Sorted Ages List:", ages)
# "min" Gives the minimum number from list
min_age = min(ages)
#"max" Gives the maximum number from list
max_age = max(ages)
print("Minimum Age:", min_age)
print("Maximum Age:", max_age)

# Add the min age and the max age again to the list
# "append" adds the element to the list
ages.append(min_age)
ages.append(max_age)
print("Updated List of Ages with Min and Max Ages:", ages)
```

Sort the ages list again, and calculate the median, average and range of the elements for the ages list

```python
ages.sort()
# Find the median age (one middle item or two middle items divided by two)
# Get the length of the list and Divide the length by 2
# if reminder is 0, Then get the middle and middle -1 element and divide by 2
# if reminder is other than 0, Then get the middle element and divide by 2
lst_Length = len(ages)
print("Ages List Length:", lst_Length)
num = lst_Length // 2
if lst_Length % 2 == 0:
    middlenumber = (ages[num] + ages[num - 1]) / 2
else:
    middlenumber = ages[num]/2
print("Median Age Value:", middlenumber)

# Find the average age (sum of all items divided by their number)
# Iterate the element in ages and add it total counter-"sum-num" and then divide by the length of the list
sum_num = 0
for t in ages:
    sum_num = sum_num + t
avg_ages = sum_num / len(ages)
print("Average Age:", avg_ages)

# Find the range of the ages (max minus min)
print("Range of Ages:", max_age - min_age)
```

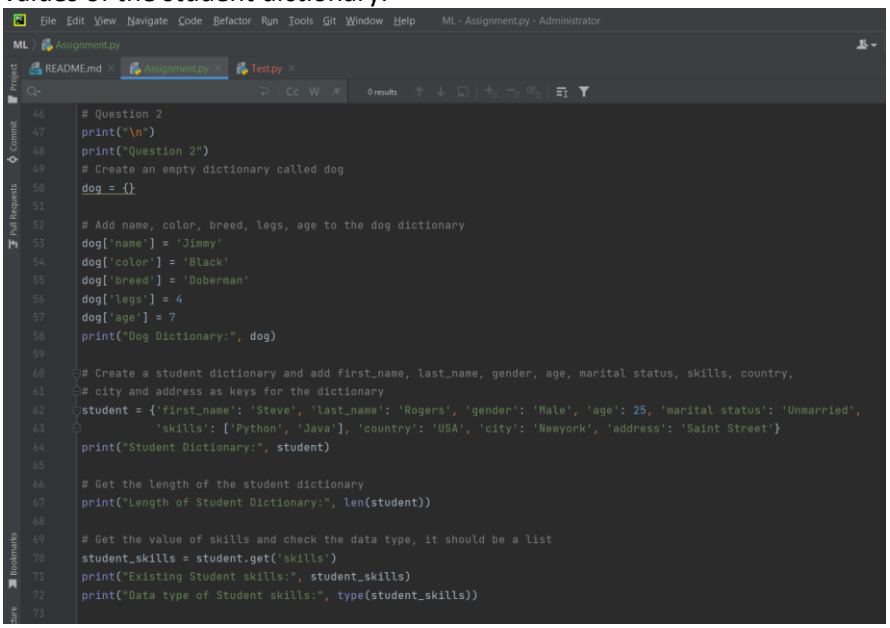
Question1 Output:

```
C:\Users\Administrator\Documents\GitHub\ML\venv\Scripts\python.exe C:/Users/Administrator/Documents/GitHub/ML/Assignment.py


Question 1
Sorted Ages List: [19, 19, 20, 22, 24, 24, 24, 25, 25, 26]
Minimum Age: 19
Maximum Age: 26
Updated List of Ages with Min and Max Ages: [19, 19, 20, 22, 24, 24, 24, 25, 25, 26, 19, 26]
Ages List Length: 12
Median Age Value: 24.0
Average Age: 22.75
Range of Ages: 7
```

Question2:

Create a empty dictionary of “dog”, and add ‘name’, ‘color’, ‘breed’, ‘legs’, ‘age’

Create a student dictionary with keys first_name, last_name, gender, age, marital status, skills, country, city and address and get the student skills and perform updating of skills and print the keys and values of the student dictionary.




```python
# Question 2
print("\n")
print("Question 2")
# Create an empty dictionary called dog
dog = {}

# Add name, color, breed, legs, age to the dog dictionary
dog['name'] = 'Jimmy'
dog['color'] = 'Black'
dog['breed'] = 'Doberman'
dog['legs'] = 4
dog['age'] = 7
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 = {'first_name': 'Steve', 'last_name': 'Rogers', 'gender': 'Male', 'age': 25, 'marital status': 'Unmarried',
           'skills': ['Python', 'Java'], 'country': 'USA', 'city': 'Newyork', 'address': 'Saint Street'}
print("Student Dictionary:", student)

# 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
student_skills = student.get('skills')
print("Existing Student skills:", student_skills)
print("Data type of Student skills:", type(student_skills))
```

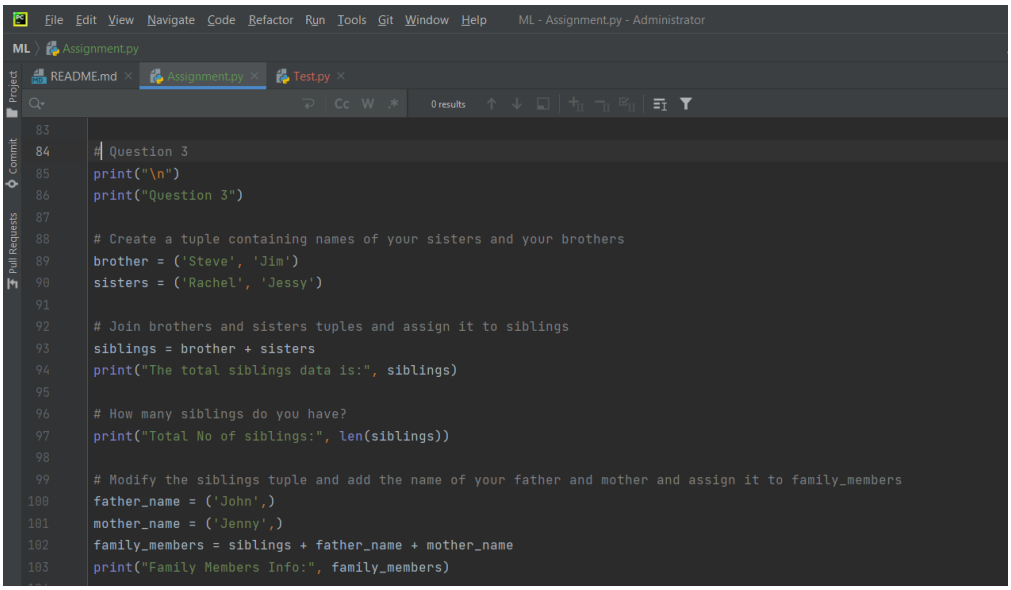
```python

# Modify the skills values by adding one or two skills
student.update({'skills': ['React, JavaScript']})
print("Update Student skills:", student.get('skills'))

# Get the dictionary keys as a list
print("Student Dictionary Keys:", student.keys())

# Get the dictionary values as a list
print("Student Dictionary Value:", student.values())
```

Question2 Output:

```
Question 2
Dog Dictionary: {'name': 'Jimmy', 'color': 'Black', 'breed': 'Doberman', 'legs': 4, 'age': 7}
Student Dictionary: {'first_name': 'Steve', 'last_name': 'Rogers', 'gender': 'Male', 'age': 25, 'marital status': 'Unmarried', 'skills': ['Python', 'Java'], 'country': 'USA',
 'city': 'Newyork', 'address': 'Saint Street'}
Length of Student Dictionary: 9
Existing Student skills: ['Python', 'Java']
Data type of Student skills: <class 'list'>
Update Student skills: ['React, JavaScript']
Student Dictionary Keys: dict_keys(['first_name', 'last_name', 'gender', 'age', 'marital status', 'skills', 'country', 'city', 'address'])
Student Dictionary Value: dict_values(['Steve', 'Rogers', 'Male', 25, 'Unmarried', ['React, JavaScript'], 'USA', 'Newyork', 'Saint Street'])
```

Question3:

Created a tuple containing names of your sisters and brothers and add it to sibling's tuple and the add father_name, mother_name to the family tuple



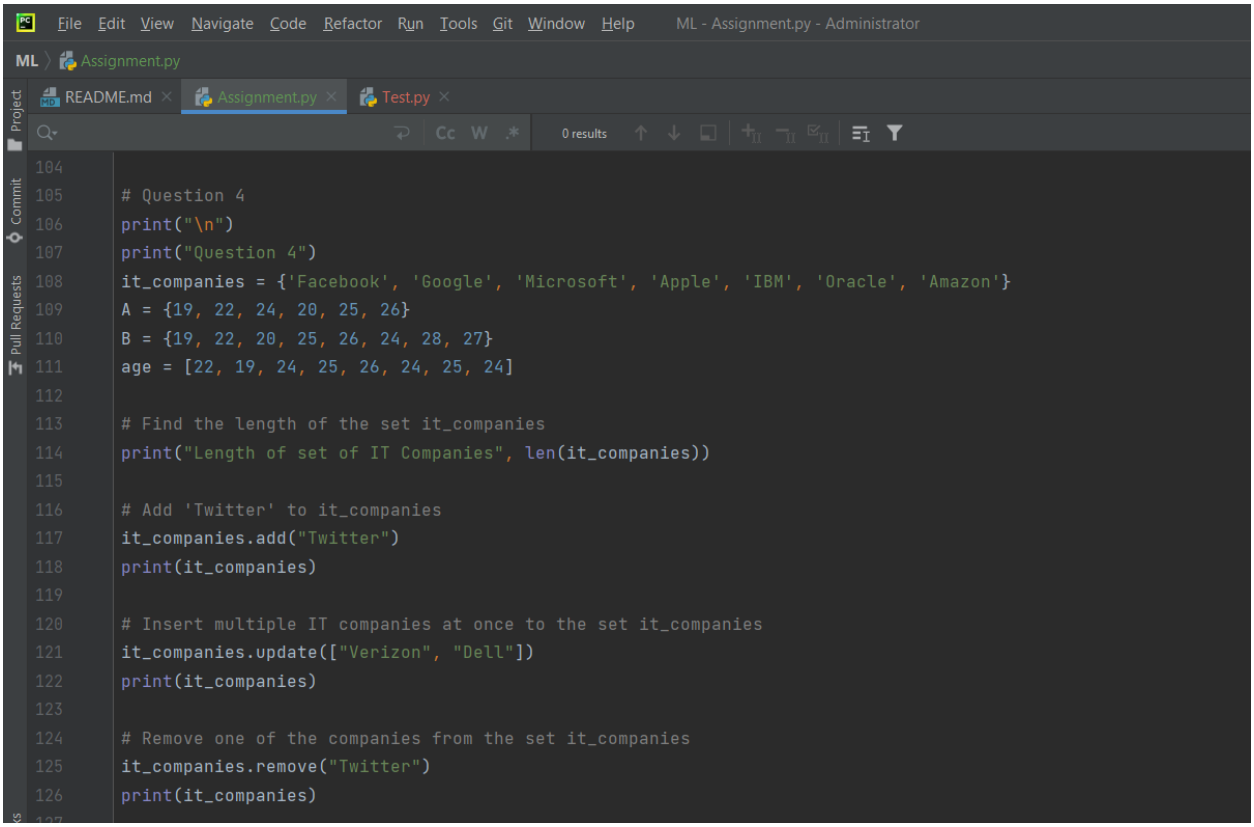
Question3 Output:

```
Question 3
The total siblings data is: ('Steve', 'Jim', 'Rachel', 'Jessy')
Total No of siblings: 4
Family Members Info: ('Steve', 'Jim', 'Rachel', 'Jessy', 'John', 'Jenny')
```

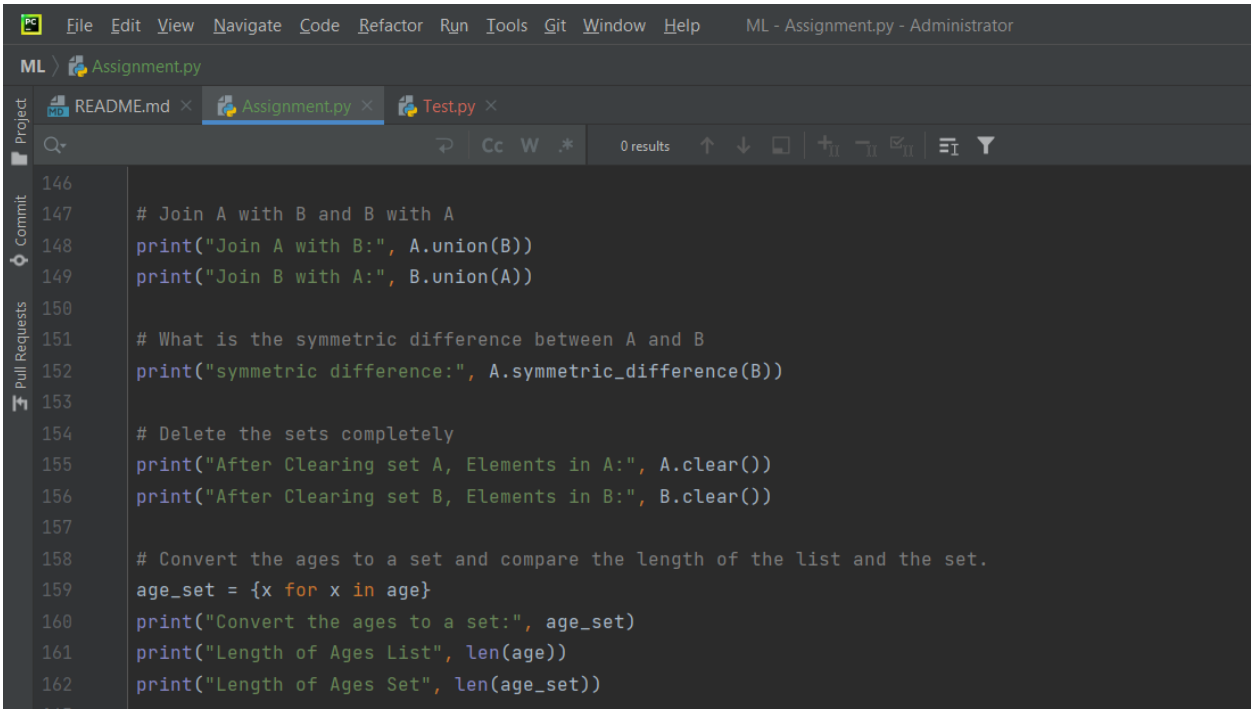
Question4:

Four set of Information given. And perform add, multiple insert, remove, opeations on “it_companies”

On Set A and Set B, Join, Intersection, and check operations of disjoint, subset in A & B



```
127
128 # What is the difference between remove and discard
129
130 # Both the Remove and discard methods are used to remove the specified item from the set
131 # by passing as argument to respective methods. The difference between those two are
132 # "Remove" method raises a KeyError exception
133 # "Discard" method does not raise an exception
134
135 # Join A and B
136 print("Union of A & B sets:", A | B)
137
138 # Find A intersection B
139 print("Intersection of A & B sets:", A & B)
140
141 # Is A subset of B
142 print("Is A subset of B:", A <= B)
143
144 # Are A and B disjoint sets
145 print("Are A and B disjoint sets:", A.isdisjoint(B))
146
```



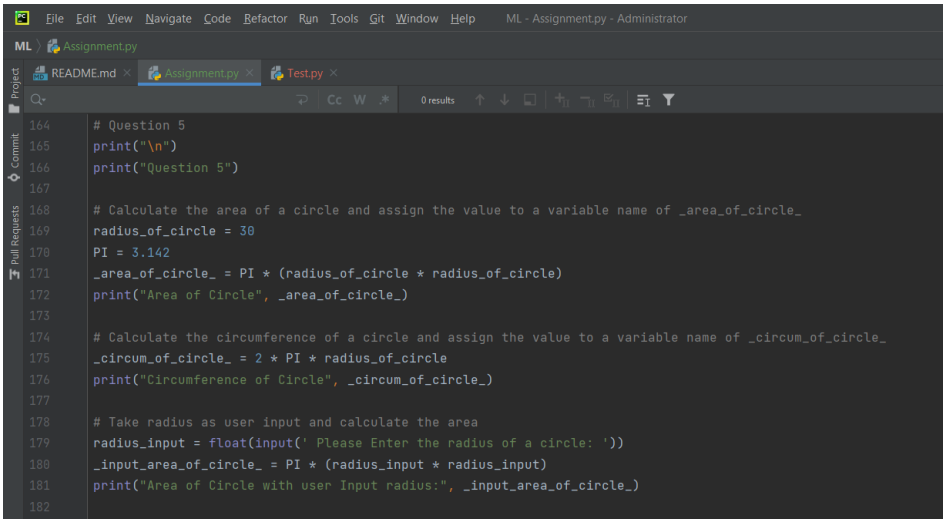

```
146
147 # Join A with B and B with A
148 print("Join A with B:", A.union(B))
149 print("Join B with A:", B.union(A))
150
151 # What is the symmetric difference between A and B
152 print("symmetric difference:", A.symmetric_difference(B))
153
154 # Delete the sets completely
155 print("After Clearing set A, Elements in A:", A.clear())
156 print("After Clearing set B, Elements in B:", B.clear())
157
158 # Convert the ages to a set and compare the length of the list and the set.
159 age_set = {x for x in age}
160 print("Convert the ages to a set:", age_set)
161 print("Length of Ages List", len(age))
162 print("Length of Ages Set", len(age_set))
```

Question4 Output:

```
Question 4
Length of set of IT Companies 7
{'Microsoft', 'Twitter', 'Facebook', 'Oracle', 'Apple', 'IBM', 'Amazon', 'Google'}
{'Microsoft', 'Twitter', 'Facebook', 'IBM', 'Oracle', 'Dell', 'Apple', 'Verizon', 'Amazon', 'Google'}
{'Microsoft', 'Facebook', 'IBM', 'Oracle', 'Dell', 'Apple', 'Verizon', 'Amazon', 'Google'}
Union of A & B sets: {19, 20, 22, 24, 25, 26, 27, 28}
Intersection of A & B sets: {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}
symmetric difference: {27, 28}
After Clearing set A, Elements in A: None
After Clearing set B, Elements in B: None
Convert the ages to a set: {19, 22, 24, 25, 26}
Length of Ages List 8
Length of Ages Set 5
```

Question5:

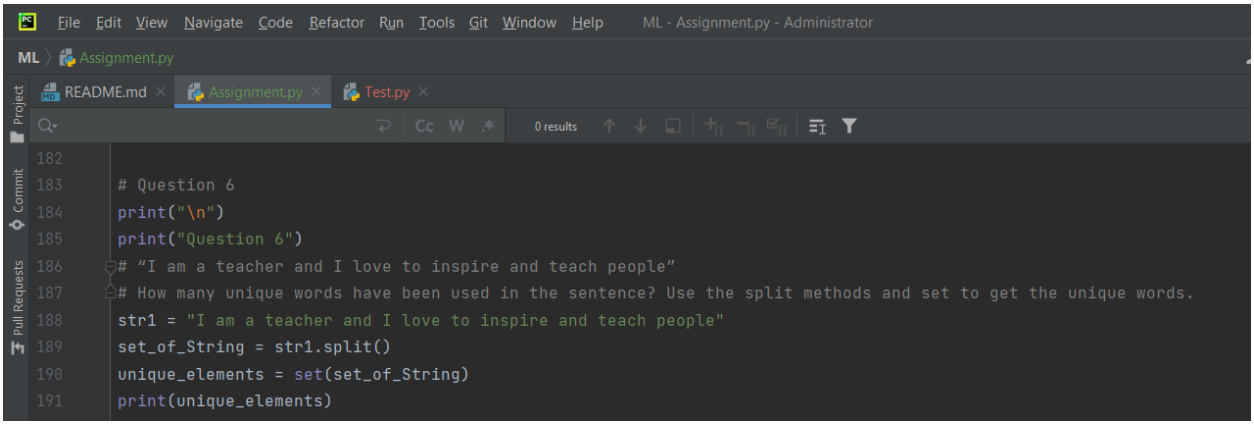
Calculate the area and circumference of circle using the defined radius and also radius from the user input.



Question5 Output:

```
Question 5
Area of Circle 2827.7999999999997
Circumference of Circle 188.51999999999998
 Please Enter the radius of a circle: 10
Area of Circle with user Input radius: 314.2
```

Question6:

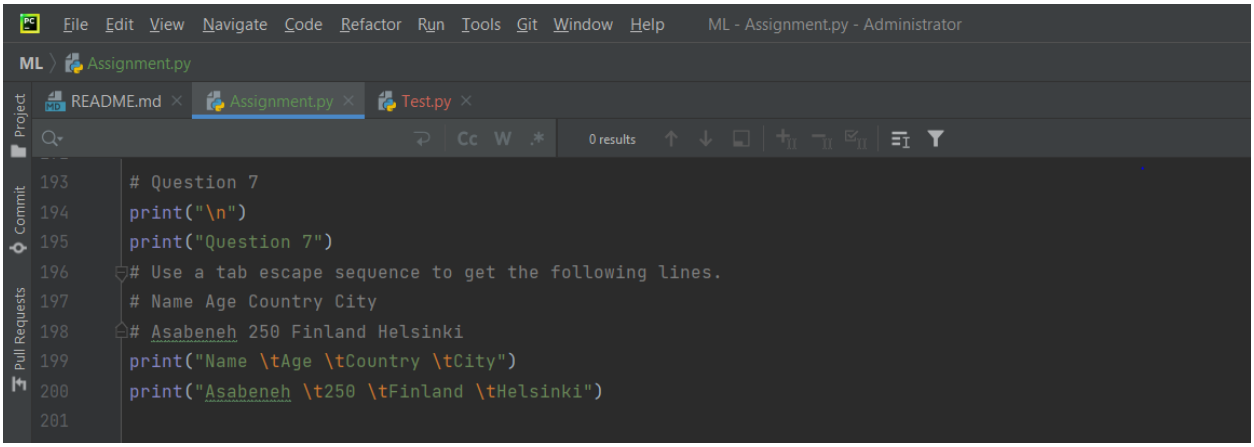


Question6 Output:

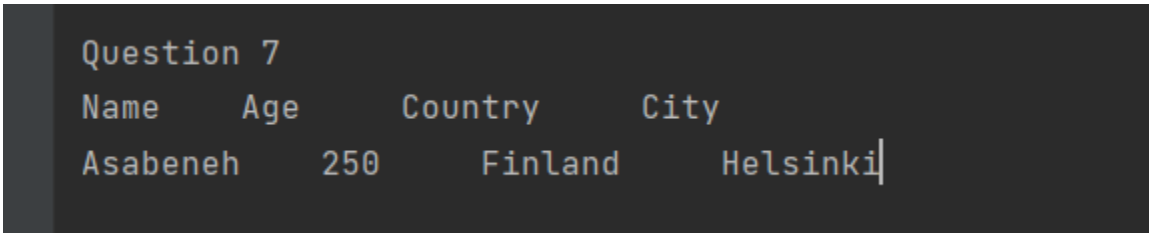
```
Question 6
{'am', 'and', 'people', 'a', 'teach', 'I', 'inspire', 'love', 'teacher', 'to'}
```

Question7:

For the sentence given, print the sentence with tab escape sequence

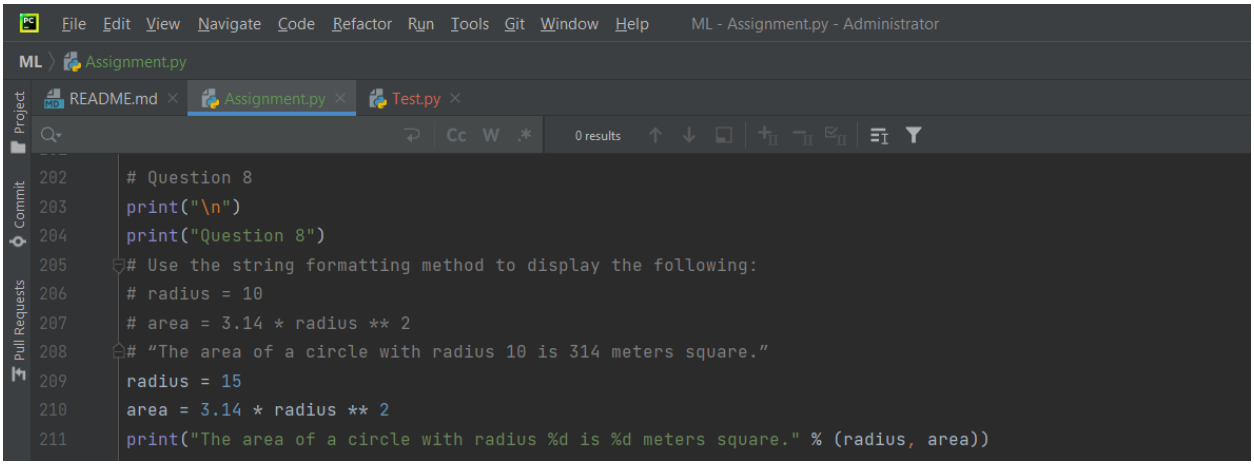


Question7 Output:

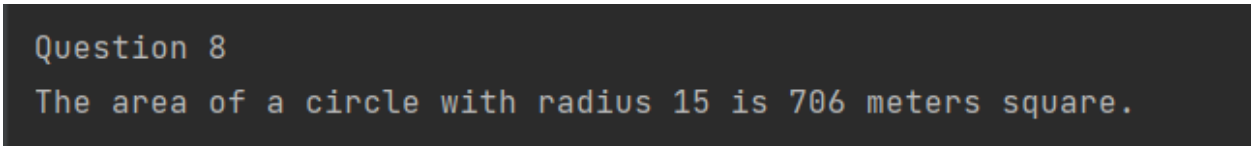


Question8:

Print the area of the circle in the specific format along with the radius

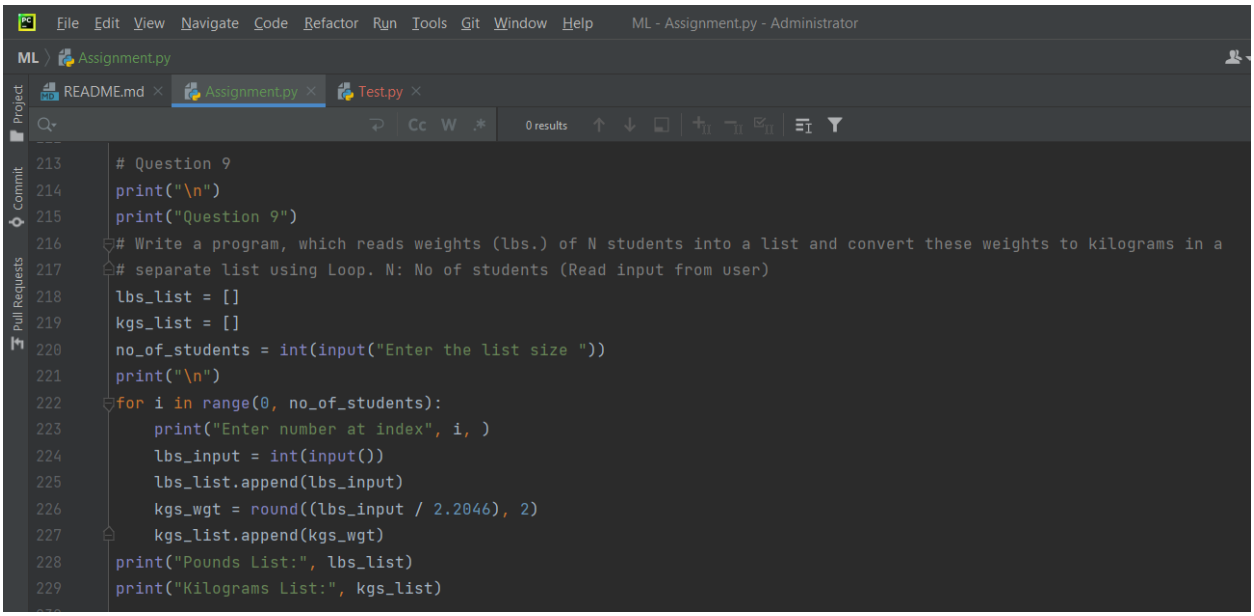


Question8 Output:



Question9:

Convert the list of “lbs” from the user input and then converting to “kilograms” by iterating the each element of lbs list and dividing by 2.2046 and printing the converted weight of kgs.



```
# Question 9
print("\n")
print("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)
lbs_list = []
kgs_list = []
no_of_students = int(input("Enter the list size "))
print("\n")
for i in range(0, no_of_students):
    print("Enter number at index", i, )
    lbs_input = int(input())
    lbs_list.append(lbs_input)
    kgs_wgt = round((lbs_input / 2.2046), 2)
    kgs_list.append(kgs_wgt)
print("Pounds List:", lbs_list)
print("Kilograms List:", kgs_list)
```

Question9 Output:

```
Question 9
Enter the list size 4


Enter number at index 0
150
Enter number at index 1
155
Enter number at index 2
145
Enter number at index 3
148
Pounds List: [150, 155, 145, 148]
Kilograms List: [68.04, 70.31, 65.77, 67.13]
```

Question10:

Question 10:-

The data is divided equally to two sets. They are

Training part :- 1, 2, 3, 10

Testing part :- 6, 6, 7, 11

Here $k = 3$

Euclidean distance, $d = \sqrt{(x - x_1)^2 + (y - y_1)^2}$

$\Rightarrow$ distance from 6 to 1, 2, 3, 10

$d_1 = \sqrt{(6-1)^2} = 5$

$d_2 = \sqrt{(6-2)^2} = 4$

$d_3 = \sqrt{(6-3)^2} = 3$

$d_4 = \sqrt{(6-10)^2} = 4$

} X

There are maximum no of dots (·)'s, 6 is changed from X to ·.

⇒ distance from 7 to 1, 2, 3, 10

$d_1 = \sqrt{(7-1)^2} = 6$

$d_2 = \sqrt{(7-2)^2} = 5$ } .

$d_3 = \sqrt{(7-3)^2} = 4$ } X

$d_4 = \sqrt{(7-10)^2} = 3$ } .

There are maximum no. of dots (.)'s, 10 is remains same as (.)

⇒ Distance from 11 to 1, 2, 3, 10

$d_1 = \sqrt{(11-1)^2} = 10$

$d_2 = \sqrt{(11-2)^2} = 9$ } .

$d_3 = \sqrt{(11-3)^2} = 8$ } X

$d_4 = \sqrt{(11-10)^2} = 1$ } .

There are maximum no. of dots (.)'s, so 11 remains same as (.)

2) Confusion matrix :-

	0	1
0	TN (0)	FP (2)
1	FN (0)	TP (2)

Accuracy :- $\frac{TP+TN}{P+N} = \frac{2+0}{2+2} = \frac{2}{4}$

$= 0.5$

Sensitivity :- $\frac{TP}{TP+FN} = \frac{2}{2+0} = 1$

Specificity :- $\frac{TN}{FP+TN} = \frac{0}{2+0} = 0$

Related Links:

Source Code:

https://github.com/VijayTarakaRamarao/ML/blob/main/Assignment.py

Video Recording:

https://github.com/VijayTarakaRamarao/ML/blob/main/ML_Assignment_Recording.mp4