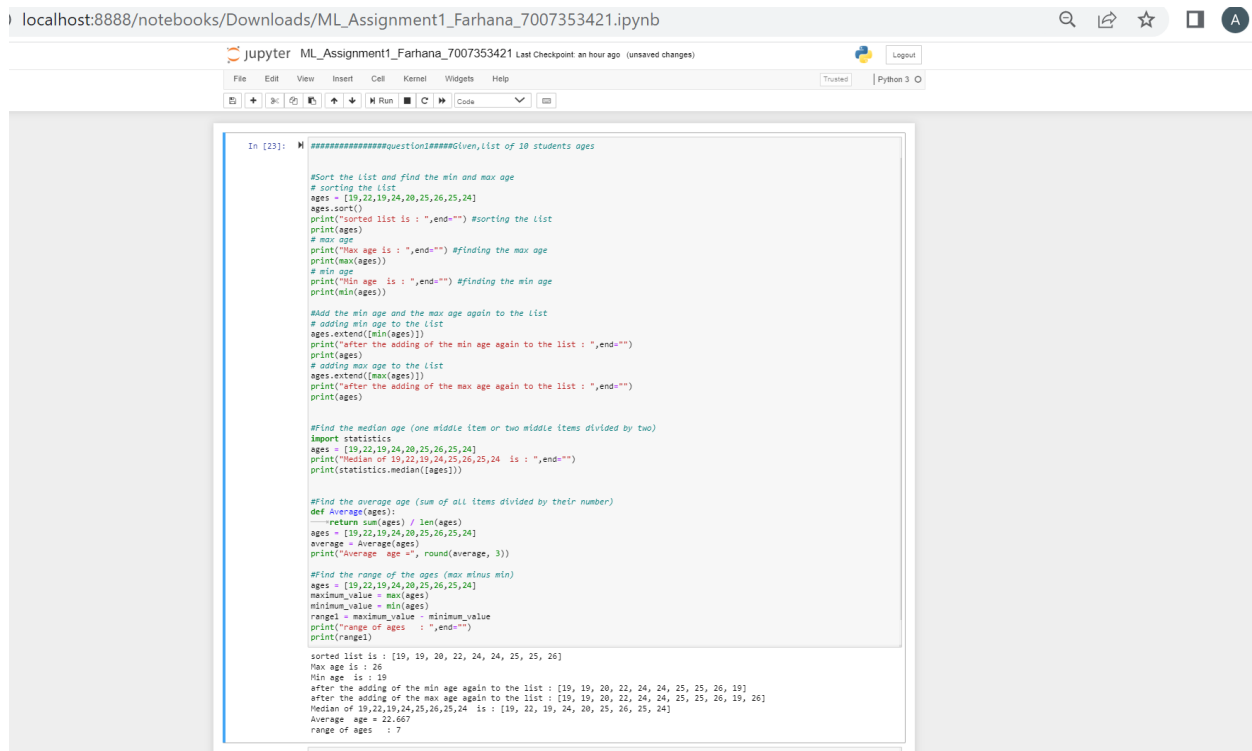


## Question 1

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)



The screenshot shows a Jupyter Notebook interface with a single code cell. The code cell contains a Python script that performs the following steps:

```
In [23]: #####Question1#####Given, list of 10 students ages

#Sort the list and find the min and max age
# sorting the list
ages = [19,22,19,24,20,25,26,25,24]
ages.sort()
print("sorted list is : ",end="") #sorting the list
print(ages)
# max age
print("Max age is : ",end="") #finding the max age
print(max(ages))
# min age
print("Min age is : ",end="") #finding the min age
print(min(ages))

#Add the min age and the max age again to the list
# adding min age to the list
ages.extend([min(ages)])
print("after the adding of the min age again to the list : ",end="")
print(ages)
# adding max age to the list
ages.extend([max(ages)])
print("after the adding of the max age again to the list : ",end="")
print(ages)

#Find the median age (one middle item or two middle items divided by two)
import statistics
ages = [19,22,19,24,20,25,26,25,24]
print("Median of 19,22,19,24,20,25,26,25,24 is : ",end="")
print(statistics.median(ages))

#Find the average age (sum of all items divided by their number)
def Average(ages):
    return sum(ages) / len(ages)
ages = [19,22,19,24,20,25,26,25,24]
average = Average(ages)
print("Average age = ", round(average, 3))

#Find the range of the ages (max minus min)
ages = [19,22,19,24,20,25,26,25,24]
maximum_value = max(ages)
minimum_value = min(ages)
range1 = maximum_value - minimum_value
print("range of ages : ",end="")
print(range1)

sorted list is : [19, 19, 20, 22, 24, 24, 25, 25, 26]
Max age is : 26
Min age is : 19
after the adding of the min age again to the list : [19, 19, 20, 22, 24, 24, 25, 25, 26, 19]
after the adding of the max age again to the list : [19, 19, 20, 22, 24, 24, 25, 25, 26, 19, 26]
Median of 19,22,19,24,20,25,26,25,24 is : 22.5
Average age = 22.667
range of ages : 7
```

The output of the code is displayed below the code cell, showing the sorted list, the maximum and minimum ages, the list after adding the minimum and maximum ages, the median age, the average age, and the range of ages.

A list of 10 student's ages are sorted in an order given ages=[ 19, 22, 19, 24, 20, 25, 26, 24, 25, 24] it is sorted list is : [19, 19, 20, 22, 24, 24, 25, 25, 26] by using sort() function i.e. ages.sort()

For finding maximum age from the given list by using max(ages), and is printed as 26  
Similarly minimum age from the given list is 19 by using min(ages).

Adding minimum and maximum

ages of students to the already given list by using .extend([minimum (ages)]) and the .extend([maximum(ages)]).

Output is [19, 19, 20, 22, 24, 24, 25, 25, 26, 19]

Output is [19, 19, 20, 22, 24, 24, 25, 25, 26, 19, 26] respectively

Then statistics is used to get the median age like one middle items divided by two

with the `statistics.median([ages])` median function is used. As one middle items is printed from given ages list. Output is [19, 22, 19, 24, 20, 25, 26, 25, 24]

To Find the average age as sum of all items divided by their number , `def` is used `def Average(ages):`  
`return sum(ages) / len(ages)`

And with average formula

`average = Average(ages)` it gives output as 22.667

To find the range of ages (min minus max) , get the `max[ages]` and `min[ages]` and apply formula  
`range1 = maximum_value - minimum_value`

Output is 7

---

---

## **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

The screenshot shows a Jupyter Notebook window titled 'Untitled3' with a last checkpoint 19 minutes ago. The code in the cell is as follows:

```
In [18]: #####question2#####
#Create an empty dictionary called dog
dog = {}

#Add name, color, breed, legs, age to the dog dictionary
dog['name'] = 'Blondie'
dog['colour'] = 'Golden brown'
dog['breed'] = 'Golden Retriever'
dog['legs'] = 'Four'
dog['age'] = 'Eight years'
print(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': 'ayesha', 'last_name': 'farhana', 'gender': 'female', 'age': '24', 'marital status': 'married', 'skills': 'Co
print(student)

#Get the length of the student dictionary
p=len(student)
print(p)

#Get the value of skills and check the data type, it should be a list
print(student['skills'])
print(type(student['skills']))

#Modify the skills values by adding one or two skills
student['skills'] = 'Collaboration talent, Leadership experience'
print(student)

#Get the dictionary keys as a list
student = {'first_name': 'ayesha', 'last_name': 'farhana', 'gender': 'female', 'age': '24', 'marital status': 'married', 'skills': 'Co
keylist = (student.keys())
print(keylist)

#Get the dictionary values as a list
keylist = (student.values())
print(keylist)

{'name': 'Blondie', 'colour': 'Golden brown', 'breed': 'Golden Retriever', 'legs': 'Four', 'age': 'Eight years'}
{'first_name': 'ayesha', 'last_name': 'farhana', 'gender': 'female', 'age': '24', 'marital status': 'married', 'skills': 'Co
laboration talent', 'country': 'India', 'city': 'warangal', 'address': '#731 st'}
0
Collaboration talent
<class 'str'>
{'first_name': 'ayesha', 'last_name': 'farhana', 'gender': 'female', 'age': '24', 'marital status': 'married', 'skills': 'Co
laboration talent, Leadership experience', 'country': 'India', 'city': 'warangal', 'address': '#731 st'}
dict_keys(['first_name', 'last_name', 'gender', 'age', 'marital status', 'skills', 'country', 'city', 'address'])
dict_values(['ayesha', 'farhana', 'female', '24', 'married', 'Collaboration talent', 'India', 'warangal', '#731 st'])
```

First to Create an empty dictionary called dog as dog = {}

Add name, color, breed, legs, age to the dog dictionary are assigned

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

entered as keys and values and assigned variable as 'student'

length of the student of the dictionary is len(student).

To Get the value of skills and check the data type, it should be a list  
(type(student["skills"]))

Modify the skills values by adding one or two skills. student["skills"]="Collaboration talent,Leadership experience".

Get the dictionary keys as a list. student =

{"first\_name": "ayesha", "last\_name": "farhana", "gender": "female", "age": "24", "marital status": "married", "skills": "Collaboration talent", "country": "india", "city": "warangal", "address": "#731 st"}.

keylist =(student.keys()).

dict\_keys(['first\_name', 'last\_name', 'gender', 'age', 'marital status', 'skills', 'country', 'city', 'address'])

To Get the dictionary values as a list. keylist = (student.values())

{'name': 'Blondie', 'colour': 'Golden brown', 'breed': 'Golden Retriever', 'legs': 'Four', 'age': 'Eight years'}

dict\_values(['ayesha', 'farhana', 'female', '24', 'married', 'Collaboration talent', 'india', 'warangal', '#731 st'])

---

---

### 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

localhost:8888/notebooks/Downloads/Untitled3.ipynb?kernel\_name=python3

jupyter Untitled3 Last Checkpoint 20 minutes ago (unsaved changes) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

```
In [19]: #Create a tuple containing names of your sisters and your brothers (imaginary siblings are
#fine)
# with the use of string
bros_tuple1 = ('adil', 'awez', 'mohtadun') #names of your brothers
sisters_tuple2 = ('shaziya', 'amena', 'zeba') #names of your sisters
print("\nTuple names of your sisters and your brothers: ")
print(bros_tuple1)
print(sisters_tuple2)

#Join brothers and sisters tuples and assign it to siblings
siblings = bros_tuple1+sisters_tuple2
print(siblings)

#How many siblings do you have?
count=len(siblings)
print(count)

#Modify the siblings tuple and add the name of your father and mother and assign it to
#family_members
family_members = list(siblings)
family_members = siblings + ('rasheed','sulthana') #modifying with father and mother
siblings = tuple(family_members)
print(siblings)

Tuple names of your sisters and your brothers:
('adil', 'awez', 'mohtadun')
('shaziya', 'amena', 'zeba')
('adil', 'awez', 'mohtadun', 'shaziya', 'amena', 'zeba')
6
('adil', 'awez', 'mohtadun', 'shaziya', 'amena', 'zeba', 'rasheed', 'sulthana')
```

In [ ]:

To Create a tuple containing names of your sisters and your brothers (imaginary siblings are fine) with the use of string

bros\_tuple1 = ('adil', 'awez', 'mohtadun') are names of your brothers

sisters\_tuple2 = ('shaziya', 'amena', 'zeba') are names of your sisters then prints both brother and sister tuple

Join brothers and sisters tuples and assign it to siblings , siblings = bros\_tuple1+sisters\_tuple2

How many siblings do you have? count=len(siblings)

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

```
family_members = list(siblings)
```

```
family_members = siblings + ('rasheed','sulthana') modifying with father and mother by names as  
rasheed and sulthana
```

```
siblings = tuple(family_members) and prints the sibling tuple. As
```

```
('adil', 'awez', 'mohtadun', 'shaziya', 'amena', 'zeba', 'rasheed', 'sulthana')
```

-----  
-----

#### **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.

```
In [20]: #####question4#### Given

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
count=len(it_companies)
print(count)

Add 'Twitter' to it_companies
#it1 = it_companies,('Twitter')
#it_companies = tuple(it1)

newTuple = ('Twitter',)
it_companies = tuple(it_companies) + newTuple
print(it_companies)

#Insert multiple IT companies at once to the set it_companies
newTuple = ('Tata Consultancy service', 'HCL', 'Cognizant', 'Infosys', 'Atmacs',)
it_companies = tuple(it_companies) + newTuple
print(it_companies)

#Remove one of the companies from the set it_companies
z = list(it_companies)
z.remove('Google')
it_companies_remove = tuple(z)
print("Removed one company:", it_companies_remove)

What is the difference between remove and discard
#The discard() method removes the specified item from the set. This method is different from the remove() method, because th

#Join A and B
z = tuple(A) + tuple(B)
print("Joining A and B:",z)

#Find A intersection B
z = A.intersection(B)
print("A intersection B:",z)

# Is A subset of B
print("A is subset of B" ,end="")
print(A.issubset(B))

Make A and B disjoint sets
result = A.isdisjoint(B)
print("A and B disjoint sets:", result)

#Join A with B and B with A
AB= tuple(A) + tuple(B)
print(AB)
BA= tuple(B) + tuple(A)
print(BA)
ABA = AB + BA
print("Join A with B and B with A:",ABA)

What is the symmetric difference between A and B
symmetric_dif= A.symmetric_difference(B)
print("the symmetric difference between A and B is:",symmetric_dif)

#Delete the sets completely
c = A.clear()
print("updated A set", c)

d = B.clear()
```

```

print("A is subset of B:",end="")
print(A.issubset(B))

#Are A and B disjoint sets
result = A.isdisjoint(B)
print("A and B disjoint sets:", result)

#Join A with B and B with A
AB= tuple(A) + tuple(B)
print(AB)
BA= tuple(B) + tuple(A)
print(BA)
ABA = AB + BA
print("Join A with B and B with A:",ABA)

#What is the symmetric difference between A and B
symmetric_dif= A.symmetric_difference(B)
print("the symmetric difference between A and B is:",symmetric_dif)

#Delete the sets completely
c = A.clear()
print("updated A set", c)

d = B.clear()
print("updated B set", d)

#Convert the ages to a set and compare the length of the list and the set.
age_set = set(age)
print("Converting the ages to a set is:", age_set)

list_length=len(age)
print("the length of the list:", list_length)

set_length=len(age_set)
print("the length of the set:", set_length)

```

7

```

('Microsoft', 'Apple', 'Facebook', 'Oracle', 'Amazon', 'Google', 'IBM', 'Twitter')
('Microsoft', 'Apple', 'Facebook', 'Oracle', 'Amazon', 'Google', 'IBM', 'Twitter', 'Tata Consultancy service', 'HCL', 'Cognizant', 'Infosys', 'Atmacs')
Removed one company: ('Microsoft', 'Apple', 'Facebook', 'Oracle', 'Amazon', 'IBM', 'Twitter', 'Tata Consultancy service', 'HCL', 'Cognizant', 'Infosys', 'Atmacs')
Joining A and B: (19, 20, 22, 24, 25, 26, 19, 20, 22, 24, 25, 26, 19, 20, 22, 24, 25, 26, 27, 28)
A intersection B: {19, 20, 22, 24, 25, 26}
A is subset of B: True
A and B disjoint sets: False
(19, 20, 22, 24, 25, 26, 19, 20, 22, 24, 25, 26, 27, 28)
(19, 20, 22, 24, 25, 26, 27, 28, 19, 20, 22, 24, 25, 26)
Join A with B and B with A: (19, 20, 22, 24, 25, 26, 19, 20, 22, 24, 25, 26, 27, 28, 19, 20, 22, 24, 25, 26)
the symmetric difference between A and B is: (27, 28)
updated A set None
updated B set None
Converting the ages to a set is: {19, 22, 24, 25, 26}
the length of the list: 8
the length of the set: 5

```

it\_companies = {'Facebook', 'Google', 'Microsoft', 'Apple', 'IBM', 'Oracle', 'Amazon'} to Find the length of the set it\_companies by using  
count=len(it\_companies) i.e. 7

To Add 'Twitter' to it\_companies , A variable is used as newTuple to assign for Twitter and then tuple is used to combine Twitter in the given it\_companies

newTuple = ('Twitter',) it\_companies = tuple(it\_companies) + newTuple and printed it the output is ('Facebook', 'Google', 'IBM', 'Microsoft', 'Apple', 'Amazon', 'Oracle', 'Twitter').

To Insert multiple IT companies at once to the set it\_companies added few companies ('Tata Consultancy service' , 'HCL', 'Cognizant', 'Infosys', 'Atmacs' , )

adding this multiple compies to it\_companies and output is ('Facebook', 'Google', 'IBM', 'Microsoft', 'Apple', 'Amazon', 'Oracle', 'Twitter', 'Tata Consultancy service', 'HCL', 'Cognizant', 'Infosys', 'Atmacs')

To Remove one of the companies from the set it\_companies, First list of it\_companies are assigned to varaiaible and then it convinient to remove one company among all.

z = list(it\_companies) z.remove('Google') . it\_companies\_remove = tuple(z) . tuple has been used to collect

What is the difference between remove and discard

The discard() method removes the specified item from the set. This method is different from the remove() method, because the remove() method will raise an error if the specified item does not exist, and the discard() method will not. Joining both set and A and B

A = {19, 22, 24, 20, 25, 26} is the set given B = {19, 22, 20, 25, 26, 24, 28, 27} , #Join A and B . J = tuple(A)

+ tuple(B) .Find A intersection B     $z = A.intersection(B)$     Is A subset of B, Disjoint sets are false  
 Join A with B and B with A ,  $AB = tuple(A) + tuple(B)$  ,  $BA = tuple(B) + tuple(A)$  ,  $ABA = AB + BA$   
 What is the symmetric difference between A and B and the formula is  $symmetric\_dif = A.symmetric\_difference(B)$  . To Delete the sets completely  
 $c = A.clear()$      $d = B.clear()$   
 To Convert the ages to a set and compare the length of the list and the set.  
 Converting the ages to a set is  $age\_set = set(age)$  . The length of the list is  $len(age)$  . the length of the set is  $len(age\_set)$

---

---

---

---

### 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.

localhost:8888/notebooks/Downloads/Untitled3.ipynb?kernel\_name=python3

jupyter Untitled3 Last Checkpoint: 23 minutes ago (unsaved changes) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

```
In [*]: # #####question5##### Given
#The radius of a circle is 30 meters.

radius_of_circle = 30

#Calculate the area of a circle and assign the value to a variable name of _area_of_circle_
pi = 3.14
_area_of_circle_ = pi * radius_of_circle * radius_of_circle
print("Area of a circle = %.2f" % _area_of_circle_)

#Calculate the circumference of a circle and assign the value to a variable name of _circum_of_circle_
_circum_of_circle_ = 2 * pi * radius_of_circle
print("Circumference of a circle = %.2f" % _circum_of_circle_)

#Take radius as user input and calculate the area.
radiusOfCircle = float(input("Enter the radius of a circle:")) # User input
area = pi * radiusOfCircle * radiusOfCircle #Area Formula
print("Area of a circle = %.2f" % area)

Area of a circle = 2826.00
Circumference of a circle = 188.40

Enter the radius of a circle: 
```

In [ ]: #



localhost:8888/notebooks/Downloads/Untitled4.ipynb?kernel\_name=python3

jupyter Untitled4 Last Checkpoint a minute ago (unsaved changes)

```
In [1]: #####Question5#### Given
#The radius of a circle is 30 meters.

radius_of_circle = 30

#Calculate the area of a circle and assign the value to a variable name of _area_of_circle_
pi = 3.14
_area_of_circle_ = pi * radius_of_circle * radius_of_circle
print('Area of a circle = %.2f' % _area_of_circle_)

#Calculate the circumference of a circle and assign the value to a variable name of _circum_of_circle_
_circum_of_circle_ = 2 * pi * radius_of_circle
print('Circumference of a circle = %.2f' % _circum_of_circle_)

#Take radius as user input and calculate the area.
radiusOfCircle = float(input("Enter the radius of a circle:")) # User input
area = pi * radiusOfCircle * radiusOfCircle #Area Formula
print("Area of a circle = %.2f" % area)

Area of a circle = 2826.00
Circumference of a circle = 188.40
Enter the radius of a circle: 10
```

localhost:8888/notebooks/Downloads/Untitled4.ipynb?kernel\_name=python3

jupyter Untitled4 Last Checkpoint a minute ago (unsaved changes)

```
In [2]: #####Question5#### Given
#The radius of a circle is 30 meters.

radius_of_circle = 30

#Calculate the area of a circle and assign the value to a variable name of _area_of_circle_
pi = 3.14
_area_of_circle_ = pi * radius_of_circle * radius_of_circle
print('Area of a circle = %.2f' % _area_of_circle_)

#Calculate the circumference of a circle and assign the value to a variable name of _circum_of_circle_
_circum_of_circle_ = 2 * pi * radius_of_circle
print('Circumference of a circle = %.2f' % _circum_of_circle_)

#Take radius as user input and calculate the area.
radiusOfCircle = float(input("Enter the radius of a circle:")) # User input
area = pi * radiusOfCircle * radiusOfCircle #Area Formula
print("Area of a circle = %.2f" % area)

Area of a circle = 2826.00
Circumference of a circle = 188.40
Enter the radius of a circle:10
Area of a circle = 314.00
```

Assigned radius is 10 variable as radius\_of\_circle = 30, then to Calculate the area of a circle and assign the value to a variable name of \_area\_of\_circle\_ , it is given in question pi value as pi = 3.14 ,the formula for area of circle as \_area\_of\_circle\_ = pi \* radius\_of\_circle \* radius\_of\_circle , and to Calculate the circumference of a circle and assign the value to a variable name of \_circum\_of\_circle ;

\_circum\_of\_circle = 2 \* pi \* radius\_of\_circle

radius as user input and calculate the area. it is enter as 10

area = pi \* radiusOfCircle \* radiusOfCircle is 314.00

## 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.

```
localhost:8888/notebooks/Downloads/Untitled4.ipynb?kernel_name=python3

jupyter Untitled4 Last Checkpoint: a few seconds ago (unsaved changes)

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

In [1]: #####question5#### Given
#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.
sentence = "I am a teacher and I love to inspire and teach people"
print(sentence)
allWords = sentence.split()
print('words split:', allWords)
unique_words = set(allWords)
print("The number of unique words are:", len(unique_words))

sentence: I am a teacher and I love to inspire and teach people
words split: ['I', 'am', 'a', 'teacher', 'and', 'I', 'love', 'to', 'inspire', 'and', 'teach', 'people']
The number of unique words are: 10

In [ ]: 
```

The solution for the given sentence “I am a teacher and I love to inspire and teach people” it is assigning to the variable as ‘sentence’ and printd it. To find unique words in this sentence , first words have to be separated , by using split() function it will be easier to separate the words and assigned to one variable here as ‘allWords’ i.e. allWords = sentence.split(). Unique words printed by using set i.e. unique\_words = set(allWords) then the length of the unique\_words is len(unique\_words) , that is 10.

-----  
-----  
-----  
-----

## Question 7

Use a tab escape sequence to get the following lines.

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

```
localhost:8888/notebooks/Downloads/Untitled4.ipynb?kernel_name=python3

jupyter Untitled4 Last Checkpoint: 3 minutes ago (unsaved changes)

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

In [3]: #####question7#### Given
#Use a tab escape sequence to get the following lines.
#Name Age Country City
#Asabeneh 250 Finland Helsinki

string1 = "Name\tAge\tCountry\tCity"
print(string1)
string2 = "Asabeneh\t250\tFinland\tHelsinki"
print(string2)

Name      Age      Country      City
Asabeneh  250      Finland     Helsinki

In [ ]: 
```

By using \t here for tab escape sequence by assigning string1 and string2 variables divided the above given Sequence as Name Age Country City in one variable and

Asabeneh 250 Finland Helsinki in another variable by using \t to the each word. Output shows as expected.

---

### **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.”

```
In [24]: #####question8#####Given
#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."

radius = 10
#Area of circle Formula
area = 3.14 * radius ** 2
#Given sentence in string format
value = "The area of a circle with radius 10 is {area_of_circle:.2f} meters square."
print(value.format(area_of_circle = 314))

The area of a circle with radius 10 is 314.00 meters square.
```

```
In [20]: #####question8#####Given
#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."

radius = 10
#Area of circle Formula
area = 3.14 * radius ** 2
#Given sentence in string format
value = "The area of a circle with radius 10 is {area_of_circle:} meters square."
print(value.format(area_of_circle = 314))

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

Given radius as 10 and formula for area of circle is area =3.14\*radius\*\*2

By using string format as by using {} like, value =“The area of a circle with radius 10 is {area\_of\_circle:.2f} meters square.”

And it is printed in the output “The area of a circle with radius 10 is 314 meters square.”

---

### **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]

```

In [*]: #####question9#####Given
#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]

#number of students
s = int(input("Enter the number of students: "))
#this line read inputs from user using map() function
a = list(map(int,input("\nEnter the weights: ").strip().split()))[:s]
print("\nL1: ", a)
x=[i/2.2048364 for i in a]
x=['%.2f' % elem for elem in x]
print(x)

```

Enter the number of students: 4

Enter the weights:

```

In [25]: #####question9#####Given
#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]

#number of students
s = int(input("Enter the number of students: "))
#this line read inputs from user using map() function
a = list(map(int,input("\nEnter the weights: ").strip().split()))[:s]
print("\nL1: ", a)
x=[i*0.453592 for i in a]
x=['%.2f' % elem for elem in x]
print(x)

```

Enter the number of students: 4

Enter the weights: 150 155 145 148

L1: [150, 155, 145, 148]  
 ['68.04', '70.31', '65.77', '67.13']

In this problem, assigning variable as 's' to read the number of students into the list `s = int(input("Enter the number of students: "))`. To read the weights of students, a variable is assigned as 'a' i.e. `a = list(map(int,input("\nEnter the weights: ").strip().split()))[:s]` by using `map()` functions and enters weights of the students then using for loop to get the iteration of 'a list' with 'i' variable and assigning it to 'x' variable i.e. `1pound=0.453592`, multiplying `0.453592` with `i` (weight in pounds) it is given with the split then it printed in list L1: [150, 155, 145, 148] then another separate array is created to print the conversion of weight in pounds to weights in kilograms `['68.04', '70.31', '65.77', '67.13']`