

```
assignment1 > Question_1.py > ...
1 ages = [19, 22, 19, 24, 20, 25, 26, 24, 25, 24]
2 ages.sort() #sorts the ages array
3 Maxim=max(ages) #gives the maximum of ages
4 Minim=min(ages) #gives the minimum of ages
5 print("Maximum is ", Maxim)
6 print("Minimum is",Minim) #prints the Maximum and minimum
7 #finding the median of ages
8 length = len(ages)
9 if(length%2 ==0): #if the length of array is even, median is middle element
10     median = ages[length//2]
11
12 else: #if the length of array is odd, median of array is average of middle 2 elements
13     median = (ages[length//2]+ ages[(length//2) +1])/2
14
15 sumOfAges=sum(ages) # gives the sum of all the ages.
16 print("Average of ages is ",sumOfAges/len(ages)) #sum of ages divided by number of ages gives the average
17 range=Maxim-Minim # range = maximum - minimum
18 print("Range is ",range) #prints range
```

PROBLEMS OUTPUT DEBUG CONSOLE **TERMINAL** JUPYTER

```
bash - assignment1
('Range is ', 7)
(base) ponugotis-Air-7:assignment1 ponugotivamshi$ python3 Question_1.py
Maximum is 26
Minimum is 19
Average of ages is 22.8
Range is 7
(base) ponugotis-Air-7:assignment1 ponugotivamshi$
```

Ln 17, Col 39 Spaces: 4 UTF-8 CRLF Python 3.8.2 64-bit Go Live Prettier

Q1.

A) Given a list called ages and I sorted the list and found max and min of the given list.

Added min and max of ages list to itself again. Found Median of the list by finding whether the number of elements in the list is even or odd. Based on the result found the median as mentioned in the code above.

—> Found average of the list by adding all the elements of the list and divided with length of the list.

—> Found the range of ages with the formula Maximum - Minimum

Code and output are shown as above.

```
assignment1 > Question_2.py > ...
12 #adding values to student dictionary
13 Student["first_name"]='Vamshi'
14 Student["last_name"]='Ponugoti'
15 Student["gender"]='male'
16 Student["age"]=23
17 Student["marital_status"]='Single'
18 Student['skills']=['Java','Python',"React","Guitar"]
19 Student["country"]="India"
20 Student['city']="Hyderabad"
21 Student["Address"]="chinthakani"
22 print("Student dictionary contains",Student) #printing student dictionary
23 print("Length of student dictionary is ",len(Student)) #prints the length
24 print("data type of value of skills",type(Student['skills'])) #prints the type of values
25 Student['skills'].append('SAP') #adding SAP to student skills
26 print("list of keys are",list(Student.keys()))
27 print("list of values are",list(Student.values()))
```

PROBLEMS OUTPUT DEBUG CONSOLE **TERMINAL** JUPYTER

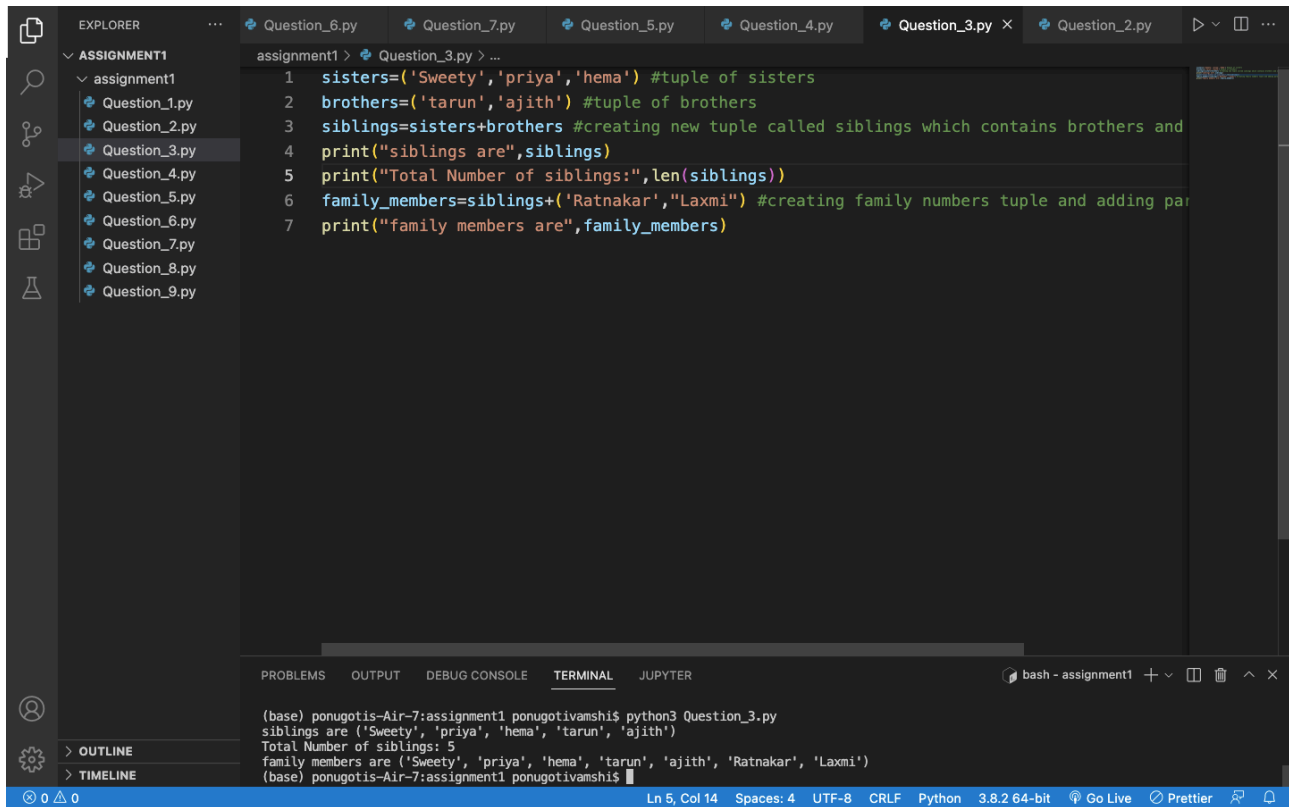
```
bash - assignment1
(base) ponugotis-Air-7:assignment1 ponugotivamshi$ python3 Question_2.py
dog dictionary contains {'name': 'kruger', 'color': 'black', 'breed': 'Husky', 'legs': 4, 'age': 5}
Student dictionary contains {'first_name': 'Vamshi', 'last_name': 'Ponugoti', 'gender': 'male', 'age': 23, 'marital_status': 'Single', 'skills': ['Java', 'Python', 'React', 'Guitar'], 'country': 'India', 'city': 'Hyderabad', 'Address': 'chinthakani'}
Length of student dictionary is 9
data type of value of skills <class 'list'>
list of keys are ['first_name', 'last_name', 'gender', 'age', 'marital_status', 'skills', 'country', 'city', 'Address']
list of values are ['Vamshi', 'Ponugoti', 'male', 23, 'Single', ['Java', 'Python', 'React', 'Guitar', 'SAP'], 'India', 'Hyderabad', 'chinthakani']
(base) ponugotis-Air-7:assignment1 ponugotivamshi$
```

Ln 20, Col 28 Spaces: 4 UTF-8 CRLF Python 3.8.2 64-bit Go Live Prettier

Q2. Created an empty dictionary called dog and added values to it like name, color, breed, legs, age

—>Created another dictionary named Student and added first_name, last_name, gender, age, marital status, skills, country, city and address to it. Found length of the dictionary . Using Student[skills] found the skills of the student. Modified the skills values by adding SAP to it. Printed the list of keys and values separately.

Code and it's output is as shown above.



The screenshot shows a VS Code editor with a file explorer on the left containing a folder named 'ASSIGNMENT1' with files 'Question_1.py' through 'Question_9.py'. The main editor window displays 'Question_3.py' with the following Python code:

```
1 sisters=('Sweety','priya','hema') #tuple of sisters
2 brothers=('tarun','ajith') #tuple of brothers
3 siblings=sisters+brothers #creating new tuple called siblings which contains brothers and
4 print("siblings are",siblings)
5 print("Total Number of siblings:",len(siblings))
6 family_members=siblings+('Ratnakar','Laxmi') #creating family numbers tuple and adding par
7 print("family members are",family_members)
```

The bottom panel shows the 'TERMINAL' output for the command 'python3 Question_3.py':

```
(base) ponugotis-Air-7:assignment1 ponugotivamshi$ python3 Question_3.py
siblings are ('Sweety', 'priya', 'hema', 'tarun', 'ajith')
Total Number of siblings: 5
family members are ('Sweety', 'priya', 'hema', 'tarun', 'ajith', 'Ratnakar', 'Laxmi')
```

Q3. Created a tuple with names of sisters and brothers. Joined brothers and sisters tuples to the new tuple called siblings. By using the length function found the number of siblings. Created a family_members tuple and added siblings tuple and added parents name to it.

Code and output as shown above.

```
Question_8.py Question_6.py Question_7.py Question_5.py Question_4.py x Question_3.py Question_2.py
assignment1 > Question_4.py > ...
1 #creating a new set of it companies.
2 it_companies = {'Facebook', 'Google', 'Microsoft', 'Apple', 'IBM', 'Oracle', 'Amazon'}
3 #creating a set A
4 A = {19, 22, 24, 20, 25, 26}
5 #creating a set B
6 B = {19, 22, 20, 25, 26, 24, 28, 27}
7 #creating a list of ages
8 age = [22, 19, 24, 25, 26, 24, 25, 24]
9 print("length of it_companies",len(it_companies))
10 it_companies.add('Twitter') #adding Twitter to the it_companies
11 print("it_companies are",it_companies)
12 it_companies.update({'Flipkart','Garuda',"Ford"}) #adding multiple it companies to actual
13 print("updated it companies are",it_companies)
14 it_companies.remove('Flipkart') # removing flipkart from the it_companies list.
15 print("Updated list:",it_companies)
16 #

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER
bash - assignment1 + - [ ] ^ x

(base) ponugotis-Air-7:assignment1 ponugotivamshi$ python3 Question_4.py
length of it_companies 7
it_companies are {'Amazon', 'Twitter', 'Oracle', 'IBM', 'Microsoft', 'Apple', 'Google', 'Facebook'}
updated it companies are {'Amazon', 'Twitter', 'Oracle', 'Garuda', 'Ford', 'IBM', 'Microsoft', 'Apple', 'Flipkart', 'Google', 'Facebook'}
Updated list: {'Amazon', 'Twitter', 'Oracle', 'Garuda', 'Ford', 'IBM', 'Microsoft', 'Apple', 'Google', 'Facebook'}
A union B is: {19, 20, 22, 24, 25, 26, 27, 28}
A intersection B is: {19, 20, 22, 24, 25, 26}
True
False
joining A with B {19, 20, 22, 24, 25, 26, 27, 28}
joining B with A {19, 20, 22, 24, 25, 26, 27, 28}
symmetric difference between A and B {27, 28}
None
None
None
None
set of ages: {19, 22, 24, 25, 26}
length of set of list is: 5
length of set is : 8
(base) ponugotis-Air-7:assignment1 ponugotivamshi$
```

```
Question_8.py Question_6.py Question_7.py Question_5.py Question_4.py x Question_3.py Question_2.py
assignment1 > Question_4.py > ...
15 print("Updated list:",it_companies)
16 #
17 print("A union B is:",A.union(B)) #prints A union B
18 print("A intersection B is: ",A.intersection(B)) #prints A intersection B
19 print(A.issubset(B)) #checking is A is subset of B
20 print(A.isdisjoint(B)) # cheking whether the A and B are disjoint sets
21 print("joining A with B",A.union(B))
22 print("joining B with A",B.union(A))
23 print("symmetric difference between A and B",A.symmetric_difference(B))
24 print(A.clear()) # deleting set A
25 print(B.clear()) # deleting set B
26 print(it_companies.clear()) #deleting set it_companies
27 print("set of ages:",set(age)) #creating set of ages
28 print("length of set of list is: ",len(set(age)))
29 print("length of set is :",len(age))

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER
bash - assignment1 + - [ ] ^ x

(base) ponugotis-Air-7:assignment1 ponugotivamshi$ python3 Question_4.py
length of it_companies 7
it_companies are {'Amazon', 'Twitter', 'Oracle', 'IBM', 'Microsoft', 'Apple', 'Google', 'Facebook'}
updated it companies are {'Amazon', 'Twitter', 'Oracle', 'Garuda', 'Ford', 'IBM', 'Microsoft', 'Apple', 'Flipkart', 'Google', 'Facebook'}
Updated list: {'Amazon', 'Twitter', 'Oracle', 'Garuda', 'Ford', 'IBM', 'Microsoft', 'Apple', 'Google', 'Facebook'}
A union B is: {19, 20, 22, 24, 25, 26, 27, 28}
A intersection B is: {19, 20, 22, 24, 25, 26}
True
False
joining A with B {19, 20, 22, 24, 25, 26, 27, 28}
joining B with A {19, 20, 22, 24, 25, 26, 27, 28}
symmetric difference between A and B {27, 28}
None
None
None
set of ages: {19, 22, 24, 25, 26}
length of set of list is: 5
length of set is : 8
(base) ponugotis-Air-7:assignment1 ponugotivamshi$
```

Q4. Created a set called it_companies. And created sets named A and B. And created a list of ages. Found the length of the set it_companies. Added Twitter to the set. Added new companies to the set. Performed the operations as given in the question such as joining A and B, A intersection B etc., Code and output are shown as above.

The screenshot shows the VS Code interface with the Explorer sidebar on the left displaying a folder named 'ASSIGNMENT1' containing files 'Question_1.py' through 'Question_9.py'. The main editor window displays the code for 'Question_5.py' with the following Python code:

```
1 import math #importing math package
2 radius=30 #initializing radius with 30
3 _area_of_circle_ = int(math.pi*(radius**2)) #calculating area of circle
4 print("Area of circle is: ",_area_of_circle_)
5 _circum_of_circle_ = int(2*math.pi*radius) #calculating circumference of circle
6 print("circumference of the circle is ",_circum_of_circle_)
7 rad=int(input("Enter the Radius ")) #taking input radius
8 print("Area of circle is: ",int(math.pi*(rad**2)))
```

Below the code editor, the 'TERMINAL' tab is active, showing the command prompt output:

```
(base) ponugotis-Air-7:assignment1 ponugotivamshi$ python3 Question_5.py
Area of circle is: 2827
circumference of the circle is 188
Enter the Radius 6
Area of circle is: 113
(base) ponugotis-Air-7:assignment1 ponugotivamshi$
```

The status bar at the bottom indicates the file is at 'Ln 8, Col 52' with 'Spaces: 4', 'UTF-8' encoding, 'CRLF' line endings, and 'Python 3.8.2 64-bit' interpreter.

Q5.Calculated the area of circle with radius 30 and it was assigned it to the variable named `_area_of_circle_` . Calculated the circumference of the circle and assigned it the variable named `_circum_of_circle_` . Againtaken radius as input and found radius of the circle.Code and output are shown above

The screenshot shows the VS Code interface with the Explorer sidebar on the left displaying a folder named 'ASSIGNMENT1' containing files 'Question_1.py' through 'Question_9.py'. The main editor window displays the code for 'Question_6.py' with the following Python code:

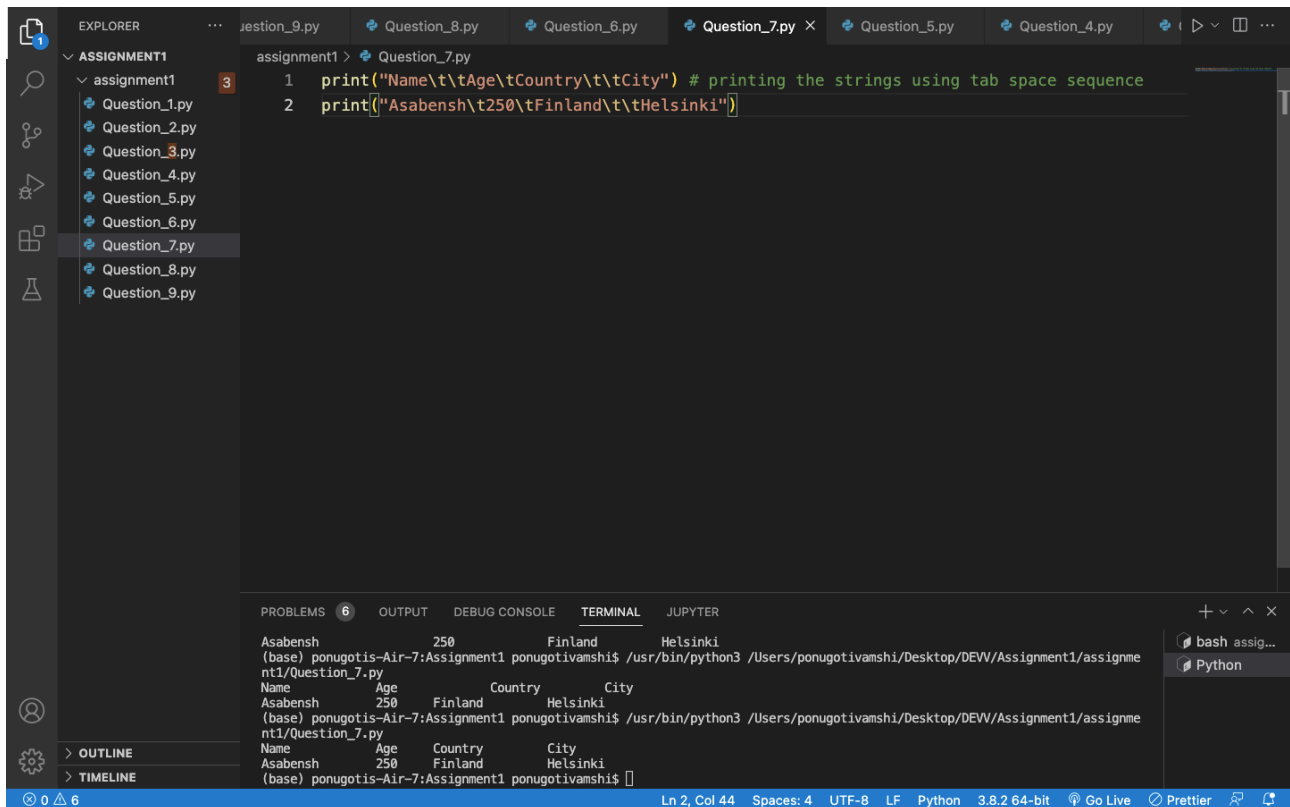
```
1 phrase="I am a teacher and I love to inspire and teach people"
2 print("No of unique words in the given phrase are: ",len(set(phrase.split(' '))))
```

Below the code editor, the 'TERMINAL' tab is active, showing the command prompt output:

```
(base) ponugotis-Air-7:assignment1 ponugotivamshi$ python3 Question_6.py
No of unique words in the given phrase are: 10
(base) ponugotis-Air-7:assignment1 ponugotivamshi$
```

The status bar at the bottom indicates the file is at 'Ln 2, Col 82' with 'Spaces: 4', 'UTF-8' encoding, 'CRLF' line endings, and 'Python 3.8.2 64-bit' interpreter.

Q6. Found the number of unique words used to create the given sentence with the code shown in above.



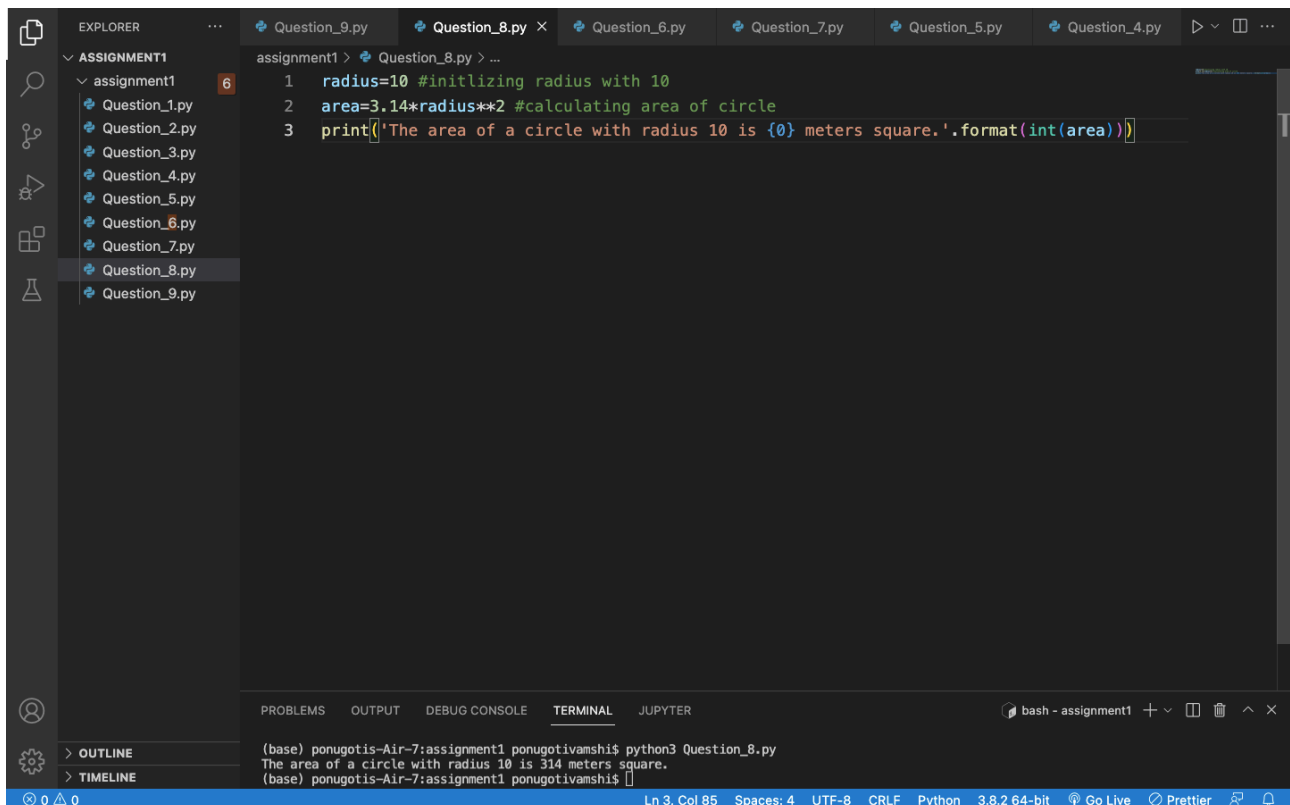
The screenshot shows the Visual Studio Code editor with a file explorer on the left containing a folder named 'ASSIGNMENT1' with subfolders 'assignment1' and 'Question_1.py' through 'Question_9.py'. The main editor window displays 'Question_7.py' with the following code:

```
1 print("Name\tAge\tCountry\tCity") # printing the strings using tab space sequence
2 print("Asabensh\t250\tFinland\tHelsinki")
```

The terminal at the bottom shows the output of the script:

```
Asabensh      250      Finland      Helsinki
(base) ponugotis-Air-7:Assignment1 ponugotivamshi$ /usr/bin/python3 /Users/ponugotivamshi/Desktop/DEV/Assignment1/assignment1/Question_7.py
Name      Age      Country      City
Asabensh      250      Finland      Helsinki
(base) ponugotis-Air-7:Assignment1 ponugotivamshi$ /usr/bin/python3 /Users/ponugotivamshi/Desktop/DEV/Assignment1/assignment1/Question_7.py
Name      Age      Country      City
Asabensh      250      Finland      Helsinki
(base) ponugotis-Air-7:Assignment1 ponugotivamshi$
```

Q7. Using tab space sequence printed the strings as given. Code and output as shown in above.



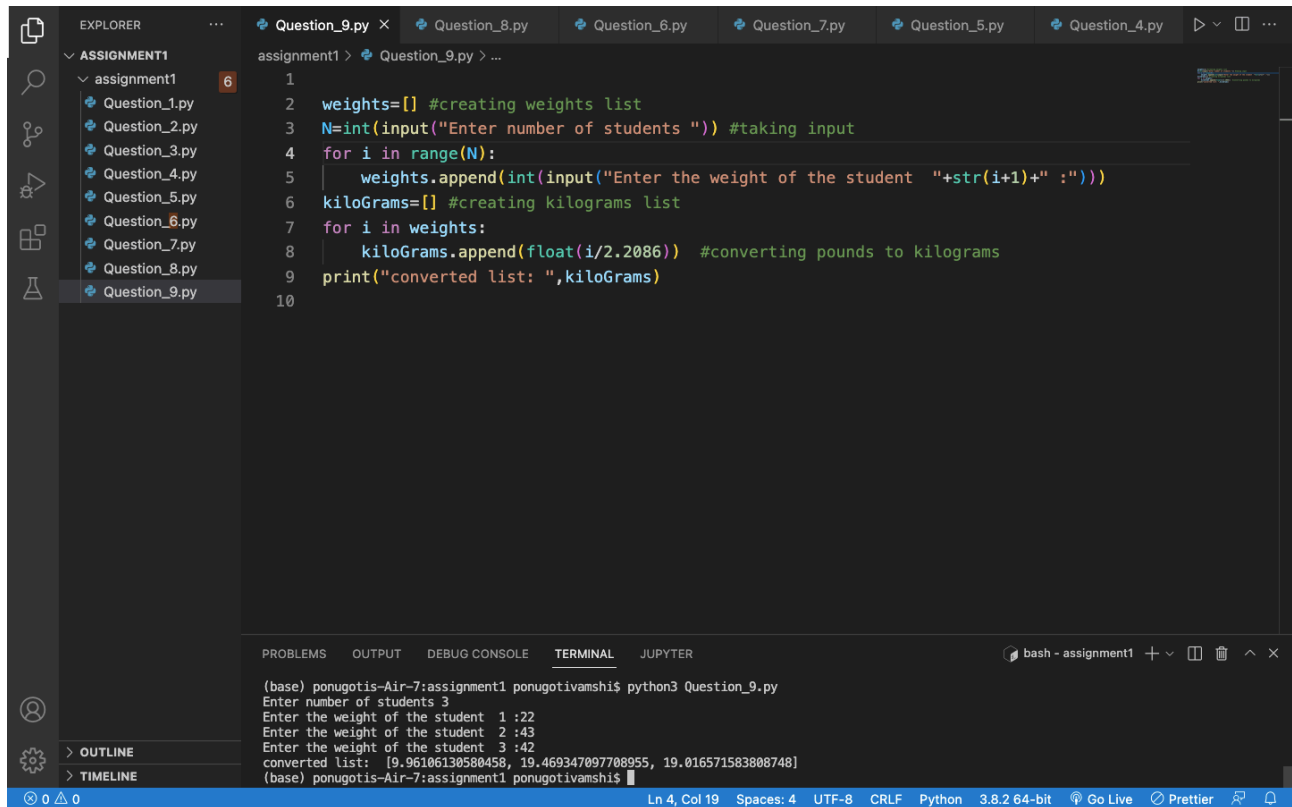
The screenshot shows the Visual Studio Code editor with a file explorer on the left containing a folder named 'ASSIGNMENT1' with subfolders 'assignment1' and 'Question_1.py' through 'Question_9.py'. The main editor window displays 'Question_8.py' with the following code:

```
1 radius=10 #initlizing radius with 10
2 area=3.14*radius**2 #calculating area of circle
3 print('The area of a circle with radius 10 is {0} meters square.'.format(int(area)))
```

The terminal at the bottom shows the output of the script:

```
(base) ponugotis-Air-7:assignment1 ponugotivamshi$ python3 Question_8.py
The area of a circle with radius 10 is 314 meters square.
(base) ponugotis-Air-7:assignment1 ponugotivamshi$
```

Q8. Using the string formatting method displayed the area of circle. Code as shown as above.



```
1
2 weights=[] #creating weights list
3 N=int(input("Enter number of students ")) #taking input
4 for i in range(N):
5     weights.append(int(input("Enter the weight of the student "+str(i+1)+" :")))
6 kiloGrams=[] #creating kilograms list
7 for i in weights:
8     kiloGrams.append(float(i/2.2086)) #converting pounds to kilograms
9 print("converted list: ",kiloGrams)
10
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

bash - assignment1

(base) ponugotis-Air-7:assignment1 ponugotivamshi\$ python3 Question_9.py
Enter number of students 3
Enter the weight of the student 1 :22
Enter the weight of the student 2 :43
Enter the weight of the student 3 :42
converted list: [9.96186198580458, 19.469347897708955, 19.016571583808748]
(base) ponugotis-Air-7:assignment1 ponugotivamshi\$

Q9. Taken Weights as input in pounds and calculated in terms of kgs. Code and output as shown in above.

Video link: https://drive.google.com/file/d/1EdinI9Xg7uTHFloYDHYM3Tw27ywGG__s/view?usp=sharing
Github link: https://github.com/vxp28550/ML_Assignment



By splitting the data into 2 equal parts for training & testing.

training: $[1, 3, 6, 11]$

$\Rightarrow [',', '*', '*', '.']$

testing: $[2, 6, 7, 10]$

$\Rightarrow [',', '*', '.', '.']$

confusion Matrix:-

		prediction.	
		0	x
Truth	0	TN	FP
	x	FN	TP

The nearest neighbours for 2 are $[1, 3, 6]$ for which

the values are $[',', '*', '*']$

predicted for 2 = $'*'$

but the value is $'.'$ so it is $'FP'$

\Rightarrow for nearest neighbours of $6 = [6, 7, 3]$
 which are having values = $['*', '0', '*']$
 from this predicted value for $6 = *$
 and given also $*$, so it is true positive.

\Rightarrow ~~for~~ nearest neighbours of $7 = [6, 6, 10]$ which are having values
 = $['*', '*', '0']$, so the predicted value
 is $*$ and the given is $'0'$. so it
 is false positive.

\Rightarrow ~~for~~ nearest neighbours of $10 = [11, 7, 6]$
 which are having values = $['0', '0', '*']$
 so the predicted value is $'0'$ and
 actual value is also $'0'$. so it is
 True negative.

confusion Matrix:-

	0	1
0	1	2
1	0	1

\Rightarrow Accuracy:

$$\frac{(TP + TN)}{P + N} = \frac{1 + 1}{3 + 1} = \frac{2}{4} = \frac{1}{2} = 50\%$$

$$\text{sensitivity} = \frac{TP}{TP+FN} = \frac{1}{1+0} = 1$$

$$\text{specificity} = \frac{TN}{FP+TN} = \frac{1}{2+1} = \frac{1}{3}$$