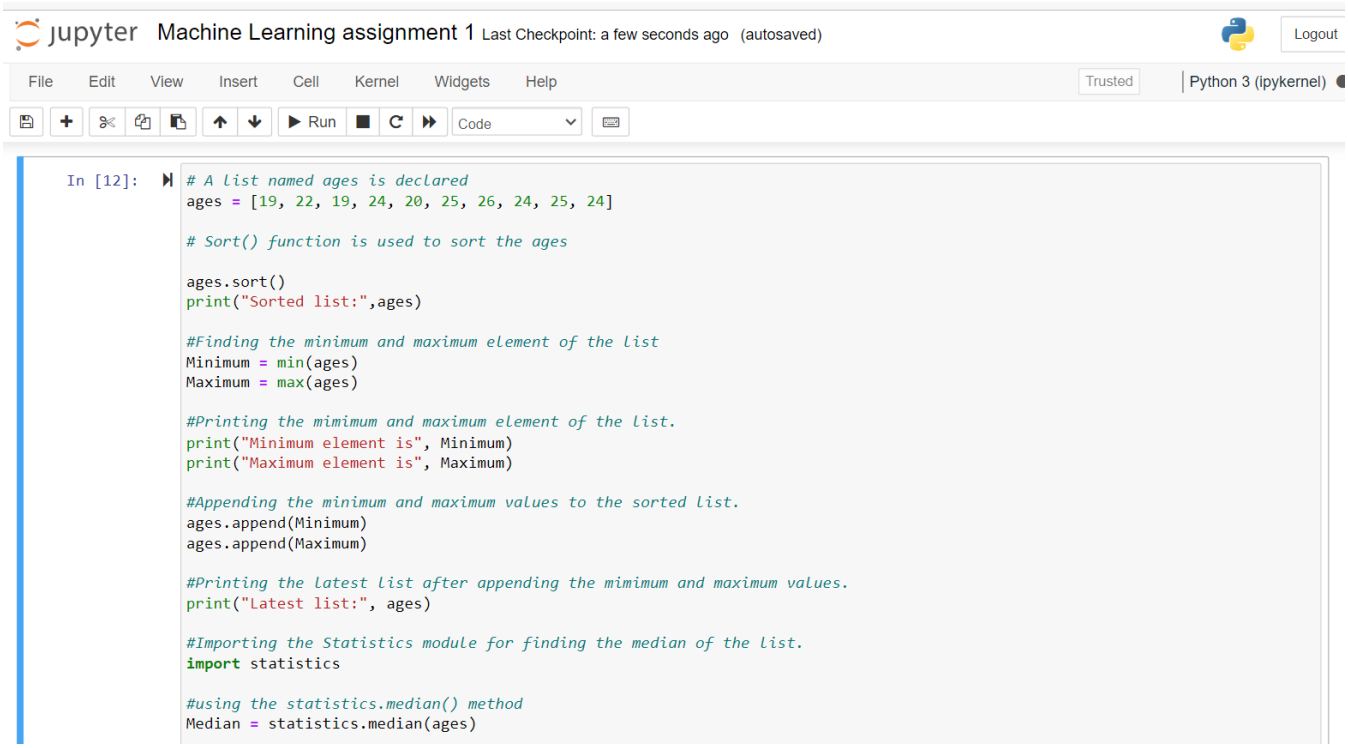
Machine Learning – Assignment 1

GitHub link: https://github.com/AldenaHaneesha/MachineLearning---Assignment-1

Question 1:

Solution:




```
#Printing the Median of the list
print("Median is:", Median)

#Finding the average of the list
Average = sum(ages)/len(ages)

#Printing the average element
print ("Avereage element is:", Average)

# The Range of ages is
print("The Range of list difference : ", ages[len(ages)-1]-ages[0])
```

Output:

```
Sorted list: [19, 19, 20, 22, 24, 24, 24, 25, 25, 26]
Minimum element is 19
Maximum element is 26
Latest list: [19, 19, 20, 22, 24, 24, 24, 25, 25, 26, 19, 26]
Median is: 24.0
Avereage element is: 22.75
The Range of list difference :  7
```

Explanation:

In this solution, we have initially declared a list named ages with the specific values and sorted the list "ages" using the sort () function.

The Minimum element in the list is shown by using the function min(list) and the maximum element is shown by using the function max(list) and appended the minimum and the maximum values to the sorted list.

The length of the list was found using the length function which determines the capacity of the list.

The range of the list is shown by making difference between last and first element in the list.

Range = ages[len(ages)-1]-ages[0])

Question 2:

Solution:

Machine Learning – Assignment 1

In [21]:

```
#A dictionary named Dog is Declared.

dog = { "name" : "Tommy"  , "color" : "black", "breed" : "pumalian" , "legs" : 4, "age" : 10}
#Printing the Dictionary

print("Dictionary Dog : ",dog)

"""Declaring the Student dictionary including the fields such as first name, last name, age,marital status,
skills, country and address."""

student = {"first_name" : "Haneesha","last_name" : "Aldena", "gender" : "Female","age" : 22,"marital_status" : "Single",
    "skills" : ["C", "C++", "Python", "Java"],"country" : "USA","city" : "kansas city", "address" : "8392 W 99th Ct"}
#Printing the Student Dictionary

print("Student:", student)

#Finding and printing the length of student dictonary

print("Length of the student dictonary is: ",len(student))

#Printing the values of skills and the data type.

print("Skills : ",student["skills"],"and Data type of skills : ",type(student["skills"]))

# Modifying the Skills list by appending the skill #JavaScript to the list.

student["skills"].append("JavaScript")

print("After Adding JavaScript Skill : ",student["skills"])

#Displaying student keys

print("Student keys are : ",student.keys())

#Displaying Values of student

print("Student values are : ",student.values())
```

Output:

```
Dictionary Dog :  {'name': 'Tommy', 'color': 'black', 'breed': 'pumalian', 'legs': 4, 'age': 10}
Student: {'first_name': 'Haneesha', 'last_name': 'Aldena', 'gender': 'Female', 'age': 22, 'marital_status': 'Single', 'skill
s': ['C', 'C++', 'Python', 'Java'], 'country': 'USA', 'city': 'kansas city', 'address': '8392 W 99th Ct'}
Length of the student dictonary is:  9
Skills :  ['C', 'C++', 'Python', 'Java'] and Data type of skills :  <class 'list'>
After Adding JavaScript Skill :  ['C', 'C++', 'Python', 'Java', 'JavaScript']
Student keys are :  dict_keys(['first_name', 'last_name', 'gender', 'age', 'marital_status', 'skills', 'country', 'city', 'a
ddress'])
Student values are :  dict_values(['Haneesha', 'Aldena', 'Female', 22, 'Single', ['C', 'C++', 'Python', 'Java', 'JavaScrip
t'], 'USA', 'kansas city', '8392 W 99th Ct'])
```

Explanation:

In this solution a dictionary named Dog has been declared.

Machine Learning – Assignment 1

dog = {“name" : "Tommy" , "color" : "black", "breed" : "pumalian" , "legs" : 4, "age" : 10}

and the keys and values of the dictionary has been printed.

Then a dictionary named student has been declared which includes the fields such as first name, last name, age, marital status, skills, country, and address. The length of student dictonary is found using the function len().

- ➔ len(student))

The values of skills are printed using the student[“skills”] and the data type of skills has been found using the type() function.

- ➔ print("Skills : ",student["skills"],"and Data type of skills : ",type(student["skills"]))

The Skills list is modified by appending the skill #JavaScript to the list.

- ➔ student["skills"].append("JavaScript")

The Student keys are printed using student.keys().

- ➔ print("Student keys are : ",student.keys())

The values of the student dictionary are displayed using student.values().

- ➔ print("Student values are : ",student.values())

Question 3:

Solution:

Machine Learning – Assignment 1

In [27]:

```
#A tuple named brothers is declared
brothers = ("Jeevitesh", "kishore", "likhit")
print("Brothers: ",brothers)

#A tuple named sister is declared

sisters = ("Priyanka", "priya")

print("Sisters : ",sisters)

#joining the brothers tuple and sisters tuple to the tuple named siblings

siblings = brothers + sisters

#Printing the tuple - Siblings
print ("Siblings: ", siblings)

#Total number of siblings
print("Total Siblings : ", len(siblings))

#Modifying the tuple value by adding mother and father name

H = list(siblings)
Father = "Samba"
Mother = "Suseela"
H.append(Father)
H.append(Mother)

# Assigning to family members

family_members = tuple(H)

print("Family Members : ",family_members)
```

Output:

```
Brothers:  ('Jeevitesh', 'kishore', 'likhit')
Sisters :  ('Priyanka', 'priya')
Siblings:  ('Jeevitesh', 'kishore', 'likhit', 'Priyanka', 'priya')
Total Siblings :  5
Family Members :  ('Jeevitesh', 'kishore', 'likhit', 'Priyanka', 'priya', 'Samba', 'Suseela')
```

Explanation:

In this problem the tuples named brothers and sisters are declared which consists of the names of brothers and sisters.

Both the tuples' brothers and sisters and appended using the concatenation operator "+" and assigned to a new tuple named siblings.

The length of the tuple siblings is found using the function len() (len(siblings)).

Later the tuple is modified by adding the father's name and mother name.

The names have been appended using the append () function.

Question 4:

Machine Learning – Assignment 1

Solution:

In [28]:

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

print("IT Companies : ", it_companies)
#Printing the length of it_companies

print("Length of it_compamies is: ",len(it_companies))

#Adding the item Twitter to it_companies

it_companies.add("Twiter")
print("it_companies after adding Twitter: ",it_companies)

#inserting multiple it companies to the set it_companies.

it_companies_2 = { 'TCS', 'Virtusa' }

it_companies.update(it_companies_2)

print("Adding Adding Items to it_compaies : ",it_companies)

#Removing item Virtusa from the set

it_companies.remove("Virtusa")

print("After Removing Virtusa from Set : ",it_companies)

#Difference betweeb remove() and discard

"""
The Remove() and Discard() method used to delete the item in the set. The
main differnece between remove() and discard() is remove() will raise an
issue if the element is not availabe in the set but discard element does not
raise an error if so.
"""
```

```
raise an error if so.
"""
#Example :
#Doesn't show any error
it_companies.discard("Virtusa")

print("After using discard function : ", it_companies)

#it_companies.remove("Virtusa")

# Shows Error if element is not available
#print("After using remove function : ", it_companies)

#Declring the sets A and B

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

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

#Joining the sets A and B
C=A.union(B)

print("Joining Set A and Set B : ",C)

#Finding the intersection of sets A and B

I = A.intersection(B)

print("Intersection of set A and set B : ", I)

#Checking Weather A is Subset of B or not

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

Machine Learning – Assignment 1

```
#Checking weather A is disjoint set of B or not

D = A.isdisjoint(B)
print("Weather A is disjoint set of B : ",D)

#Joining A with B and B with A

print("Joining Set A with Set B : ",A.union(B))

print("Joining Set B with Set A : ",B.union(A))

#Symmetric difference between A and B

Sy = A.symmetric_difference(B)

print("Symmetric Difference Between Set A and Set B : ",Sy)

#Deleting the entire Sets (Set A and Set B)

A.clear()
B.clear()

print(A)
print(B)

#Converting ages into complete set

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

print("Age :",age)

sage = set(age)

print("After converting age into set : ",sage, type(sage))

print("length of set age : ",len(sage))
print("length of list age : ",len(age))
```

Output:

```
IT Companies :  {'Google', 'Facebook', 'IBM', 'Microsoft', 'Amazon', 'Apple', 'Oracle'}
Length of it_compamies is:  7
it_companies after adding Twitter:  {'Google', 'Twiter', 'Facebook', 'IBM', 'Microsoft', 'Amazon', 'Apple', 'Oracle'}
Adding Adding Items to it_compaies :  {'Google', 'TCS', 'IBM', 'Virtusa', 'Amazon', 'Twiter', 'Facebook', 'Apple', 'Microsof
t', 'Oracle'}
After Removing Virtusa from Set :  {'Google', 'TCS', 'IBM', 'Amazon', 'Twiter', 'Facebook', 'Apple', 'Microsoft', 'Oracle'}
After using discard function :  {'Google', 'TCS', 'IBM', 'Amazon', 'Twiter', 'Facebook', 'Apple', 'Microsoft', 'Oracle'}
Joining Set A and Set B :  {19, 20, 22, 24, 25, 26, 27, 28}
Intersection of set A and set B :  {19, 20, 22, 24, 25, 26}
A is subset of B :  True
Weather A is disjoint set of B :  False
Joining Set A with Set B :  {19, 20, 22, 24, 25, 26, 27, 28}
Joining Set B with Set A :  {19, 20, 22, 24, 25, 26, 27, 28}
Symmetric Difference Between Set A and Set B :  {27, 28}
set()
set()
Age : [22, 19, 24, 25, 26, 24, 25, 24]
After converting age into set :  {19, 22, 24, 25, 26} <class 'set'>
length of set age :  5
length of list age :  8
```

Machine Learning – Assignment 1

Explanation:

A set named it_companies declared below

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

Length of the set is found using the len() function.

print("Length : ",len(it_companies))

New Items are added to the Set using the add() function.

it_companies.add("Twitter")

Later a new Set has been declared

it_companies_2 = { 'Virtusa', 'JPMC' }

The old set (it_companies) is updated with new set (it_companies_2) using update function

it_companies.update(it_companies_2)

An Item can be removed from the Set using the remove function. Virtusa is removed from the set.

it_companies.remove("Virtusa")

Difference between remove () and discard is as follows

The Remove() and Discard() method used to delete the item in the set. The

main difference between remove() and discard() is remove() will raise an

issue if the element is not available in the set but discard element does not

raise an error if so.

Set A and Set B is declared as below

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

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

Two sets are joined using the union function. The elements in both sets are assigned to another set below. Common elements are assigned only once after the union elements

C=A.union(B)

Intersection between two sets can be found using the intersection () function. Only common elements are assigned after the intersection operation.

I = A.intersection(B)

The issubset() function is used to determine whether a function is subset or not. This Function gives the true or false as return values.

S = A.issubset(B)

Machine Learning – Assignment 1

The isdisjoint () function is used to determine whether a function is disjoint or not. This Function gives the true or false as return values.

D = A.isdisjoint(B)

union is used to join the two sets.

print("Joining A with B : ",A.union(B))

print("Joining B with A : ",B.union(A))

Symmetric_difference() is used to check the symmetric difference between two function. This method returns all the methods in the set except the intersection elements.

Sy = A.symmetric_difference(B)

Clear method is used to delete the elements in the Set. It removes all the elements in the Set.

A.clear()

B.clear()

The Age Set is declared as below

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

Typecasting is a method of converting object of one type to another type. Here the list is converted into the age

sage = set(age)

The length of list before converting and length after converting is shown using the len() function.

print("length of set age : ",len(sage))

print("length of list age : ",len(age))

Question 5:
Solution:

In [3]:

```
#The radius of a circle is 30 meters
r=30

#Determining the area of the circle

_area_of_circle_ = 3.14 * (r**2)

print("The area of circle is: ", _area_of_circle_ )

#Finding the Circumference of the circle

_circum_of_circle_ = 2 * 3.14 * r

print("Circumference of circle : ", _circum_of_circle_)

print()

input_radius = int(input("Enter the radius "))

area = 3.14 * input_radius

print("area of circle with the radius determined  by the user:", area)
```

Machine Learning – Assignment 1

Output:

```
The area of circle is:  2826.0
Circumference of circle :  188.4

Enter the radius 6
area of circle with the radius determined  by the user: 18.84
```

Explanation:

Initially the radius of the circle is initialized as 30 meters.
The area of the circle is determined by formula 3.14*(radius**2).

_area_of_circle_ = 3.14 * (r**2)

#Finding The Circumference of the circle is determined by the formula 2*3.14*radius.

_circum_of_circle_ = 2 * 3.14 * r

Now the radius of the circle is determined by the input given by the user.

The input radius is given the type int.

input_radius = int(input("Enter the radius "))

And the area of the circle is calculated.

area = 3.14 * input_radius

Question 6:

Solution:

In [35]:
```
#Declaring the string

str = "I am a teacher and I love to inspire and teach people"

print("The Given string is : ", str)

# Stored the string str in a set using the split() function.

Set = set(str.split(" "))

#finding the number of unique words.

print("Number of unique words:", len(Set))

#Printing the unique words of a string by printing the set.

print("Unique words : ", Set)
```

```
The Given string is :  I am a teacher and I love to inspire and teach people
Number of unique words: 10
Unique words :  {'teacher', 'people', 'inspire', 'and', 'to', 'love', 'I', 'am', 'teach', 'a'}
```

Output:

```
The Given string is :  I am a teacher and I love to inspire and teach people
Number of unique words: 10
Unique words :  {'teacher', 'people', 'inspire', 'and', 'to', 'love', 'I', 'am', 'teach', 'a'}
```

Explanation:

A string named str is declared as below:

str = "I am a teacher and I love to inspire and teach people"

Now the String is Stored in a set using the split() function.

Set = set(str.split(" "))

A set does not allow the duplicate elements. Hence when the string is stored in a set only the unique words will be present in that set.

Now, the number of unique words in the string can be found using the len() function on set:

print ("Number of unique words:", len(Set))

The unique words can be displayed by printing the set.

print ("Unique words: ", Set)

Question 7:

Solution:

```
In [39]: #Printing According to the pattern
         print("Name \t\t Age \t Country \t City")
         print("Asabeneh \t 250 \t Finland \t Helsinki")
```

Output:

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

Explanation:

The given pattern has been printed using the escape tab sequence “\t” and the second line is displayed using a new print function.

Machine Learning – Assignment 1

Question 8:

Solution:

```
In [48]:
#Using String formatting method

radius = 10
area = 3.14 * radius ** 2
print("The Area of Circle with radius {} is {} meters square.".format(radius,area))
```

Output:

```
The Area of Circle with radius 10 is 314.0 meters square.
```

Explanation:

The radius of the circle is declared as 10 meters and the area of the circle is calculated by using the formula 3.14 * (r**2). Now the area is formatted using the strung format() function.

The format() method formats the specified value(s) and insert them inside the string's placeholder.

Question 9:

Solution:

```
#An empty list named l which reads the weights of N students is declared

l = []

#Taking input from User

n = int(input("Enter number of weights : "))

# Appending the values to list l
for i in range(0, n):
    ele = int(input())
    l.append(ele) # adding the element

print("The weights of N students:", l)

#A list named Kg is declared which contains the values converted from lbs to Kilograms

kg=[]

for i in range(0, n):
    ele =0.45359237 * i
    kg.append(ele)

print("The weights of N students converted to Kilograms from lbs:", kg)
```

Machine Learning – Assignment 1

Output:

```
Enter number of weights : 5
3
5
7
9
2
The weights of N students: [3, 5, 7, 9, 2]
The weights of N students converted to Kilograms from lbs: [0.0, 0.45359237, 0.90718474, 1.3607771100000001, 1.81436948]
```

Explanation:

In this problem an empty list named l which reads the weights of N students is declared

l = []

The number of students is taken from the users as below:

n = int(input("Enter number of weights : "))

and the values are appended to the list as below:

“”for i in range(0, n):

ele = int(input())

l.append(ele) # adding the element”"

A list named Kg is declared which contains the values converted from lbs to Kilograms

kg=[]

1 lb = 0.45359237 kg.
This value is used for the converting lbs to kilograms as below:

for i in range(0, n):

ele =0.45359237 * i

kg.append(ele)