

# Machine Learning (Assignment # 1)

## Question 1

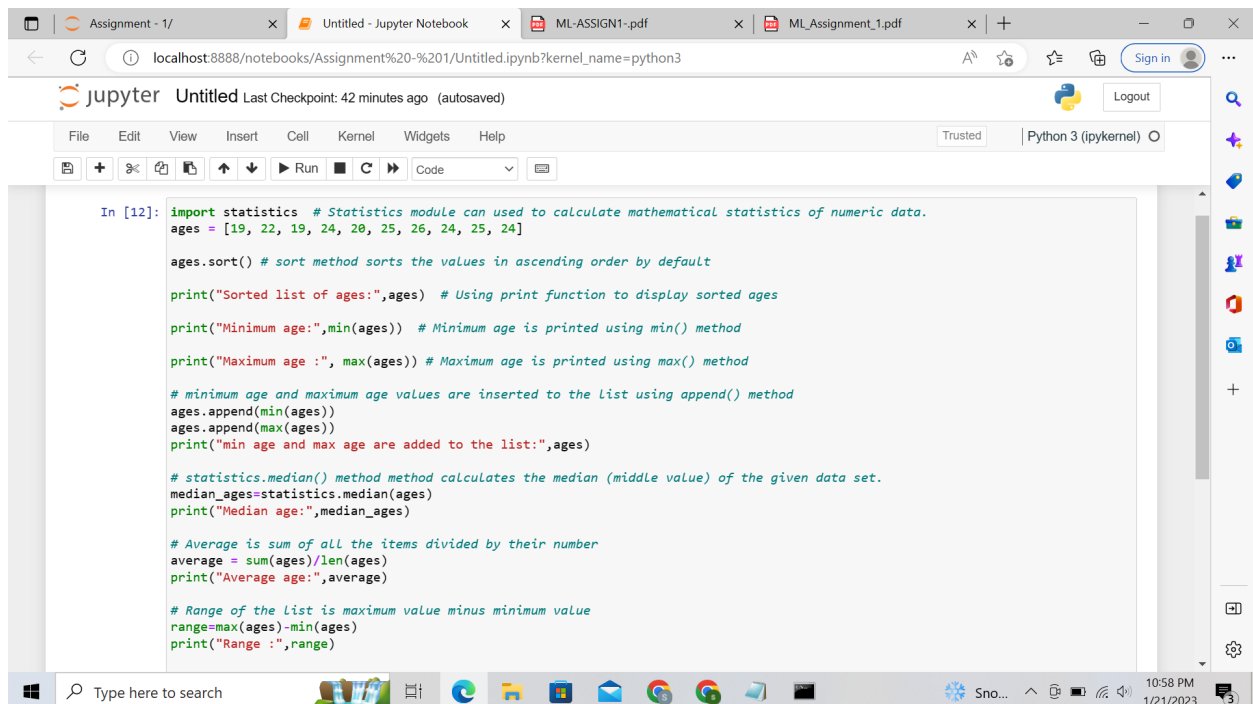
The following is a list of 10 students ages:

```
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)

Ans

Python code :



```
In [12]: import statistics # Statistics module can used to calculate mathematical statistics of numeric data.
ages = [19, 22, 19, 24, 20, 25, 26, 24, 25, 24]

ages.sort() # sort method sorts the values in ascending order by default

print("Sorted list of ages:", ages) # Using print function to display sorted ages

print("Minimum age:", min(ages)) # Minimum age is printed using min() method

print("Maximum age :", max(ages)) # Maximum age is printed using max() method

# minimum age and maximum age values are inserted to the List using append() method
ages.append(min(ages))
ages.append(max(ages))
print("min age and max age are added to the list:", ages)

# statistics.median() method calculates the median (middle value) of the given data set.
median_ages = statistics.median(ages)
print("Median age:", median_ages)

# Average is sum of all the items divided by their number
average = sum(ages)/len(ages)
print("Average age:", average)

# Range of the List is maximum value minus minimum value
range = max(ages) - min(ages)
print("Range :", range)
```

- 1) Python has a built-in module named statistics that we can use to calculate mathematical statistics of numeric data.
- 2) So we have imported the statistics module.
- 3) The ages list is sorted using the sort() method.
- 4) The minimum age and maximum age is printed using the min() and max() method.
- 5) The minimum age and maximum age is appended to the list using append() function.
- 6) The Statistics module has a median method to calculate the median of the list.
- 7) Average of the list can be calculated by adding all the items using sum() function and divide by the length of the list i.e number of items
- 8) Range of the list is the maximum value of the list minus the minimum value of the list.

## Output

The screenshot shows a Jupyter Notebook window titled "Untitled" with a last checkpoint 44 minutes ago. The interface includes a menu bar (File, Edit, View, Insert, Cell, Kernel, Widgets, Help), a toolbar with icons for file operations and execution, and a code editor. The code in the cell performs the following operations:

- Initializes a list `ages` with values `[19, 19, 20, 22, 24, 24, 24, 25, 25, 26]`.
- Appends the minimum and maximum values of `ages` to the list using `append()`.
- Prints the updated list: `print("min age and max age are added to the list:", ages)`.
- Calculates the median of `ages` using `statistics.median(ages)` and prints it: `print("Median age:", median_ages)`.
- Calculates the average of `ages` using `sum(ages)/len(ages)` and prints it: `print("Average age:", average)`.
- Calculates the range of `ages` using `max(ages)-min(ages)` and prints it: `print("Range :", range)`.

The output of the code execution is displayed below the code cell:

```
Sorted list of ages: [19, 19, 20, 22, 24, 24, 24, 25, 25, 26]
Minimum age: 19
Maximum age : 26
min age and max age are added to the list: [19, 19, 20, 22, 24, 24, 24, 25, 25, 26, 19, 26]
Median age: 24.0
Average age: 22.75
Range : 7
```

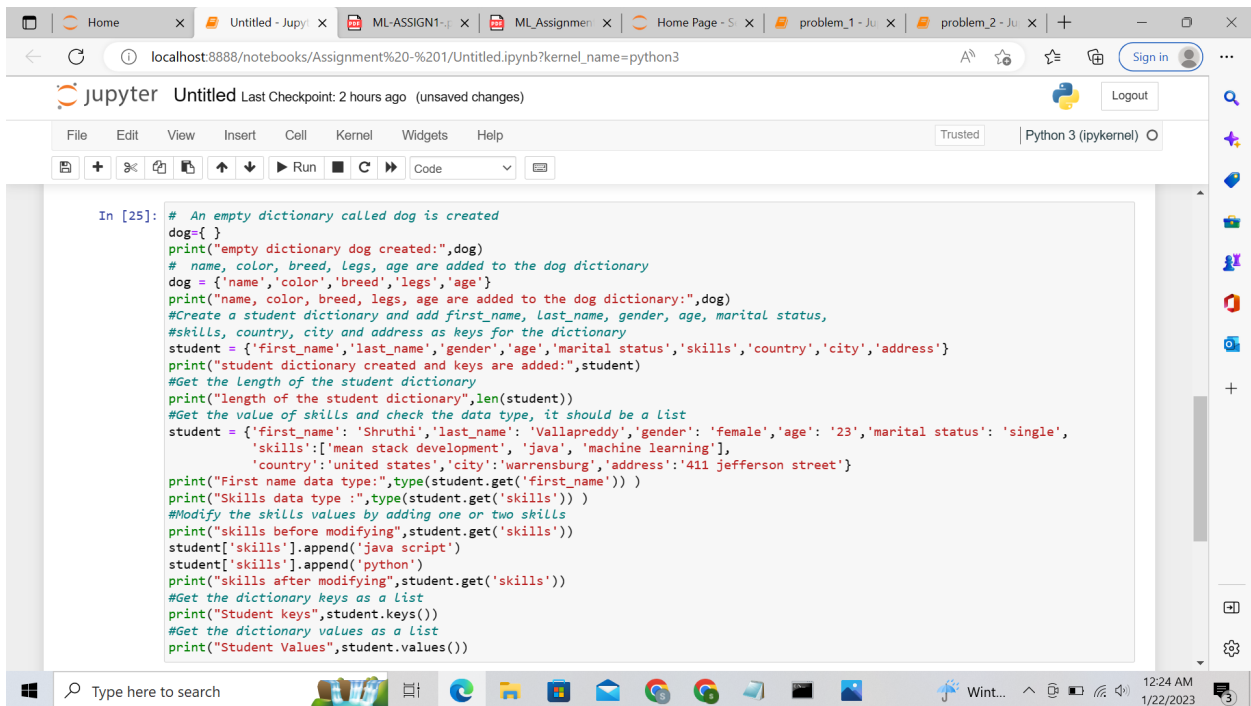
The bottom of the window shows the Windows taskbar with the search bar and various application icons. The system clock indicates 11:01 PM on 1/21/2023.

## 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

Ans

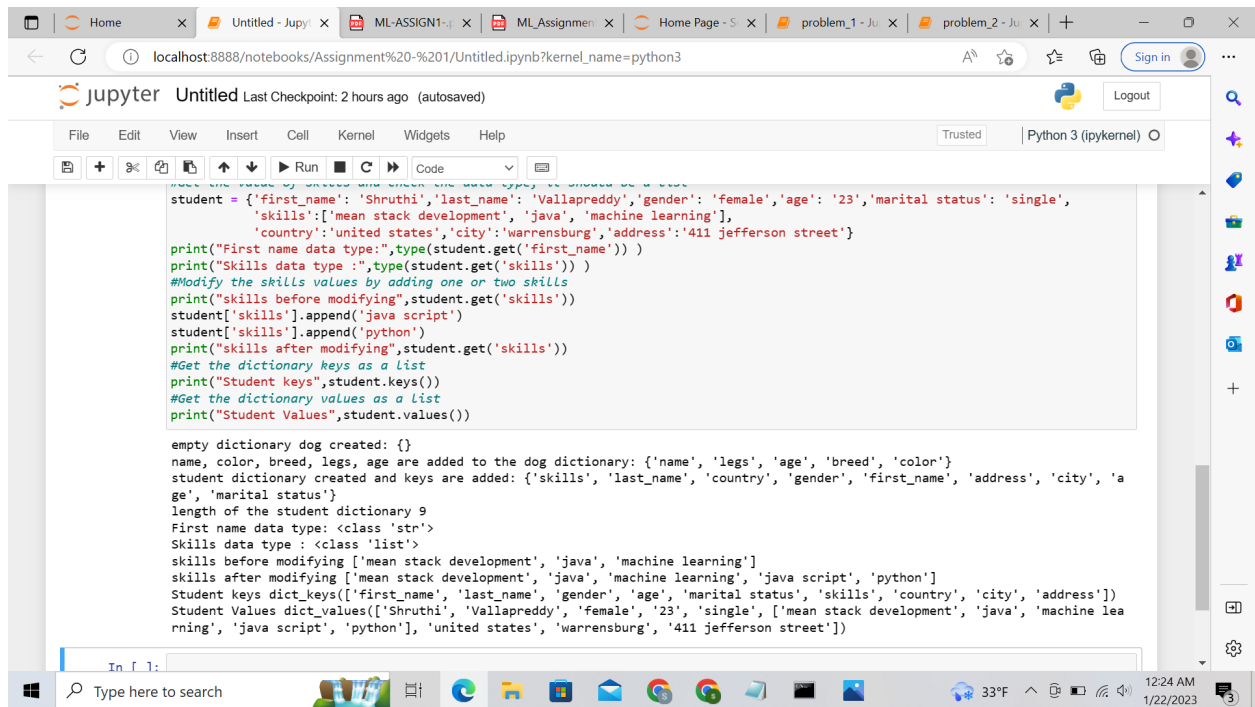
Python code



```
In [25]: # An empty dictionary called dog is created
dog={}
print("empty dictionary dog created:",dog)
# name, color, breed, legs, age are added to the dog dictionary
dog = {'name','color','breed','legs','age'}
print("name, color, breed, legs, age are added to the 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','last_name','gender','age','marital status','skills','country','city','address'}
print("student dictionary created and keys are added:",student)
#Get the length of the student dictionary
print("length of the student dictionary",len(student))
#Get the value of skills and check the data type, it should be a list
student = {'first_name': 'Shruthi','last_name': 'Vallapreddy','gender': 'female','age': '23','marital status': 'single',
'skills':['mean stack development','java','machine learning'],
'country':'united states','city':'warrensburg','address':'411 jefferson street'}
print("first name data type:",type(student.get('first_name')))
print("Skills data type :",type(student.get('skills')))
#Modify the skills values by adding one or two skills
print("skills before modifying",student.get('skills'))
student['skills'].append('java script')
student['skills'].append('python')
print("skills after modifying",student.get('skills'))
#Get the dictionary keys as a list
print("Student keys",student.keys())
#Get the dictionary values as a list
print("Student Values",student.values())
```

1. Empty dictionary named dog is created.
2. Name, color, breed, legs, age to the dog dictionary are added to the dictionary
3. Student dictionary is created and keys are added to the dictionary
4. The length of the dictionary is printed using len() function
5. The values are given to the dictionary. The key skills have the value type of list.
6. The values of skills field is modified by adding two more skills
7. Dictionary keys are printed as a list using student.keys().
8. Dictionary values are printed as a list using student.values().

## Output



The screenshot displays a Jupyter Notebook interface in a web browser. The browser's address bar shows the URL: `localhost:8888/notebooks/Assignment%20-%201/Untitled.ipynb?kernel_name=python3`. The Jupyter interface includes a top bar with the 'jupyter' logo, the notebook title 'Untitled', and a 'Last Checkpoint: 2 hours ago (autosaved)' status. Below this is a menu bar with options: File, Edit, View, Insert, Cell, Kernel, Widgets, and Help. A toolbar contains icons for saving, opening, and running code. The main area shows a code cell with the following Python code:

```
student = {'first_name': 'Shruthi', 'last_name': 'Vallapreddy', 'gender': 'female', 'age': '23', 'marital status': 'single',
           'skills': ['mean stack development', 'java', 'machine learning'],
           'country': 'united states', 'city': 'warrensburg', 'address': '411 jefferson street'}

print("First name data type:", type(student.get('first_name')))
print("Skills data type :", type(student.get('skills')))
#Modify the skills values by adding one or two skills
print("skills before modifying", student.get('skills'))
student['skills'].append('java script')
student['skills'].append('python')
print("skills after modifying", student.get('skills'))
#Get the dictionary keys as a list
print("Student keys", student.keys())
#Get the dictionary values as a list
print("Student Values", student.values())
```

The output of the code is displayed below the code cell:

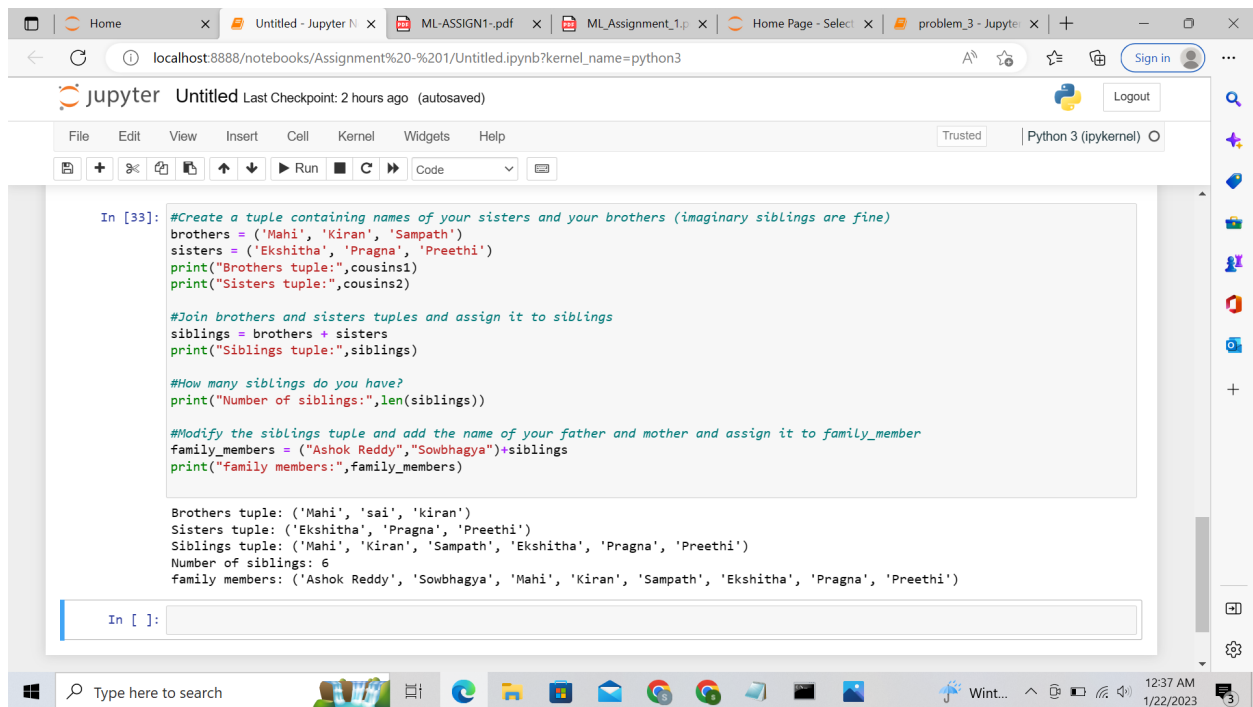
```
empty dictionary dog created: {}
name, color, breed, legs, age are added to the dog dictionary: {'name', 'legs', 'age', 'breed', 'color'}
student dictionary created and keys are added: {'skills', 'last_name', 'country', 'gender', 'first_name', 'address', 'city', 'a
ge', 'marital status'}
length of the student dictionary 9
First name data type: <class 'str'>
Skills data type : <class 'list'>
skills before modifying ['mean stack development', 'java', 'machine learning']
skills after modifying ['mean stack development', 'java', 'machine learning', 'java script', 'python']
Student keys dict_keys(['first_name', 'last_name', 'gender', 'age', 'marital status', 'skills', 'country', 'city', 'address'])
Student Values dict_values(['Shruthi', 'Vallapreddy', 'female', '23', 'single', ['mean stack development', 'java', 'machine lea
rning', 'java script', 'python'], 'united states', 'warrensburg', '411 jefferson street'])
```

The bottom of the image shows a Windows taskbar with the search bar, task view button, and several open applications including Edge, File Explorer, and Chrome. The system clock indicates 12:24 AM on 1/22/2023.

### 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

Ans



The screenshot shows a Jupyter Notebook window titled 'Untitled' with a Python 3 (ipykernel) environment. The code in the cell is as follows:

```
In [33]: #Create a tuple containing names of your sisters and your brothers (imaginary siblings are fine)
brothers = ('Mahi', 'Kiran', 'Sampath')
sisters = ('Ekshitha', 'Pragna', 'Preethi')
print("Brothers tuple:",cousins1)
print("Sisters tuple:",cousins2)

#Join brothers and sisters tuples and assign it to siblings
siblings = brothers + sisters
print("Siblings tuple:",siblings)

#How many siblings do you have?
print("Number of siblings:",len(siblings))

#Modify the siblings tuple and add the name of your father and mother and assign it to family_member
family_members = ("Ashok Reddy", "Sowbhagya")+siblings
print("family members:",family_members)
```

The output of the code is displayed below the cell:

```
Brothers tuple: ('Mahi', 'sai', 'kiran')
Sisters tuple: ('Ekshitha', 'Pragna', 'Preethi')
Siblings tuple: ('Mahi', 'Kiran', 'Sampath', 'Ekshitha', 'Pragna', 'Preethi')
Number of siblings: 6
family members: ('Ashok Reddy', 'Sowbhagya', 'Mahi', 'Kiran', 'Sampath', 'Ekshitha', 'Pragna', 'Preethi')
```

- 1) Brothers and sisters tuple is created and assigned some values
- 2) Brothers and sisters tuples are added and assigned to siblings tuple
- 3) Total number of siblings can be found out using the len() function.
- 4) Mother and father name is added to the siblings tuple and assigned to family\_members tuple

## Question 4

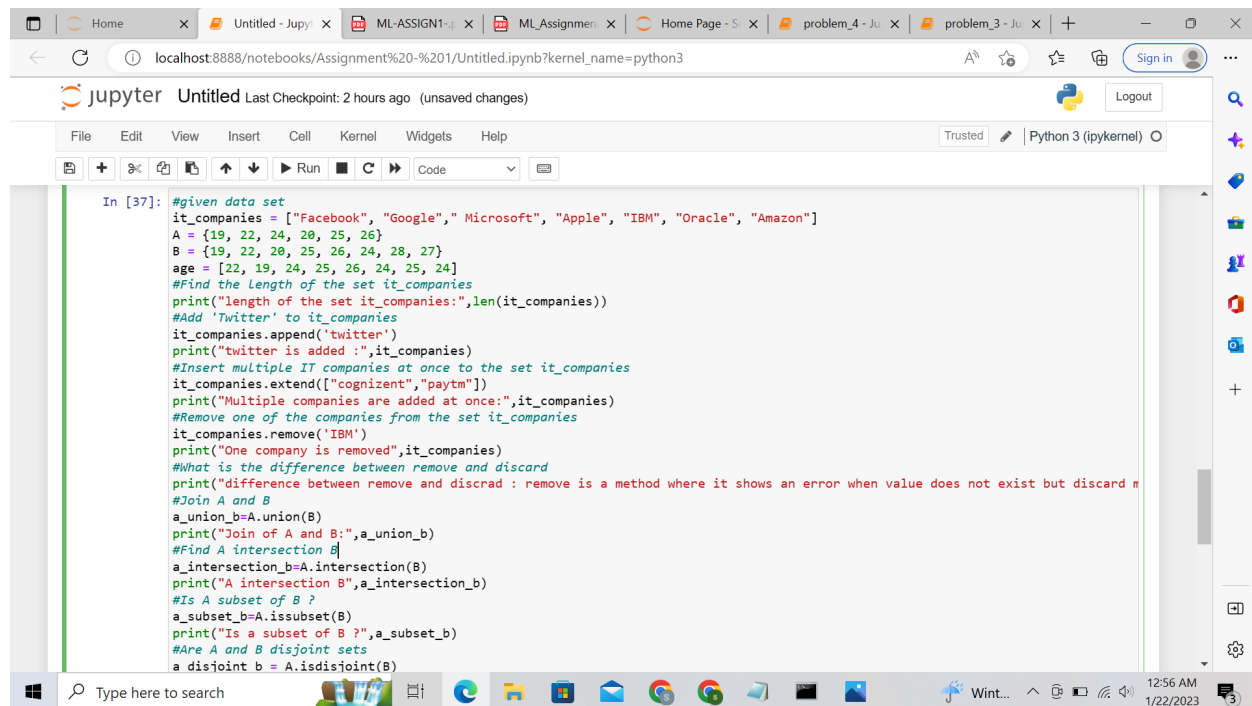
it\_companies = {'Facebook', 'Google', 'Microsoft', 'Apple', 'IBM', 'Oracle', 'Amazon'}

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.



The screenshot shows a Jupyter Notebook window with the following code in a cell:

```
In [37]: #given data set
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
print("length of the set it_companies:",len(it_companies))
#Add 'Twitter' to it_companies
it_companies.append('twitter')
print("twitter is added :",it_companies)
#Insert multiple IT companies at once to the set it_companies
it_companies.extend(["cognizent","paytm"])
print("Multiple companies are added at once:",it_companies)
#Remove one of the companies from the set it_companies
it_companies.remove('IBM')
print("One company is removed",it_companies)
#What is the difference between remove and discard
print("difference between remove and discard : remove is a method where it shows an error when value does not exist but discard m
#Join A and B
a_union_b=A.union(B)
print("Join of A and B:",a_union_b)
#Find A intersection B
a_intersection_b=A.intersection(B)
print("A intersection B",a_intersection_b)
#Is A subset of B ?
a_subset_b=A.issubset(B)
print("Is a subset of B ?",a_subset_b)
#Are A and B disjoint sets
a_disjoint_b = A.isdisjoint(B)
```

- 1) Sets are written with curly brackets, set is a collection which is unordered, unchangeable, and unindexed
- 2) To add one item to a set use the add() method and To add items from another set into the current set, use the update() method
- 3) To remove an item in a set, use the remove() or the discard() method. remove method raises an error when value doesn't exist whereas discard method doesn't raise an error

```
print("Are A and B disjoint sets:",a_disjoint_b)
#What is the symmetric difference between A and B
a_symmetric_difference_b = A.symmetric_difference(B)
print("symmetric difference between A and B:",a_symmetric_difference_b)
#Delete the sets completely
A.clear()
B.clear()
print ("A set after deleting:",A)
print ("B set after deleting:",B)
#Convert the ages to a set and compare the length of the list and the set.
age_set=set(age)
print(age_set)
print ("difference between length of age",len(age)-len(age_set))

length of the set it_companies: 7
twitter is added : ['Facebook', 'Google', 'Microsoft', 'Apple', 'IBM', 'Oracle', 'Amazon', 'twitter']
Multiple companies are added at once: ['Facebook', 'Google', 'Microsoft', 'Apple', 'IBM', 'Oracle', 'Amazon', 'twitter', 'cognizent', 'paytm']
One company is removed ['Facebook', 'Google', 'Microsoft', 'Apple', 'Oracle', 'Amazon', 'twitter', 'cognizent', 'paytm']
difference between remove and discard : remove is a method where it shows an error when value does not exist but discard method does not show an error.
Join of A and B: {19, 20, 22, 24, 25, 26, 27, 28}
A intersection B {19, 20, 22, 24, 25, 26}
Is a subset of B ? True
Are A and B disjoint sets: False
symmetric difference between A and B: {27, 28}
A set after deleting: set()
```

```
length of the set it_companies: 7
twitter is added : ['Facebook', 'Google', 'Microsoft', 'Apple', 'IBM', 'Oracle', 'Amazon', 'twitter']
Multiple companies are added at once: ['Facebook', 'Google', 'Microsoft', 'Apple', 'IBM', 'Oracle', 'Amazon', 'twitter', 'cognizent', 'paytm']
One company is removed ['Facebook', 'Google', 'Microsoft', 'Apple', 'Oracle', 'Amazon', 'twitter', 'cognizent', 'paytm']
difference between remove and discard : remove is a method where it shows an error when value does not exist but discard method does not show an error.
Join of A and B: {19, 20, 22, 24, 25, 26, 27, 28}
A intersection B {19, 20, 22, 24, 25, 26}
Is a subset of B ? True
Are A and B disjoint sets: False
symmetric difference between A and B: {27, 28}
A set after deleting: set()
B set after deleting: set()
{19, 22, 24, 25, 26}
difference between length of age 3
```

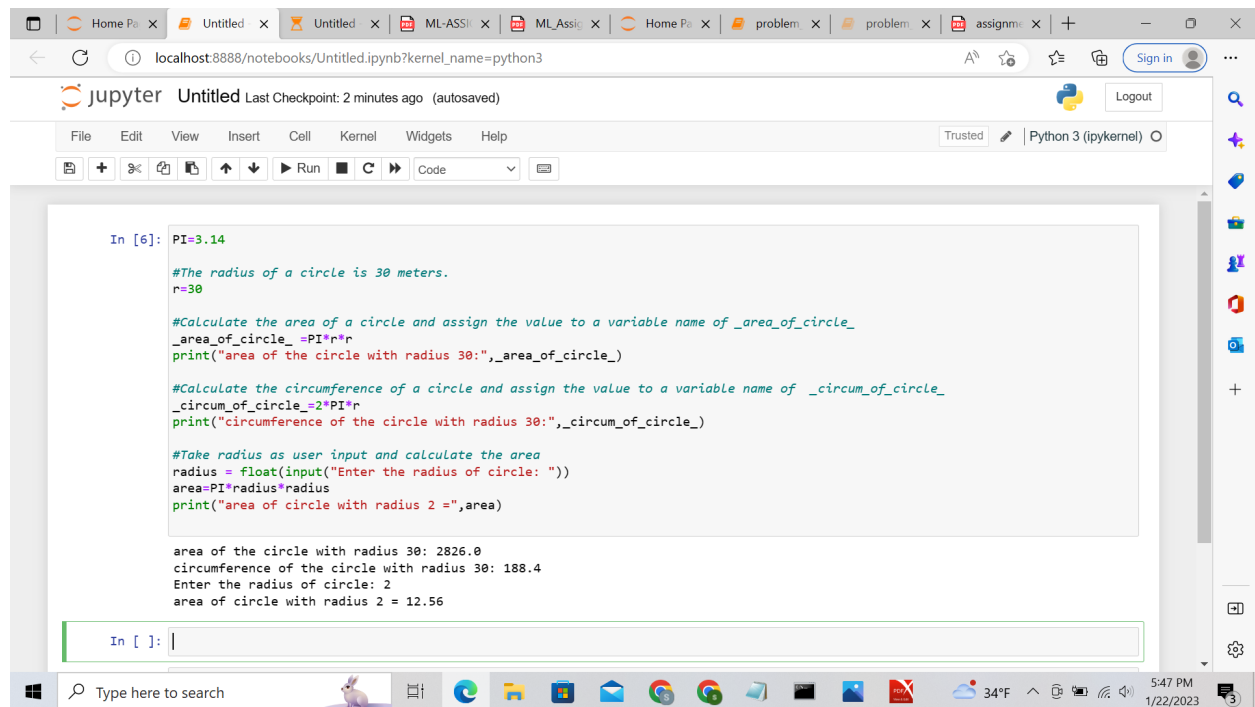
- 4) Union() return a set that contains all items from both sets, duplicates are excluded, intersection() returns a set that contains the items that exist in both set A, and set B,issubset() return True if all items in set x are present in set y, is disjoint() return True if no items in set x is present in set symmetric \_difference() return a set that contains all items from both sets, except items that are present in both sets
- 5) del keyword will delete the set completely

## 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

Ans



The screenshot shows a Jupyter Notebook running in a web browser. The notebook has a single code cell with the following Python code:

```
In [6]: PI=3.14

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

#Calculate the area of a circle and assign the value to a variable name of _area_of_circle_
_area_of_circle_ =PI*r*r
print("area of the circle with radius 30:",_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*r
print("circumference of the circle with radius 30:",_circum_of_circle_)

#Take radius as user input and calculate the area
radius = float(input("Enter the radius of circle: "))
area=PI*radius*radius
print("area of circle with radius 2 =",area)
```

The output of the code is displayed below the code cell:

```
area of the circle with radius 30: 2826.0
circumference of the circle with radius 30: 188.4
Enter the radius of circle: 2
area of circle with radius 2 = 12.56
```

The Jupyter Notebook interface includes a menu bar (File, Edit, View, Insert, Cell, Kernel, Widgets, Help), a toolbar with icons for file operations and execution, and a sidebar with various tool icons. The browser window shows the URL `localhost:8888/notebooks/Untitled.ipynb?kernel_name=python3`.

- 1) `*` is a multiplication operator.
- 2) And by using the formula of area and circumference of the circle we can calculate those by using mathematical operators
- 3) `input()` allows the user to pass the dynamic input.
- 4) And the area is calculated and printed

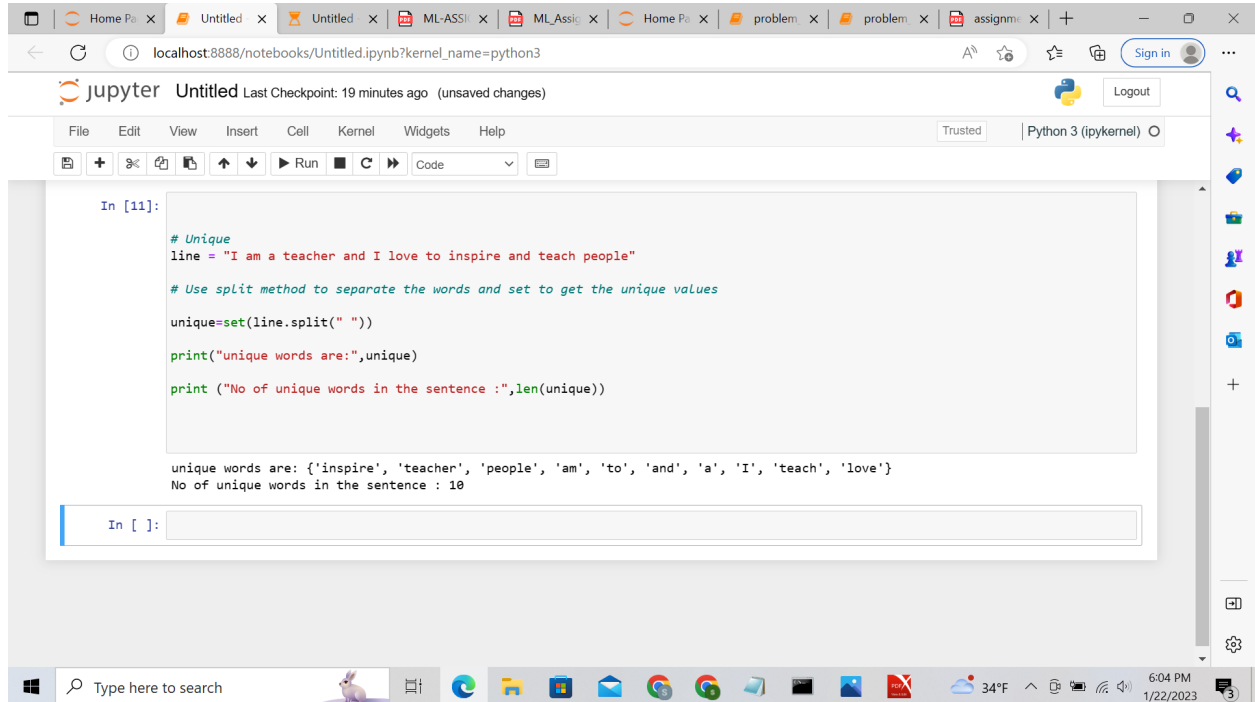


## Question 6

"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.

Ans



The screenshot shows a Jupyter Notebook running on a local host. The code cell contains the following Python code:

```
In [11]:  
  
# Unique  
line = "I am a teacher and I love to inspire and teach people"  
  
# Use split method to separate the words and set to get the unique values  
unique=set(line.split(" "))  
  
print("unique words are:",unique)  
  
print ("No of unique words in the sentence :",len(unique))
```

The output of the code is displayed below the cell:

```
unique words are: {'inspire', 'teacher', 'people', 'am', 'to', 'and', 'a', 'I', 'teach', 'love'}  
No of unique words in the sentence : 10
```

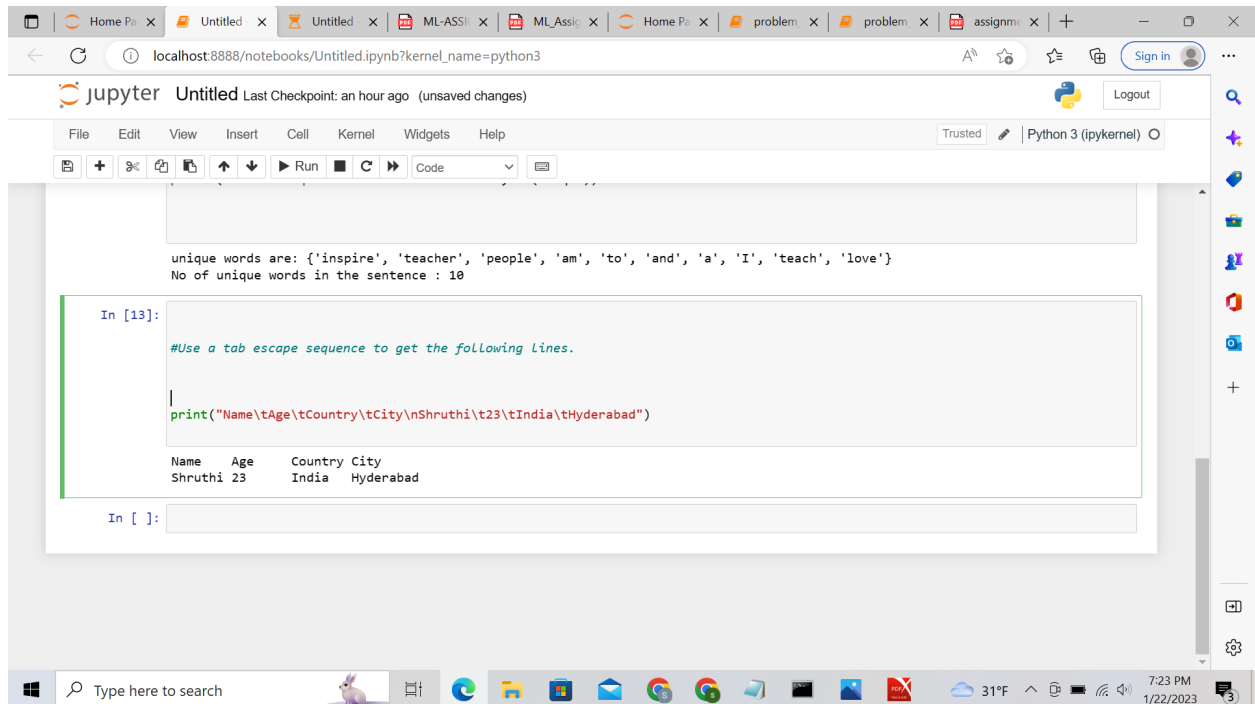
The Jupyter Notebook interface includes a menu bar (File, Edit, View, Insert, Cell, Kernel, Widgets, Help), a toolbar with icons for file operations and execution, and a sidebar with various tool icons. The browser tabs at the top show multiple open pages, including 'Home Page', 'Untitled', 'ML-ASSI', and 'problem'.

- 1) Line variable stores the sentence.
- 2) split (" ") is used to split the sentence using spaces
- 3) And those words are stored in the set.
- 4) Set only stores the unique words.
- 5) len() is used to find out the length of the set i.e number of unique words.

## Question 7

Use a tab escape sequence to get the following lines.

Name	Age	Country	City
Asabeneh	250	Finland	Helsinki



The screenshot shows a Jupyter Notebook window titled 'Untitled' with a Python 3 kernel. The code cell contains the following text:

```
unique words are: {'inspire', 'teacher', 'people', 'am', 'to', 'and', 'a', 'I', 'teach', 'love'}  
No of unique words in the sentence : 10  
  
In [13]:  
  
#Use a tab escape sequence to get the following lines.  
  
|  
print("Name\tAge\tCountry\tCity\nShruthi\t23\tIndia\tHyderabad")
```

The output of the code cell is a table:

Name	Age	Country	City
Shruthi	23	India	Hyderabad

- 1) To insert characters that are illegal in a string, use an escape character.
- 2) An escape character is a backslash \ followed by the character you want to insert.
- 3) \t is used for a tab space, \n for new line

## Question 8

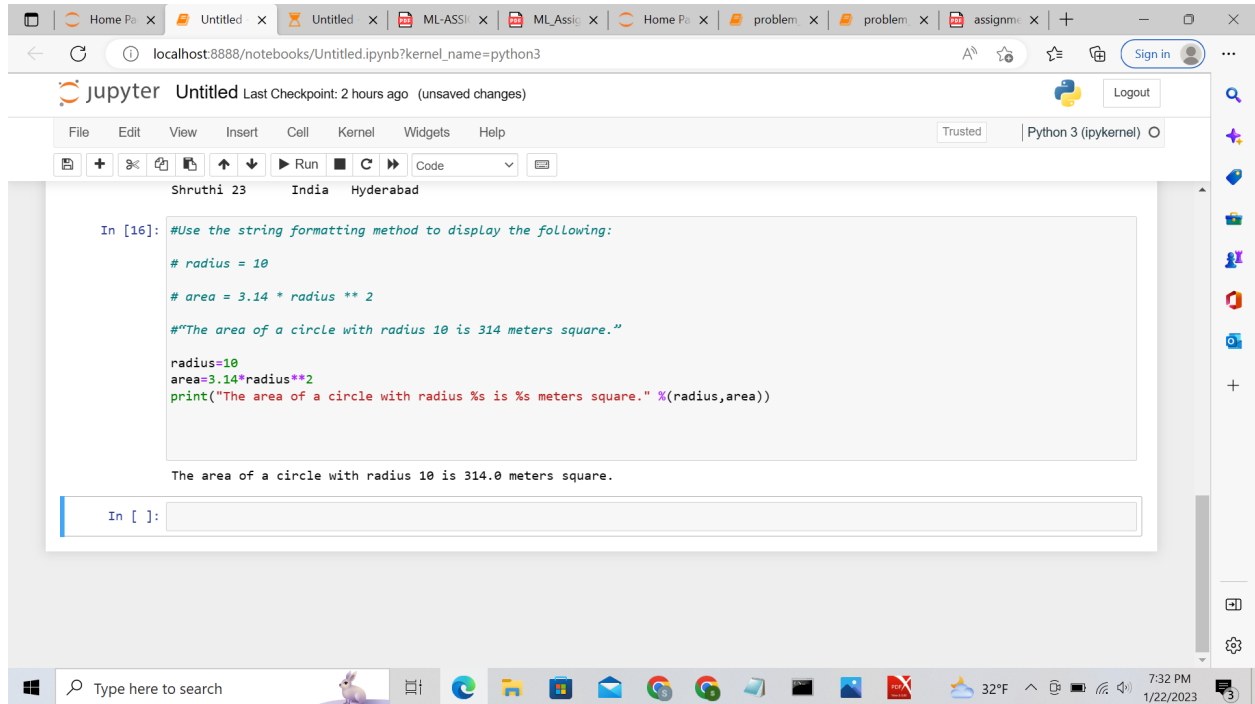
Use the string formatting method to display the following:

radius = 10

area = 3.14 \* radius \*\* 2

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

Ans



```
In [16]: #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=3.14*radius**2
print("The area of a circle with radius %s is %s meters square." % (radius,area))
```

The area of a circle with radius 10 is 314.0 meters square.

1) \*\* is a power operator

2) The % operator in python for strings is used for something called string substitution

### Question 9

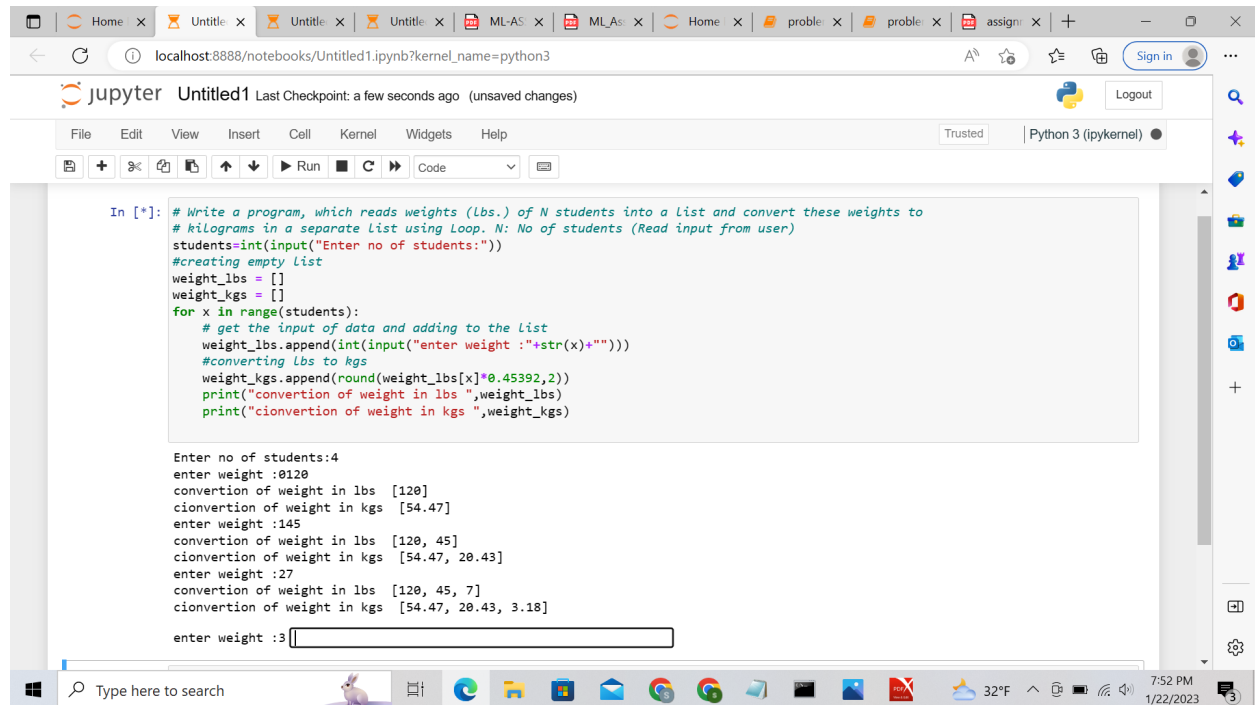
Write a program, which reads weights (lbs.) of N students into a list and convert these weights to kilograms in a separate list using Loop.

N: No of students (Read input from user)

Ex: L1: [150, 155, 145, 148]

Output: [68.03, 70.3, 65.77, 67.13]

Ans



The screenshot shows a Jupyter Notebook window titled 'Untitled1' with a Python 3 kernel. The code in the notebook is as follows:

```
In [*]: # 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)
students=int(input("Enter no of students:"))
#creating empty list
weight_lbs = []
weight_kgs = []
for x in range(students):
    # get the input of data and adding to the List
    weight_lbs.append(int(input("enter weight :"+str(x)+"")))
    #converting lbs to kgs
    weight_kgs.append(round(weight_lbs[x]*0.45392,2))
    print("conversion of weight in lbs ",weight_lbs)
    print("conversion of weight in kgs ",weight_kgs)
```

The output of the program is shown below the code:


```
Enter no of students:4
enter weight :0120
conversion of weight in lbs [120]
conversion of weight in kgs [54.47]
enter weight :145
conversion of weight in lbs [120, 45]
conversion of weight in kgs [54.47, 20.43]
enter weight :27
conversion of weight in lbs [120, 45, 7]
conversion of weight in kgs [54.47, 20.43, 3.18]
enter weight :3
```

1) Using the input() function allows user input

2) Using the for loop we will iterate the N no. of times and collect the students weights in lbs. and store it in the list after that performing the weight conversion calculations from lbs. to kgs the student weights in kgs is stored in another list

## Question 10

The diagram below shows a dataset with 2 classes and 8 data points, each with only one feature value, labeled  $f$ . Note that there are two data points with the same feature value of 6. These are shown as two x's one above the other. Provide stepwise mathematical solution, do not write code for it.



1. Divide this data equally into two parts. Use first part as training and second part as testing. Using KNN classifier, for  $K=3$ , what would be the predicted outputs for the test samples? Show how you arrived at your answer.

2. Compute the confusion matrix for this and calculate accuracy, sensitivity and specificity values.

Ans

- 1) Divided the data equally into two parts i.e training set and test set
- 2) For  $K=3$  calculated the distance and predicted the output
- 3) Computed the confusion matrix for this.
- 4) And calculated the accuracy, sensitivity and specificity values

$$Tp=2$$

$$Tn=0$$

$$Fp=2$$

$$Fn=0$$

$$\text{Accuracy} = (Tp+Tn)/(Tn+Fp+Fn+Tp)$$

$$\text{Sensitivity} = Tp/(Tp+Fn)$$

$$\text{Specificity} = Tn/(Fp+Tn)$$

$$\text{Accuracy} = 0.5$$

$$\text{Sensitivity} = 1$$

$$\text{Specificity} = 0$$

10) Divide data equally into two parts

1, 2, 3, 7 as training and

6, 6, 10, 11 as testing

KNN classification :- for  $K=3$

Euclidean distance  $d = \sqrt{(w-w_1)^2 + (h-h_1)^2}$

→ distance from 6 to 1, 2, 3, 7 is

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

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

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

$$d_4 = \sqrt{(6-7)^2} = 1$$

}  $x_i$

As there are maximum no of (.),  
6 is changed from  $x$  to .

→ distance from 10 to 1, 2, 3, 7 is

$$d_1 = \sqrt{(10-1)^2} = 9$$

$$d_2 = \sqrt{(10-2)^2} = 8$$

$$d_3 = \sqrt{(10-3)^2} = 7$$

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

}  $x_i$

As there are maximum no of points (.)  
10 is not changed it is 10.

$\Rightarrow$  distance from 11 to 1, 2, 3, 7 is

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

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

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

$$d_4 = \sqrt{(11-7)^2} = 4$$

} x

As there are maximum no of (.), there is no change, it is same as.

2) Confusion matrix

	0	1
0	$T_N^{(0)}$	$FP(2)$
1	$FN_{(0)}$	$TP(2)$

$$\text{Accuracy} = (TP + TN) / (TN + FP + FN + TP)$$

$$= 2/2+2 = 2/4 = 0.5$$

$$\text{Sensitivity} = TP / (TP + FN) = \frac{2}{2+0} = 1$$

$$\text{Specificity} = TN / (FP + TN) = 0/0+2 = 0$$

