Dt-15/02/2023

CORE:

Sequential Data Type:

- A sequential data type can hold multiple heterogeneous value and these values are stored in series of internal memory location.
- These internal memory locations are identified by concept of indexing.

These sequential data types are also referred as

- Series data type
- Compound data type
- Collection data type

❖ **Data Structures**: It is an object which hold the data during program execution time.

Life of this data structure with data is as long as program is executing.

❖ **Containers:** It hold the values.

Types of Sequential Data Type:

- List
- Tuple
- Set
- Dictionary
- String

LIST

- It is sequential data type which holds the multiple heterogeneous value.
- These values are closed by square bracket [] separated by comma.
- These value inside the square brackets are referred as **Elements**

- Using List function (list()), we can convert any value into list.

Features:

- Multiple Heterogeneous Values
- Mutable (Changeable)
- Duplicate are allows
- Order is preserved (Input Order is preserved)

- Indexing is possible
- Slicing is possible
- Nested list is possible

CRUD-Create Read Update Delete

Create List:

- Manually Created
- Using List Function
- Automated Way

Read Data from List:

- For Loop
- Indexing
- Slicing

Update:

- Inserting a value into list.
    - Append(): It accepts only one value as elements of list.
        - It append at the last.
    - Extend(): It accepts multiple value in the form of sequential data type and insert these multiple values as elements of list.
    - Insert(): insert(<index number,<Values>>)

Delete:

ASSIGNMENT:

Remove the duplicates.

For x in list1:

X

For y in list1

If x==y:

Not append

Dt-16.02.2023

Replace: It is insert a value followed by index number.

QUES MAIN
Flating the list(Flatering)

Extract Multi nested list

Add multiple value in their desire list

List comprehension

Shallow, deep copy

Delete:

- **Del():** del list[2]- It delete the value of second index. If we donot give index num,it delete the whole list.
- **Remove():** remove<Value>- it deletes a particular elements using the name of value.
- **Pop():** It deletes particular value based on

GENERIC FUNCTION:

- **Len():** length of list
- **Max():** returns maximum value
- **Min():** returns minimum value

List Specific Function:

- **Sort():** sort the list in ascending order
- **Count():** count instances of a particular value
- **Clear():** clear all the elements of list but structure still exist for further use.

ASSIGNMENT: Accept five values from user first using loop and then creat a list of all five values.

# 1.Reverse each elements (Apple-elppA)

# 2.Rotate the values inside the list by 2 places

# 3."Apple"--take first char of Apple,put it nested list[[A,Apple],[],[]]

Flatering:

List Comprehension:

Dt-21/02/2023

List Comprehension with for loop:

[Output Expression Input Sequence For loop]

Nested List Comprehension:

[[Output Expression Input Sequence] Input Sequence]

Operations:

COPY:

REFERENCE COPY:

- It is a process of creating an instance of the original data in a different variable.
- But in-turn both variables are pointing to same memory location.
- Assigning a value within another variable.

A=B

SHALLOW COPY:

- It is a process of copying values from one memory location to another memory location.
- It is implemented by. copy()
  - **Syntax-1:**

    A1=[10,20,30,40]

    B1=A1.copy()

  - **Syntax-2:**

    Import copy

    P=copy.copy(s)

DEEP COPY:

- It is a process of copying values from one memory location to another memory location including nested list values.
- **Syntax:**

  Import copy

  Y=copy.deepcopy(x)

Dt-22/02/2023

TUPLE:

It is a sequential data type which can hold multiple heterogeneous values separated by comma and enclosed in parenthesis ().

Tuple()—This

Eg: (100,200,300,400)

- Parenthesis are optional

Eg: 100,200,300,400

Properties of TUPLE/Features of TUPLE:

- Multiple heterogeneous values.
- Immutable.
- Duplicates are allowed.
- Indexing is possible.
- Slicing is possible.
- Nesting is possible.

CRUD:

Create:

Manually

Tuple Function

Read:

For loop

Indexing

Slicing

Update:

Insert

Delete

Replace

- None of them is possible as tuple is immutable.

Delete:

We can delete entire tuple by using ‘del’ keyword.

Zip Function:

It extract the value from common index of iterable.

Dt-23/02/2023

SET

It is a sequential data type which holds multiple heterogeneous values separated by comma and enclosed in braces.

Set—{100,200,300,400}

Properties of Set:

- It holds the multiple heterogeneous value.
- It is mutable.
- Duplicates are not allowed.
- Order is not preserved.
- Indexing is not possible.
- Slicing is not possible.
- Nesting partially possible.

CRUD:

Create:

- Empty set:

  Set()

Creating a normal set,

- Manually
- Set Function
- Automated

Read:

- For loop

Update:

- Insertion
  - Add()
  - Update()
  - Replace—No replacement

Delete:

DICTIONARY:

- It is a sequential data type which holds data in the form of key-value pair.
- Each key-value pair is separated by comma and enclosed in braces.
- Each key-value pair is referred as **item**.

{KEY:VALUE}

{

“A”: ”APPLE”, --- Key-Value Pair/Item

}

Key “A” is associated with value “APPLE”

Key “A” is mapped with value “APPLE”

- As we use Key “A” is associated with value “APPLE”, so dictionary can be referred as **Associative Array**.

Purpose of Learning Dictionary:

- It is a very powerful data structure.
- Dictionary is a replica of JSON

- JSON is used in REST API
- JSON is used Communication between servers
- JSON is default language in MongoDB

- To display KEYs---keys()
- To display Values—values()
- To display items—items()

Properties :

- Multiple heterogeneous values in the form of Key-Value Pair.
- Mutable
- Duplicates
  - Duplicates keys are not allowed.
  - Duplicates values are allowed.
- Indexing is not allowed
- Slicing is not allowed

CRUD:

Create:

Empty Dictionary:

  - {}
  - Dict()

Creating normal Dictionary:

- Manually
- Dict()
  - Nested tuple format
  - Variable declaration Format
- Automated

Reading:

- For loop
- Using Keys
- Using get()
- Using setdefault()

Update:

Dt-24/02/2023

Update:

- Insertion
  - Inserting a value
- Setdefault
- Replace
  - Assigning a new value in a existing key

Delete:

- Del
- Pop(<Key>)
- Popitems()
- Clear

Update:

**==Variable length argument

Dictionary Comprehension:

It is elegant way of creating a dictionary.

Two major components:

Input sequence

Source of input: For loop

Output expression

Business Logic—Output which is stored in dictionary.

Syntax:

{<Output Expression>   <Input Sequence>}

Dt-27/02/23:

Dictionary Comprehension:

Dictionary comprehension with if..Condition:

{<Output> <Input Sequnce> <Condition>}

Dictionary comprehension with if..else Condition:

{<Output> <if..else> <Input Sequnce>}

Dictionary comprehension with for loop Condition:

{<Output> <Input Sequnce> <for loop>}

Nested Dictionary: