

# Machine Learning ICP-1

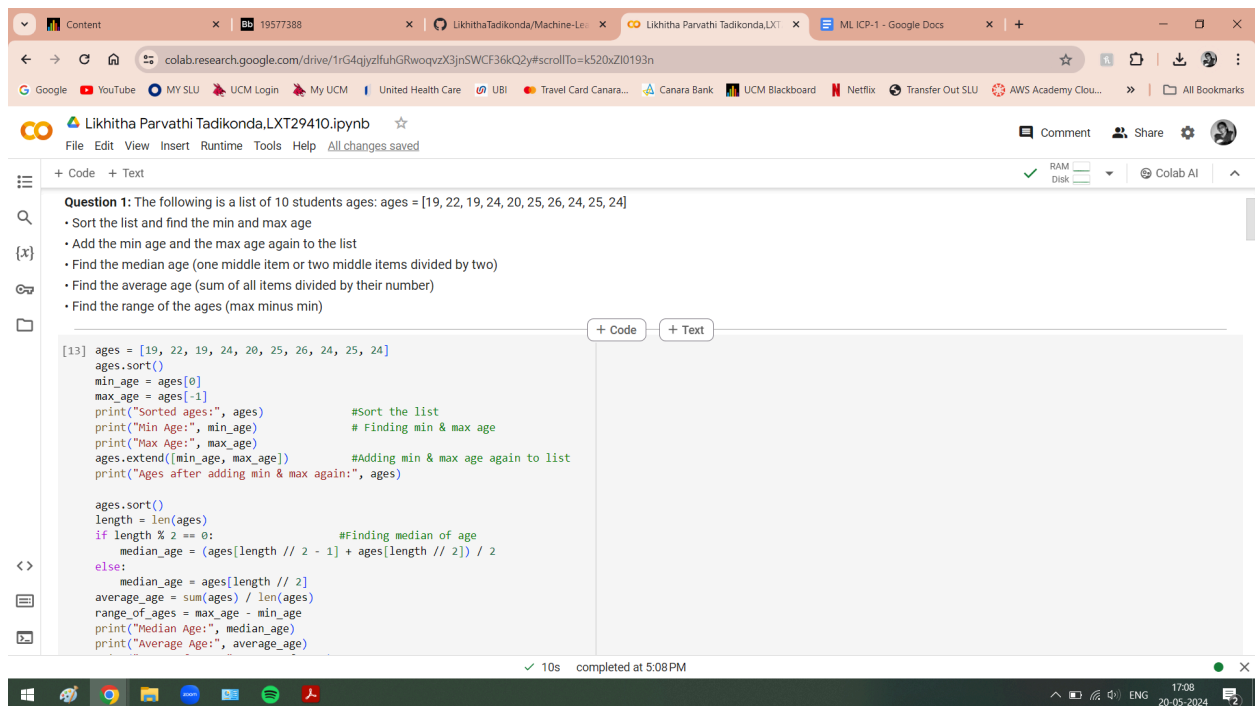
**-Likhitha Parvathi Tadikonda,  
700752941.**

- **GitHub Link:**

<https://github.com/LikhithaTadikonda/Machine-Learning-ICP-s.git>

- **Video Link:**

[https://drive.google.com/drive/folders/1ii1AJaUKDYBDz\\_F9QdKZte1cC-a6NW11?usp=sharing](https://drive.google.com/drive/folders/1ii1AJaUKDYBDz_F9QdKZte1cC-a6NW11?usp=sharing)



The screenshot shows a Google Colab notebook titled "Likhitha Parvathi Tadikonda.LXT29410.ipynb". The notebook contains a Python script that calculates various statistics for a list of ages. The script is as follows:

```
[13] ages = [19, 22, 19, 24, 20, 25, 26, 24, 25, 24]
ages.sort()
min_age = ages[0]
max_age = ages[-1]
print("Sorted ages:", ages)           #Sort the list
print("Min Age:", min_age)            # Finding min & max age
print("Max Age:", max_age)
ages.extend([min_age, max_age])       #Adding min & max age again to list
print("Ages after adding min & max again:", ages)

ages.sort()
length = len(ages)
if length % 2 == 0:                  #Finding median of age
    median_age = (ages[length // 2 - 1] + ages[length // 2]) / 2
else:
    median_age = ages[length // 2]
average_age = sum(ages) / len(ages)
range_of_ages = max_age - min_age
print("Median Age:", median_age)
print("Average Age:", average_age)
```

The script output shows the sorted ages, the minimum and maximum ages, the median age, the average age, and the range of ages. The notebook interface includes a toolbar with options like "Code", "Text", "Comment", "Share", and "Colab AI". The bottom status bar indicates the notebook was completed at 5:08 PM.

Content | 19577388 | LikhithaTadikonda/Machine-Le... | Likhitha Parvathi Tadikonda,LXT... | ML ICP-1 - Google Docs

colab.research.google.com/drive/1rG4qjyzfuhGRwoqvzX3jnSWCF36kQ2y#scrollTo=k520xZi0193n

Likhitha Parvathi Tadikonda,LXT29410.ipynb

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

```
[13] print("Range of ages:", range_of_ages)
```

Sorted ages: [19, 19, 20, 22, 24, 24, 24, 25, 25, 26]  
Min Age: 19  
Max Age: 26  
Ages after adding min & max again: [19, 19, 20, 22, 24, 24, 24, 25, 25, 26, 19, 26]  
Median Age: 24.0  
Average Age: 22.75  
Range of ages: 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

```
[14] dog = {} #Create an empty dictionary called dog
dog['name'] = 'Buddy' # Add name, color, breed, legs, age to the dog dictionary
dog['color'] = 'Brown'
dog['breed'] = 'Labrador'
dog['legs'] = 4
dog['age'] = 5
```

✓ 10s completed at 5:08 PM

Content | 19577388 | LikhithaTadikonda/Machine-Le... | Likhitha Parvathi Tadikonda,LXT... | ML ICP-1 - Google Docs

colab.research.google.com/drive/1rG4qjyzfuhGRwoqvzX3jnSWCF36kQ2y#scrollTo=k520xZi0193n

Likhitha Parvathi Tadikonda,LXT29410.ipynb

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

```
print("Dog dictionary:", dog)

student = { # Create a student dictionary, add name, gender, age, marital status, skills, country, city & address as keys for dictionary
    'first_name': 'John',
    'last_name': 'Doe',
    'gender': 'Male',
    'age': 22,
    'marital_status': 'Single',
    'skills': ['Python', 'Java'],
    'country': 'USA',
    'city': 'New York',
    'address': '123 Main St'
}
print("Student dictionary:", student)
student_length = len(student) #Get len of student dictionary
print("Len of student dictionary:", student_length)
skills = student['skills'] #Get the value of skills & check the data type, it should be a list
print("Skills:", skills)
print("Data type of skills:", type(skills))

student['skills'].extend(['C++', 'SQL']) #Modify the skills values by adding them
print("Updated skills:", student['skills'])

student_keys = list(student.keys())
print("Student dictionary keys:", student_keys)

student_values = list(student.values())
print("Student dictionary values:", student_values)
```

Dog dictionary: {'name': 'Buddy', 'color': 'Brown', 'breed': 'Labrador', 'legs': 4, 'age': 5}

✓ 10s completed at 5:08 PM

Content | 19577388 | LikhithaTadikonda/Machine-Le... | Likhitha Parvathi Tadikonda,LXT... | ML ICP-1 - Google Docs

colab.research.google.com/drive/1rG4qjyzfuhGRwoqvzX3jnSWCF36kQ2y#scrollTo=k520xZIO193n

Likhitha Parvathi Tadikonda,LXT29410.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Code Text

Student dictionary: {'first\_name': 'John', 'last\_name': 'Doe', 'gender': 'Male', 'age': 22, 'marital\_status': 'Single', 'skills': ['Python', 'Java'], 'country': 'USA', 'city': 'New York'}  
Len of student dictionary: 9  
skills: ['Python', 'Java']  
Data type of skills: <class 'list'>  
Updated skills: ['Python', 'Java', 'C++', 'SQL']  
Student dictionary keys: ['first\_name', 'last\_name', 'gender', 'age', 'marital\_status', 'skills', 'country', 'city', 'address']  
Student dictionary values: ['John', 'Doe', 'Male', 22, 'Single', ['Python', 'Java', 'C++', 'SQL'], 'USA', 'New York', '123 Main 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

```
sisters = ("Lisa", "Kate")
brothers = ("John", "Micheal")
siblings = sisters + brothers #Join brothers & sister tuples & assign it to siblings
print("Siblings:", siblings)
num_siblings = len(siblings) # How many siblings do you have?
print("Number of siblings:", num_siblings)

father = "James"
mother = "Mary"
family_members = siblings + (father, mother)
#Modify the siblings tuple & add the name of your father,mother & assign it to family_members
print("Family members:", family_members)
```

Siblings: ('Lisa', 'Kate', 'John', 'Micheal')

10s completed at 5:08 PM

Content | 19577388 | LikhithaTadikonda/Machine-Le... | Likhitha Parvathi Tadikonda,LXT... | ML ICP-1 - Google Docs

colab.research.google.com/drive/1rG4qjyzfuhGRwoqvzX3jnSWCF36kQ2y#scrollTo=k520xZIO193n

Likhitha Parvathi Tadikonda,LXT29410.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Code Text

Number of siblings: 4  
Family members: ('Lisa', 'Kate', 'John', 'Micheal', 'James', 'Mary')

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.

```
it_companies = {'Facebook', 'Google', 'Microsoft', 'Apple', 'IBM', 'Oracle', 'Amazon'}
A = {19, 22, 24, 20, 25, 26}
```

10s completed at 5:08 PM

Content | 19577388 | LikhithaTadikonda/Machine-Le... | Likhitha Parvathi Tadikonda,LXT... | ML ICP-1 - Google Docs

colab.research.google.com/drive/1rG4qjyzfuhGRwoqvzX3jnSWCF36kQ2y#scrollTo=k520xZi0193n

Likhitha Parvathi Tadikonda,LXT29410.ipynb

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

```
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]

it_companies_length = len(it_companies) #Find len of the set it_companies
print("Leng of it_companies:", it_companies_length)

it_companies.add('Twitter') #Add 'Twitter' to it_companies
print("it_companies after adding 'Twitter':", it_companies)

#Insert multiple IT companies at once to the set it_companies
it_companies.update(['Tesla', 'Netflix', 'Adobe'])
print("it_companies after adding multiple companies:", it_companies)

# Remove one of the companies from the set it_companies
it_companies.remove('Oracle')
print("it_companies after removing 'Oracle':", it_companies)

# Remove a company from the set if it exists, otherwise do nothing.
it_companies.discard('Yahoo')
print(it_companies)

A_union_B = A.union(B) #Join A and B
print("A union B:", A_union_B)

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

is_A_subset_B = A.issubset(B) # Is A subset of B
```

✓ 10s completed at 5:08 PM

Content | 19577388 | LikhithaTadikonda/Machine-Le... | Likhitha Parvathi Tadikonda,LXT... | ML ICP-1 - Google Docs

colab.research.google.com/drive/1rG4qjyzfuhGRwoqvzX3jnSWCF36kQ2y#scrollTo=k520xZi0193n

Likhitha Parvathi Tadikonda,LXT29410.ipynb

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

```
print("Is A subset of B:", is_A_subset_B)

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

A_join_B = A.union(B) # Join A with B and B with A
B_join_A = B.union(A)
print("A joined with B:", A_join_B)
print("B joined with A:", B_join_A)

A_symmetric_difference_B = A.symmetric_difference(B) #symmetric diff between A & B
print("Symmetric difference between A and B:", A_symmetric_difference_B)

del A # Delete the sets completely
del B
print("A and B sets deleted.")

age_set = set(age) # Convert ages to a set & compare the length of list & set.
print("Length of age list:", len(age))
print("Length of age set:", len(age_set))

Leng of it_companies: 7
it_companies after adding 'Twitter': {'IBM', 'Apple', 'Amazon', 'Twitter', 'Oracle', 'Microsoft', 'Facebook', 'Google'}
it_companies after adding multiple companies: {'IBM', 'Tesla', 'Amazon', 'Oracle', 'Facebook', 'Google', 'Adobe', 'Twitter', 'Microsoft', 'Apple', 'Netflix'}
it_companies after removing 'Oracle': {'IBM', 'Tesla', 'Amazon', 'Facebook', 'Google', 'Adobe', 'Twitter', 'Microsoft', 'Apple', 'Netflix'}
{'IBM', 'Tesla', 'Amazon', 'Facebook', 'Google', 'Adobe', 'Twitter', 'Microsoft', 'Apple', 'Netflix'}
A union 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: False
A joined with B: {19, 20, 22, 24, 25, 26, 27, 28}
```

✓ 10s completed at 5:08 PM

Content

19577388

LikhithaTadikonda/Machine-Le...

Likhitha Parvathi Tadikonda,LXT...

ML ICP-1 - Google Docs

colab.research.google.com/drive/1rG4qjyzfuhGRwoqvzX3jnSWCF36kQ2y#scrollTo=k520xZl0193n

GoogleYouTubeMY SLUUCM LoginMy UCMUnited Health CareUBITravel Card Canara...Canara BankUCM BlackboardNetflixTransfer Out SLUAWS Academy Clou...All Bookmarks

Likhitha Parvathi Tadikonda,LXT29410.ipynb

FileEditViewInsertRuntimeToolsHelpAll changes saved

CommentShareColab AI

+ Code+ Text

[16] B joined with A: {19, 20, 22, 24, 25, 26, 27, 28}  
Symmetric difference between A and B: {27, 28}  
A and B sets deleted.  
Length of age list: 8  
Length of age set: 5

**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

```
import math
radius = 30
_area_of_circle_ = math.pi * radius ** 2    #Calc area of circle with r=30 meters
print("Area of the circle:", _area_of_circle_)

_circum_of_circle_ = 2 * math.pi * radius    #Calc circumference of circle with r=30 meters
print("Circumference of the circle:", _circum_of_circle_)

user_radius = float(input("Enter the radius of the circle: ")) #Take radius as user input & calc the area
user_area_of_circle = math.pi * user_radius ** 2
print("Area of the circle with radius", user_radius, ":", user_area_of_circle)
```

Area of the circle: 2827.4333882308138  
Circumference of the circle: 188.49555921538757  
Enter the radius of the circle: 2  
Area of the circle with radius 2.0 : 12.566370614359172

10s completed at 5:08 PM

Windows

Google Chrome

File Explorer

Microsoft Edge

WhatsApp

Telegram

Signal

Discord

Spotify

Netflix

Amazon

Google

Microsoft

17:09

20-05-2024

ENG

Content x 19577388 x LikhithaTadikonda/Machine-Le... x Likhitha Parvathi Tadikonda,LXT... x ML ICP-1 - Google Docs x +

colab.research.google.com/drive/1rG4qjyzfuhGRwoqvzX3jnSWCF36kQ2y#scrollTo=k520xZi0193n

Google YouTube MY SLU UCM Login My UCM United Health Care UBI Travel Card Canara... Canara Bank UCM Blackboard Netflix Transfer Out SLU AWS Academy Clou... All Bookmarks

Likhitha Parvathi Tadikonda,LXT29410.ipynb ☆

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

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.

```
[17] sentence = "I am a teacher and I love to inspire and teach people"
      words = sentence.split()           #Split sentence into words
      unique_words = set(words)          #Create a set to store unique words
      num_unique_words = len(unique_words) #Print no. of unique words
      print("Number of unique words:", num_unique_words)
```

Number of unique words: 10

Question 7: Use a tab escape sequence to get the following lines. Name Age Country City Asabeneh 250 Finland Helsinki

```
[7] print("Name\tAge\tCountry\tCity\nAsabeneh\t250\tFinland\tHelsinki")
```

Name	Age	Country	City
Asabeneh	250	Finland	Helsinki

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."

```
[8] radius = 10
     area = 3.14 * radius ** 2
     print("Area of circle with radius {} is {} meters square.".format(radius, area))
```

Area of circle with radius 10 is 314.0 meters square.

10s completed at 5:08 PM

Content x 19577388 x LikhithaTadikonda/Machine-Le... x Likhitha Parvathi Tadikonda,LXT... x ML ICP-1 - Google Docs x +

colab.research.google.com/drive/1rG4qjyzfuhGRwoqvzX3jnSWCF36kQ2y#scrollTo=k520xZi0193n

Google YouTube MY SLU UCM Login My UCM United Health Care UBI Travel Card Canara... Canara Bank UCM Blackboard Netflix Transfer Out SLU AWS Academy Clou... All Bookmarks

Likhitha Parvathi Tadikonda,LXT29410.ipynb ☆

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

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)

Example:

L1: [150, 155, 145, 148]

Output: [68.03, 70.3, 65.77, 67.13]

```
N = int(input("Enter the number of students: ")) # Get no. of students from user
weights_lbs = [] # Create list to store weights in pounds
for i in range(N): # Get weights of the students in pounds
    weight_lbs = float(input(f"Enter the weight of student {i+1} in pounds: "))
    weights_lbs.append(weight_lbs)

weights_kg = [] # Create a list to store weights in kilograms
for weight_lbs in weights_lbs: # Convert weights from pounds to kilograms
    weight_kg = weight_lbs * 0.45359237
    weights_kg.append(round(weight_kg, 2))

# Print list of weights in kilograms
print("Weights in kilograms:", weights_kg)
```

Enter the number of students: 4  
Enter the weight of student 1 in pounds: 150  
Enter the weight of student 2 in pounds: 155  
Enter the weight of student 3 in pounds: 145  
Enter the weight of student 4 in pounds: 148  
Weights in kilograms: [68.04, 70.31, 65.77, 67.13]

10s completed at 5:08 PM