**Class 05: Python Data Structures: Lists, Tuples, Sets & Dictionaries**

* **Lists**

A **list** in Python is a **collection** of **multiple items** stored in a **single variable**.  
Lists are one of the most commonly used **data structures** in Python. (**lists are a sequential data structure**)

Example:

fruits=["Mango","Apple","Banana","Grape"]

print(fruits)

* **List Methods [ ]**

In Python, the append() method is used to add a new item at the end of a list.

Append-The first method to manage a list is append.

(Append means add new items)

**Append Methods:**

Append mening add ane element.

Example:

fruits = ["Mango", "Apple", "Banana", "Grape"]

fruits.append("Orange")

print(fruits)

**Insert Methods**

Example:

fruits=["Mango","Apple","Banana","Grape"]

fruits.insert(2,"Orange")

print(fruits)

**Extand Methods**

Example:

fruits=["Mango","Apple","Grape"]

fruits.extend(["Banana","Orange"])

print(fruits)

**Remove Methods**

Example:

fruits=["Mango","Apple","Grape"]

fruits.remove("Apple")

print(fruits)

**Pop Methods:**

Example:

fruits=["Mango","Apple","Grape"]

fruits.pop()

print(fruits)

**Clear Methods:**

Example:

fruits=["Mango","Apple","Grape"]

fruits.clear()

print(fruits)

**Count Methods:**

Example:

fruits=["Apple","Cherry","Apple","Banana","Apple"]

apple\_count= fruits.count("Apple")

print(apple\_count)

Example:

name=["Tonmoy","Mugdho","Tonmoy","Somrat","Tonmoy","Somrat"]

name\_of\_T=name.count("Tonmoy")

name\_of\_S=name.count("Somrat")

print("The number of Tonmoy is :", name\_of\_T)

print("The number of Somrat is :", name\_of\_S)

Total\_count= name\_of\_T + name\_of\_S

print("Total count is:", Total\_count)

**Sort Methods:**

Example:

number=[2,1,6,3,7.9,8,10]

number.sort()

print(number)

**Reverse Methods:**

Example:

Number=[9,8,7,6,5,4,3,2,1]

Number.reverse()

print(Number)

**Sort And Reverse:**

Example:

Number=[2,4,6,8,1,3,5,7,9]

Number.sort()

Number.reverse()

print(Number)

**Slicing Methods**

Example:

Country=["Bangladesh","India","US","UK","China","Canada","France"]

print(Country[0:4])

print(Country[1:5])

print(Country[:6])

**List Use Case**

- Storing User Data

- Inventory Purchase, Sales, Order

- Add to cart, Invoice, dropdown

- User Data Collection

- Table, List.

* **Tuples ( )**

Tuples are list methods are same.

**"In Python, the append, Instead, extend, remove, pop, clear, sort, reverse method does not work with tuples."**

Example:

# Convert tuple to list

fruitsTuple = ["Apple", "Cherry", "Apple", "Banana"]

fruitsList = list(fruitsTuple)

print(fruitsList)

Example:

# Convert list to tuple and tuple to list

from typing import Tuple

fruitsTuple=["Apple", "Cherry", "Apple", "Banana"]

fruitsList=list(fruitsTuple)

fruitsTuple=tuple(fruitsList)

print(fruitsTuple)

print(fruitsList)

* **Sets { }**

Example:

#SET

# Use {} for set

fruits={"Cherry", "Apple", "Banana" }

#fruits.add("Mango")

##fruits.update("Orange","Lychee")

#fruits.remove("Banana")

#fruits.clear()

fruits.pop()

print(fruits)

* Set Methods

**Union**

Example:

set1={1,2,3}

set2={4,5,6}

result=set1.union(set2)

print(result)

**intersection**

Example:

set1={1,2,3}

set2={3,4,5}

result=set1.intersection(set2)

print(result)

* **Set Use Case**

-Removing Data

-Filtering Data

-Mathematical set operation

-Unique Elements

* **Dictionaries**

#keys: values pair data

student={

    "name":"Tonmoy",

    "age": 23,

    "Subject": "CSE"

}

#print(student)

#print(student["name"])

#print(student.keys())

#rint(student.values())

#student.update({"age":30})

#print(student)

#student.pop("age")

#print(student)

student.clear()

print(student)

* **Dictionary Use Case**

-Form Submit

-Web Get, Web Post

-MongoDB

-Caching

-Store, Retrieve

-Request, Response

-Database

-B2B Api

-Payment Gateway

-SMS Api

-Email Api

-Nested Semi-Structured Data

-Rest Api

-Every Where In Web Engineering