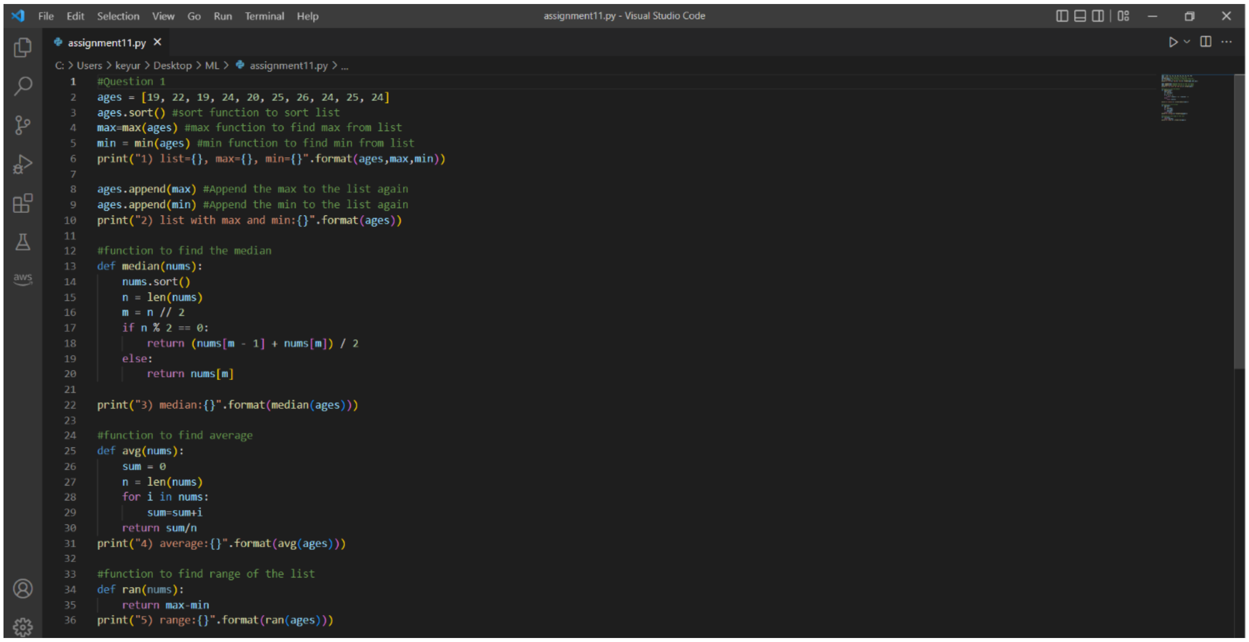
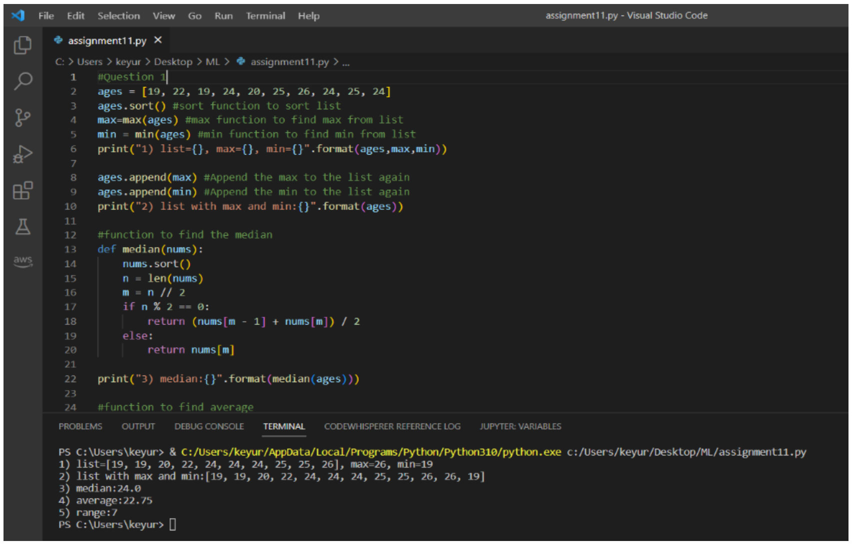
Video: https://drive.google.com/file/d/1wyOsi8-CBi1NnDcQFSfakwBE3U1V9t0X/view?usp=sharing


File Edit Selection View Go Run Terminal Help
assignment11.py - Visual Studio Code
assignment11.py
C: > Users > keyur > Desktop > ML > assignment11.py > ...

```python
1   #Question 1
2   ages = [19, 22, 19, 24, 20, 25, 26, 24, 25, 24]
3   ages.sort() #sort function to sort list
4   max=max(ages) #max function to find max from list
5   min = min(ages) #min function to find min from list
6   print("1) list={}, max={}, min={}".format(ages,max,min))
7
8   ages.append(max) #Append the max to the list again
9   ages.append(min) #Append the min to the list again
10  print("2) list with max and min:{}".format(ages))
11
12  #function to find the median
13  def median(nums):
14      nums.sort()
15      n = len(nums)
16      m = n // 2
17      if n % 2 == 0:
18          return (nums[m - 1] + nums[m]) / 2
19      else:
20          return nums[m]
21
22  print("3) median:{}".format(median(ages)))
23
24  #function to find average
25  def avg(nums):
26      sum = 0
27      n = len(nums)
28      for i in nums:
29          sum=sum+i
30      return sum/n
31  print("4) average:{}".format(avg(ages)))
32
33  #function to find range of the list
34  def ran(nums):
35      return max-min
36  print("5) range:{}".format(ran(ages)))
```

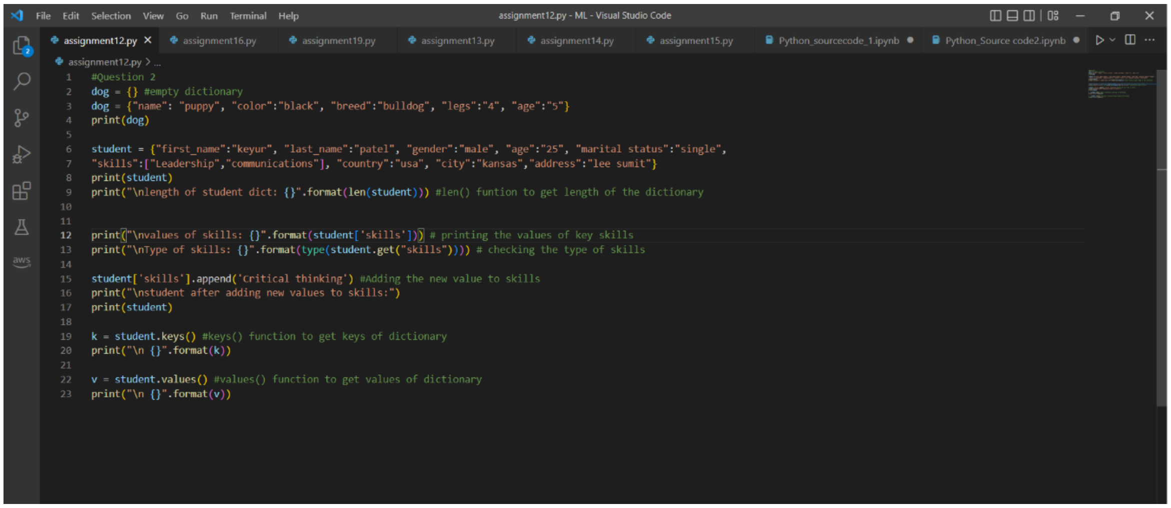
- **Question1)** first sort () function is used to sort the list (ages). Max () and Min() is used to find max and from the list. Then max and min of the list is printed. In line 8 and 9 Append function added the max and min to the list ages.
  In line 13 there is a function named median is created to find the median from the list. this function takes one parameter. This function first sorts the list then find the length of the list and then checks if list has even or odd element. And returns the median of list.
  In line 25 there is a function named age () which takes one parameter as argument. This function first computes the sum of all the element of list and divide it by the total number of list and returns the result.
  In line 34 function ran () finds the range of the list by using max-min formula.


File Edit Selection View Go Run Terminal Help
assignment11.py - Visual Studio Code
assignment11.py
C: > Users > keyur > Desktop > ML > assignment11.py > ...
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
CODEWHISPERER REFERENCE LOG
JUPYTER: VARIABLES

```python
1   #Question 1
2   ages = [19, 22, 19, 24, 20, 25, 26, 24, 25, 24]
3   ages.sort() #sort function to sort list
4   max=max(ages) #max function to find max from list
5   min = min(ages) #min function to find min from list
6   print("1) list={}, max={}, min={}".format(ages,max,min))
7
8   ages.append(max) #Append the max to the list again
9   ages.append(min) #Append the min to the list again
10  print("2) list with max and min:{}".format(ages))
11
12  #function to find the median
13  def median(nums):
14      nums.sort()
15      n = len(nums)
16      m = n // 2
17      if n % 2 == 0:
18          return (nums[m - 1] + nums[m]) / 2
19      else:
20          return nums[m]
21
22  print("3) median:{}".format(median(ages)))
23
24  #function to find average
```

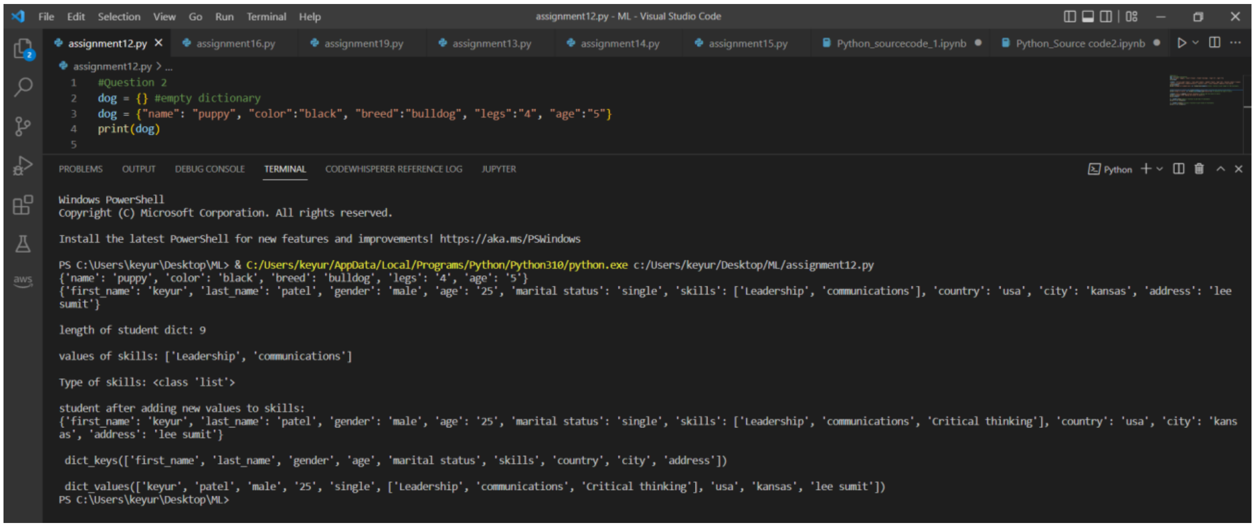
```
PS C:\Users\keyur> & C:/Users/keyur/AppData/Local/Programs/Python/Python310/python.exe c:/Users/keyur/Desktop/ML/assignment11.py
1) list=[19, 19, 20, 22, 24, 24, 24, 25, 25, 26], max=26, min=19
2) list with max and min:[19, 19, 20, 22, 24, 24, 24, 25, 25, 26, 26, 19]
3) median:24.0
4) average:22.75
5) range:7
PS C:\Users\keyur> 
```

- Above picture is the output of the first questions.



```
1   #Question 2
2   dog = {} #empty dictionary
3   dog = {"name": "puppy", "color":"black", "breed":"bulldog", "legs":"4", "age":"5"}
4   print(dog)
5
6   student = {"first_name":"keyur", "last_name":"patel", "gender":"male", "age":"25", "marital status":"single",
7   "skills":["Leadership","communications"], "country":"usa", "city":"kansas","address":"lee sumit"}
8   print(student)
9   print("\nlength of student dict: {}".format(len(student))) #len() funtion to get length of the dictionary
10
11
12  print("\nvalues of skills: {}".format(student['skills'])) # printing the values of key skills
13  print("\nType of skills: {}".format(type(student.get("skills")))) # checking the type of skills
14
15  student['skills'].append('Critical thinking') #Adding the new value to skills
16  print("\nstudent after adding new values to skills:")
17  print(student)
18
19  k = student.keys() #keys() function to get keys of dictionary
20  print("\n {}".format(k))
21
22  v = student.values() #values() function to get values of dictionary
23  print("\n {}".format(v))
```

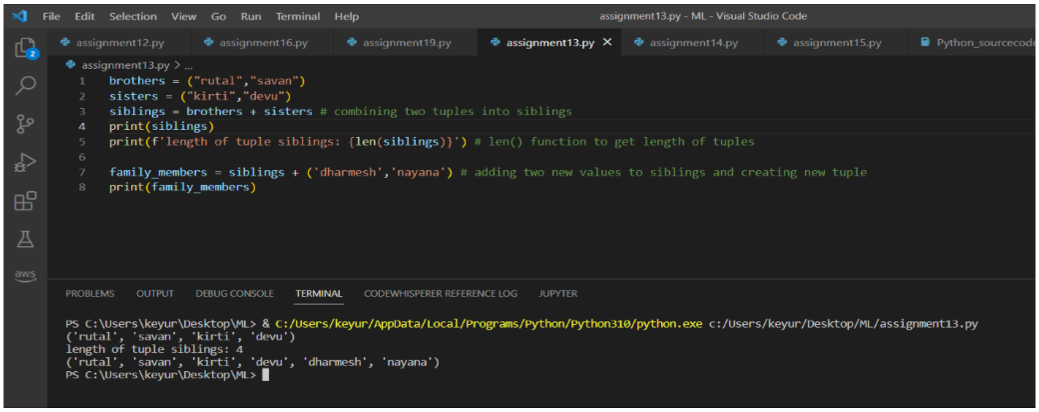
- **Question 2)** first empty dictionary is created. Then name,color,breed,legs,age keys with values are added in the empty dictionary dog.
  In line 6 student named dictionary is created. And len () function is used to get the length of dictionary student.
  To get the values of ‘skills’ key student(skills) is used. And to get the type of the values type () function is used.
  .append is used to add new values to existing key – ‘skills’
  To get the all the keys key () is used.
  To get the values values() is used and to print .format is used to print string and variable.



```
1   #Question 2
2   dog = {} #empty dictionary
3   dog = {"name": "puppy", "color":"black", "breed":"bulldog", "legs":"4", "age":"5"}
4   print(dog)
5
```

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\keyur\Desktop\ML> & C:/Users/keyur/AppData/Local/Programs/Python/Python310/python.exe c:/Users/keyur/Desktop/ML/assignment12.py
{'name': 'puppy', 'color': 'black', 'breed': 'bulldog', 'legs': '4', 'age': '5'}
{'first_name': 'keyur', 'last_name': 'patel', 'gender': 'male', 'age': '25', 'marital status': 'single', 'skills': ['Leadership', 'communications'], 'country': 'usa', 'city': 'kansas', 'address': 'lee
sumit'}

length of student dict: 9

values of skills: ['Leadership', 'communications']

Type of skills: <class 'list'>

student after adding new values to skills:
{'first_name': 'keyur', 'last_name': 'patel', 'gender': 'male', 'age': '25', 'marital status': 'single', 'skills': ['Leadership', 'communications', 'Critical thinking'], 'country': 'usa', 'city': 'kans
as', 'address': 'lee sumit'}

 dict_keys(['first_name', 'last_name', 'gender', 'age', 'marital status', 'skills', 'country', 'city', 'address'])

 dict_values(['keyur', 'patel', 'male', '25', 'single', ['Leadership', 'communications', 'Critical thinking'], 'usa', 'kansas', 'lee sumit'])
PS C:\Users\keyur\Desktop\ML>
```

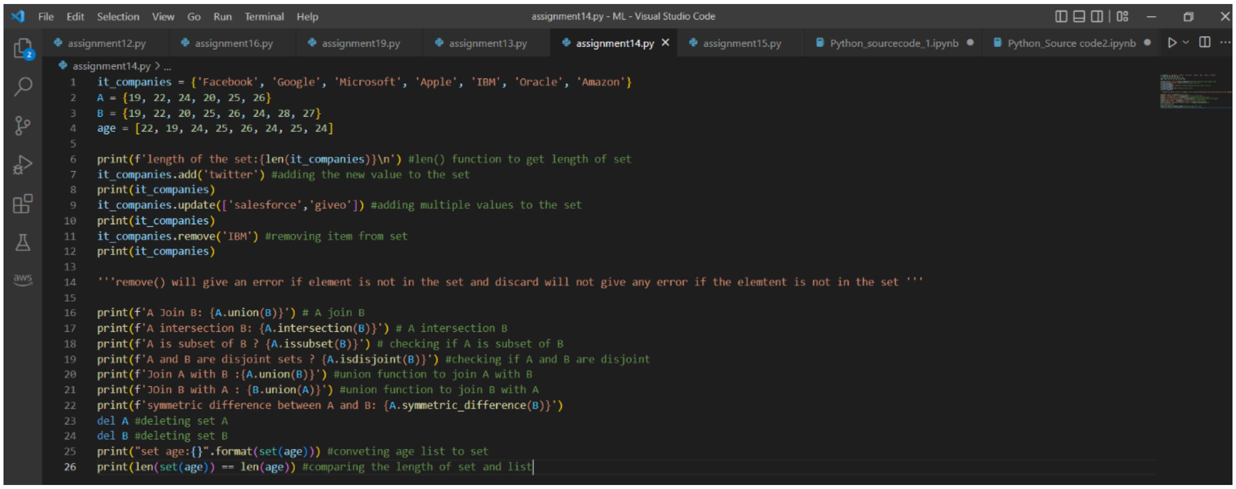
- Above picture is the output of the 2nd question.



```python
brothers = ("rutal","savan")
sisters = ("kirti","devu")
siblings = brothers + sisters # combining two tuples into siblings
print(siblings)
print(f'length of tuple siblings: {len(siblings)}') # len() function to get length of tuples

family_members = siblings + ('dharmesh','nayana') # adding two new values to siblings and creating new tuple
print(family_members)
```

```
PS C:\Users\keyur\Desktop\ML> & C:/Users/keyur/AppData/Local/Programs/Python/Python310/python.exe c:/Users/keyur/Desktop/ML/assignment13.py
('rutal', 'savan', 'kirti', 'devu')
length of tuple siblings: 4
('rutal', 'savan', 'kirti', 'devu', 'dharmesh', 'nayana')
PS C:\Users\keyur\Desktop\ML>
```

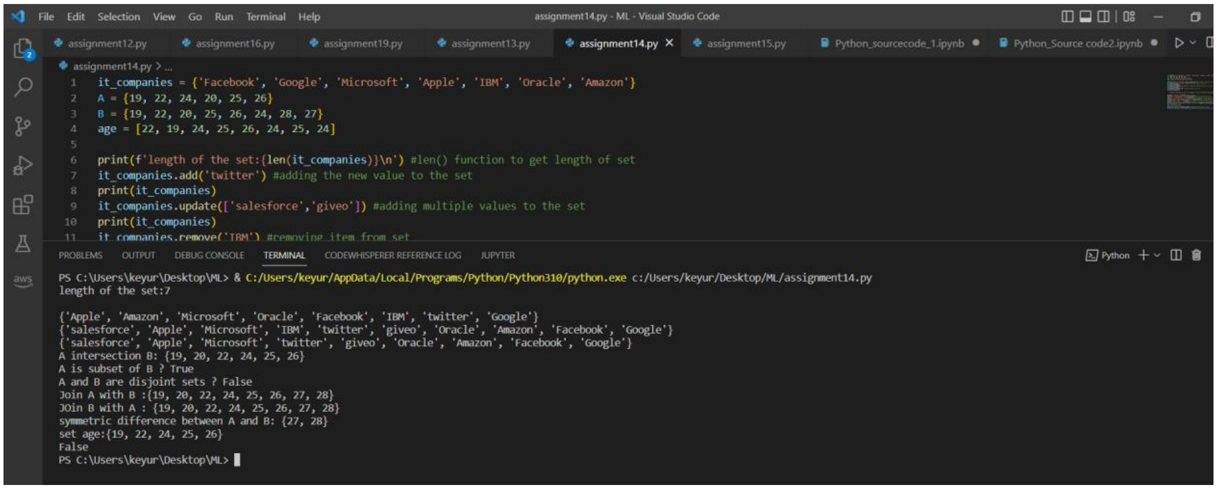
- Question 3) Tuples named brothers and sisters are created and combined them into single tuple siblings. To get the length of tuples len () is used.
  To add father and mother name and assign is to new tuple ‘+’ operator is used. Tuples are immutable so we cannot modify once we create the tuple.
  There is a output also in the above picture.


File Edit Selection View Go Run Terminal Help
assignment14.py - ML - Visual Studio Code
assignment12.py
assignment16.py
assignment19.py
assignment13.py
assignment14.py
assignment15.py
Python_sourcecode_1.ipynb
Python_Source code2.ipynb
assignment14.py > ...

```python
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(f'length of the set:{len(it_companies)}\n') #len() function to get length of set
it_companies.add('twitter') #adding the new value to the set
print(it_companies)
it_companies.update(['salesforce','giveo']) #adding multiple values to the set
print(it_companies)
it_companies.remove('IBM') #removing item from set
print(it_companies)

'''remove() will give an error if element is not in the set and discard will not give any error if the elemtent is not in the set '''

print(f'A Join B: {A.union(B)}') # A join B
print(f'A intersection B: {A.intersection(B)}') # A intersection B
print(f'A is subset of B ? {A.issubset(B)}') # checking if A is subset of B
print(f'A and B are disjoint sets ? {A.isdisjoint(B)}') #checking if A and B are disjoint
print(f'Join A with B :{A.union(B)}') #union function to join A with B
print(f'Join B with A : {B.union(A)}') #union function to join B with A
print(f'symmetric difference between A and B: {A.symmetric_difference(B)}')
del A #deleting set A
del B #deleting set B
print("set age:{}".format(set(age))) #conveting age list to set
print(len(set(age)) == len(age)) #comparing the length of set and list
```

- Question 4) To get the length of the set len () is used. And to add new element in the set add() is used. To add multiple element at a time update() is used. To remove remove() is used.
  There are two function remove and discard to remove the element from set. Diffrence is that if you used remove to remove element that is not present in the set it will give you an error while discard will not generate any error.
  To join two sets join () is used.
  Intersection () is used to get intersection of the two set.
  To check if a set is subset of another set issubset is used. It will give output true if the set is the subset of another set otherwise it will give false. Same with disjoint ().
  To know the symmetric difference between two sets symmetric_diffrence () is used. Symmetric difference means element present in set A and set B but not in both set.
  To delete a set del setname is used.

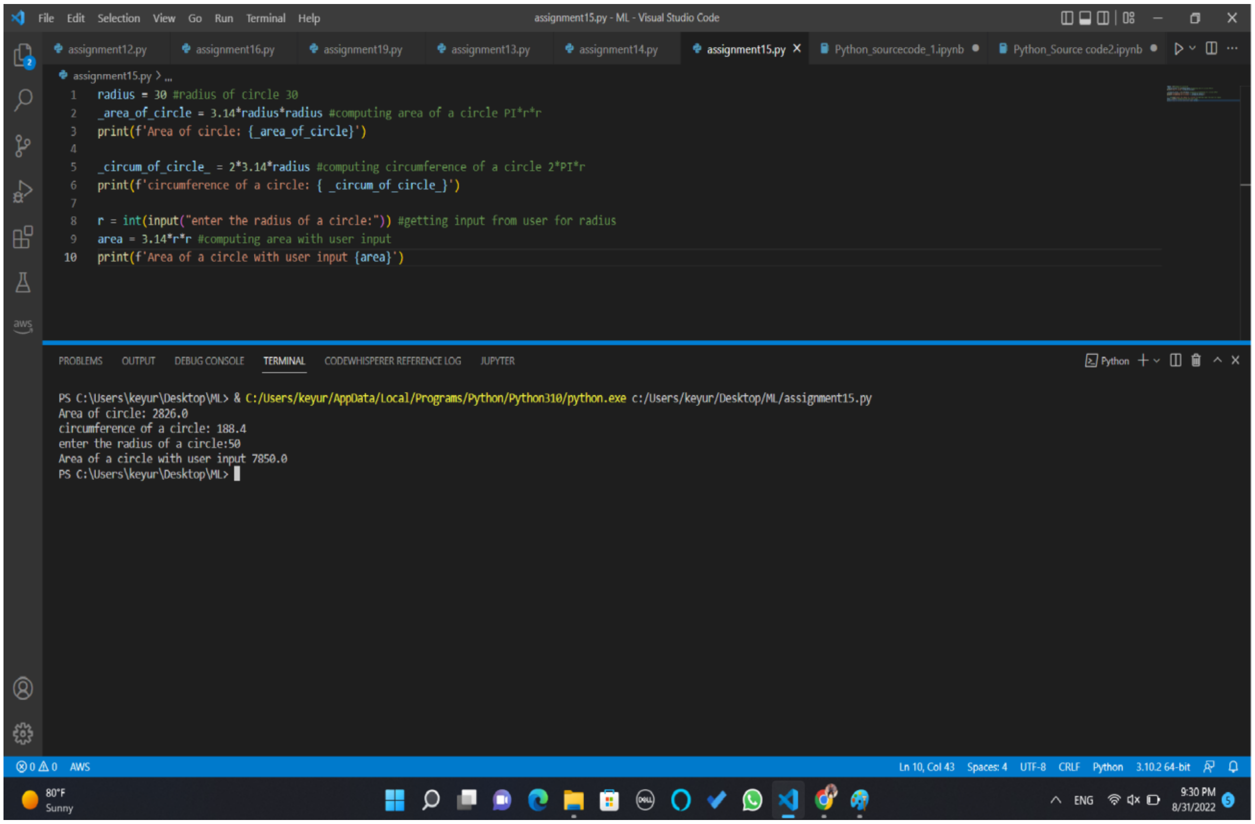
We can convert list into set by set () and it will remove all the duplicate values from the list as set does not allow duplicate values. To compare length of set and list '==' is used it gives false because length is different.



```python
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(f'length of the set:{len(it_companies)}\n') #len() function to get length of set
it_companies.add('twitter') #adding the new value to the set
print(it_companies)
it_companies.update(['salesforce','giveo']) #adding multiple values to the set
print(it_companies)
it_companies.remove('IBM') #removing item from set
```

```
PS C:\Users\keyur\Desktop\ML> & C:/Users/keyur/AppData/Local/Programs/Python/Python310/python.exe c:/Users/keyur/Desktop/ML/assignment14.py
length of the set:7

{'Apple', 'Amazon', 'Microsoft', 'Oracle', 'Facebook', 'IBM', 'twitter', 'Google'}
{'salesforce', 'Apple', 'Microsoft', 'IBM', 'twitter', 'giveo', 'Oracle', 'Amazon', 'Facebook', 'Google'}
{'salesforce', 'Apple', 'Microsoft', 'twitter', 'giveo', 'Oracle', 'Amazon', 'Facebook', 'Google'}
A intersection B: {19, 20, 22, 24, 25, 26}
A is subset of B ? True
A and B are 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 between A and B: {27, 28}
set age:{19, 22, 24, 25, 26}
False
PS C:\Users\keyur\Desktop\ML> 
```

- Above is the output of the question 4.

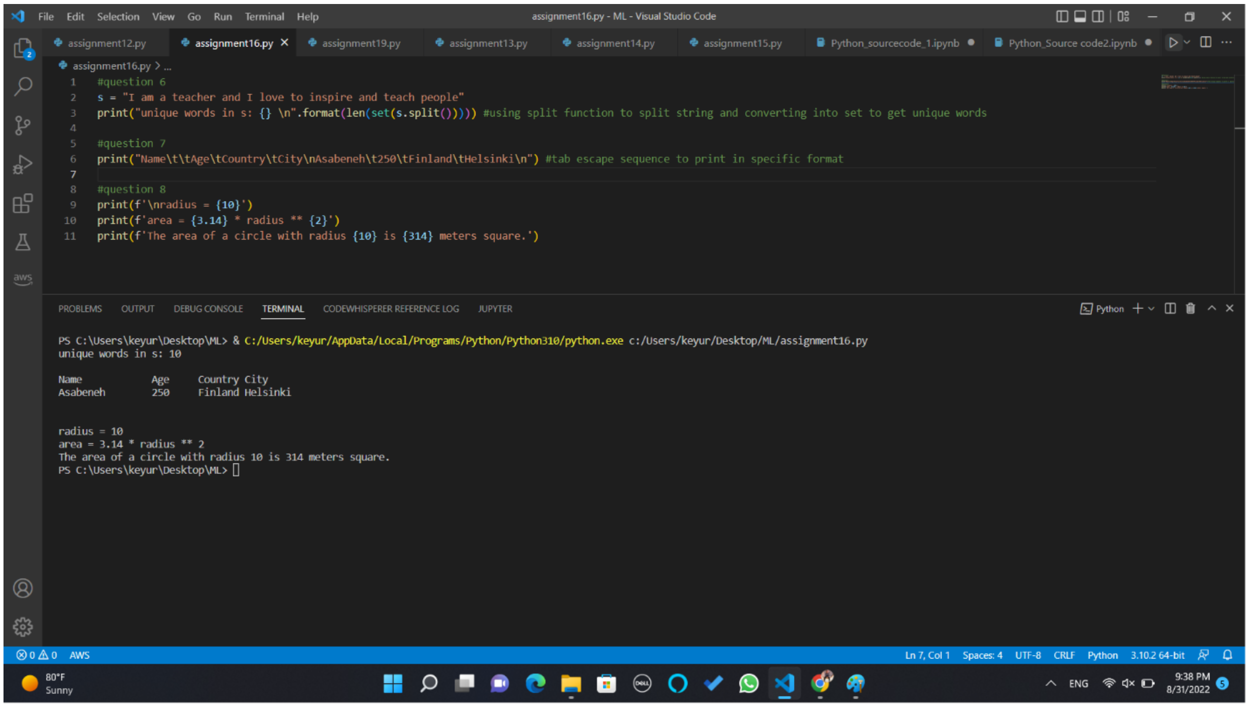


File Edit Selection View Go Run Terminal Help
assignment15.py - ML - Visual Studio Code
assignment12.py
assignment16.py
assignment19.py
assignment13.py
assignment14.py
assignment15.py
Python_sourcecode_1.ipynb
Python_Source code2.ipynb
assignment15.py > ...
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
CODEWHISPERER REFERENCE LOG
JUPYTER
Python
Ln 10, Col 43 Spaces: 4 UTF-8 CRLF Python 3.10.2 64-bit
AWS
80°F
Sunny
ENG
9:30 PM
8/31/2022

```python
radius = 30 #radius of circle 30
_area_of_circle = 3.14*radius*radius #computing area of a circle PI*r*r
print(f'Area of circle: {_area_of_circle}')

_circum_of_circle_ = 2*3.14*radius #computing circumference of a circle 2*PI*r
print(f'circumference of a circle: { _circum_of_circle_}')

r = int(input("enter the radius of a circle:")) #getting input from user for radius
area = 3.14*r*r #computing area with user input
print(f'Area of a circle with user input {area}')
```

```
PS C:\Users\keyur\Desktop\ML> & C:/Users/keyur/AppData/Local/Programs/Python/Python310/python.exe c:/Users/keyur/Desktop/ML/assignment15.py
Area of circle: 2826.0
circumference of a circle: 188.4
enter the radius of a circle:50
Area of a circle with user input 7850.0
PS C:\Users\keyur\Desktop\ML>
```

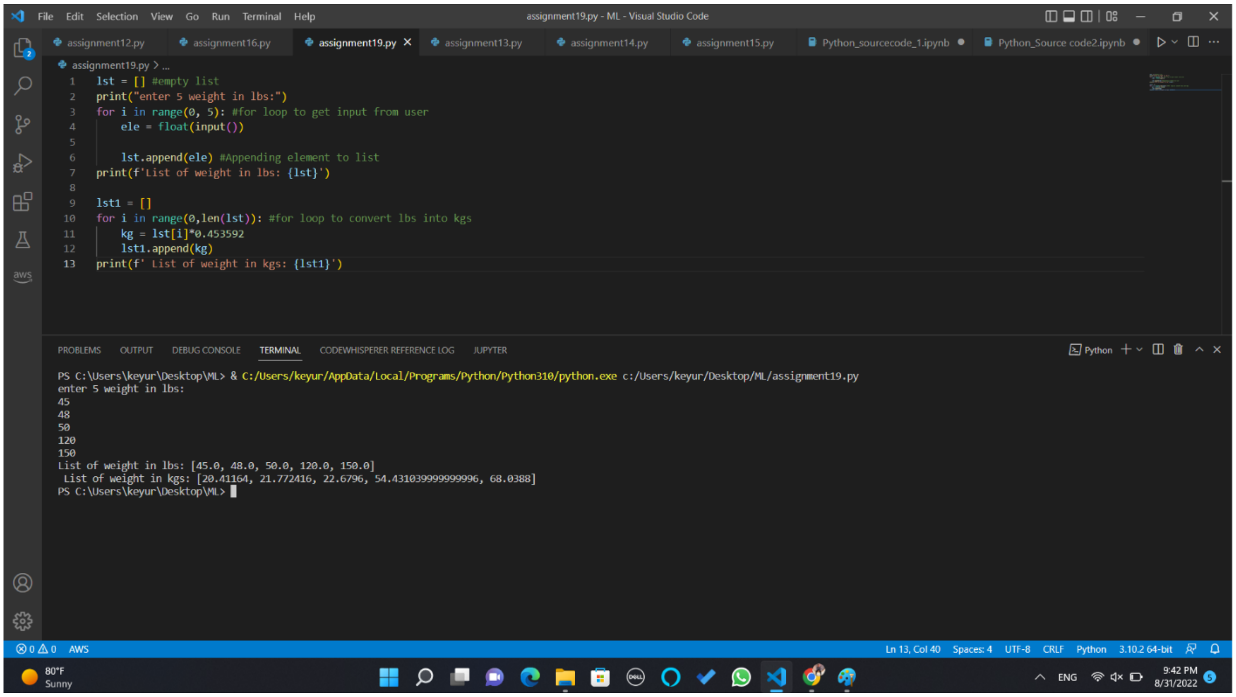
- Question 5) To compute the area of the circle PI*radius*radius is used. And the circumference of the circle is the 2*PI*radius.
  To get the input from user input() is used. After getting input from user it is stored in the r variable. which is used to compute the area of the circle. Output also included in the screenshot.



```python
#question 6
s = "I am a teacher and I love to inspire and teach people"
print("unique words in s: {} \n".format(len(set(s.split())))) #using split function to split string and converting into set to get unique words

#question 7
print("Name\t\tAge\tCountry\tCity\nAsabeneh\t250\tFinland\tHelsinki\n") #tab escape sequence to print in specific format

#question 8
print(f'\nradius = {10}')
print(f'area = {3.14} * radius ** {2}')
print(f'The area of a circle with radius {10} is {314} meters square.')
```

```
PS C:\Users\keyur\Desktop\ML> & C:/Users/keyur/AppData/Local/Programs/Python/Python310/python.exe c:/Users/keyur/Desktop/ML/assignment16.py
unique words in s: 10

Name            Age     Country City
Asabeneh        250     Finland Helsinki


radius = 10
area = 3.14 * radius ** 2
The area of a circle with radius 10 is 314 meters square.
PS C:\Users\keyur\Desktop\ML> 
```

- Question 6) there is a string given and the task is to count unique words. So first the split () is used split whole string into multiple words and then those element is added into the set. Set does not the allows duplicate values so only unique words are added into the set. And to get the total number of elemnent len () is used.
- Question 7) Task is to print data into tablar format with the help of tab escape sequences. Tab escape is ‘\t’ and for new line ‘\n’ is used.
- Question 8) Task is to print string and variable together. There are multiple ways to do that f-string is one of them. In f string variables are covered inside {}.
- Output for all questions 6,7,8 is also at the bottom of picture.



```python
lst = [] #empty list
print("enter 5 weight in lbs:")
for i in range(0, 5): #for loop to get input from user
    ele = float(input())

    lst.append(ele) #Appending element to list
print(f'List of weight in lbs: {lst}')

lst1 = []
for i in range(0,len(lst)): #for loop to convert lbs into kgs
    kg = lst[i]*0.453592
    lst1.append(kg)
print(f' List of weight in kgs: {lst1}')
```

```
PS C:\Users\keyur\Desktop\ML> & C:/Users/keyur/AppData/Local/Programs/Python/Python310/python.exe c:/Users/keyur/Desktop/ML/assignment19.py
enter 5 weight in lbs:
45
48
50
120
150
List of weight in lbs: [45.0, 48.0, 50.0, 120.0, 150.0]
 List of weight in kgs: [20.41164, 21.772416, 22.6796, 54.431039999999996, 68.0388]
PS C:\Users\keyur\Desktop\ML> 
```

- Question 9) task is to get user input for weight in lbs and convert it into kgs.
  First for loop is used to get user input 5 times with the input () and then added into the list using append ().
  Second for loop is used to access all element and convert lbs into kgs.

- Question 10)
  Training set first half of the set:

1	O
2	O
3	X
6	X

For KNN K=3 so we must take 3 nearest neighbor.

For dataset 6: nearest neighbors are 2:O,3:X and 6:X. So, prediction for 6 in testing set is X.
For dataset 7: nearest neighbors are 3:O, 6:X, 6:X so prediction for 7 is X.
For dataset 10: nearest neighbors are 6:X, 6:X, 7:X so prediction for 10 is X.
For dataset 11: nearest neighbors are 6:X, 7:X, 10:X so prediction for 11 is X

Testing set:

Dataset	Actual output	Predicted output	TP/TN/FP/FN
6	X	X	TN
7	O	X	FN
10	O	X	FN
11	O	X	FN

O: positive   X: negative
TP:0
TN:1
FP:0
FN:3

Accuracy:
(TP+TN)/(P+N)
= (0+1)/ (3+1)
= ¼

Sensitivity:
TP/P
=0

Specificity:
TN/N
=1/1
=1