DATA STRUCTURES IN PYTHON

There are Four inbuilt Data Structures in Python. They are:

1. LIST
2. TUPLE
3. SET
4. DICTIONARY

LIST :

- Lists are always enclosed with square brackets [ ]
- List is mutable (changeable)
- We can add multiple data types in a list
- It allows Duplicate values
- It allows Indexing –(Forward Indexing, Backward Indexing)
- Forward Indexing means where the index starts from left to right
- Backward Indexing means where the index starts from right to left
- Slicing is allowed in lists
- List is Growable

LIST FUNCTIONS:

- It allows only one argument for append(), count(), index(), and remove() functions
- It allows only two arguments for insert() function
- .append() : It is used to add a value at the end of a list
- .clear() : it is used to remove all elements from a list. It modifies the list-in-place and leaves it empty.
- .copy() : It returns a new list with the same elements as the original list
- .count() : It is used to count the number of occurrences of a specific element in a list
- .extend() : It is used to extend a list by appending elements from another iterable (such as list, tuple, or string) to the end of the original list
- .index() : It is used to find the index of the specific value in the list
- .insert() : It is used to insert an element at a specific position in the list
- .pop() : It is used to remove a value at the end of the list
- .remove() : It is used to remove a specific value in the list
- .reverse() : It is used to reverse the order of elements in the list
- .sort() : It is used to sort the elements of a list in ascending order

TUPLE :

- Tuple is always enclosed with open brackets ()
- Tuple is immutable (Unchangeable)
- We can add multiple data types in a Tuple
- It allows Duplicate Values
- It allows Indexing (Forward and Backward Indexing)
- Slicing is allowed in Tuples
- Tuple is Growable

TUPLE FUNCTIONS :

- .count() : It is used to count the number of occurrences of a specific element in a list
- .index() : It is used to find the index of the specific value in the list

SET :

- Set is always enclosed with curly braces {}
- We can add multiple data types in a Set
- Duplicate values are not allowed
- Slicing is not allowed
- Indexing is not allowed
- Set is an ordered format
- Set is Growable

SET FUNCTIONS :

- Only one argument is allowed in set
- .add() : The add() method is used to add elements to a set.
- .clear() : The clear() method is used to remove all elements from a set. It modifies the set in-place and leaves it empty.
- .copy() : It returns a new set with the same elements as the original set
- .difference() : The difference() method is used to find the set difference between two sets. It returns a new set that contains elements from the first set that are not present in the second set.
- .discard() : The difference() method is used to find the set difference between two sets. It returns a new set that contains elements from the first set that are not present in the second set.
- .intersection() : The intersection() method is used to find the intersection of two or more sets. It returns a new set that contains the common elements present in all the sets.
- .isdisjoint() : The isdisjoint() method is used to check if two sets are disjoint, i.e., if they have no common elements. It returns True if the sets have no intersection (no common elements), and False otherwise.
- .issubset() : The issubset() method is used to check if a set is a subset of another set. It returns True if all the elements of the first set are present in the second set, and False otherwise.
- .issuperset() : The issuperset() method is used to check if a set is a superset of another set. It returns True if the first set contains all the elements of the second set, and False otherwise.
- .pop() : It is used to remove a value at the end of the set
- .remove() : The remove() method is used to remove a specific element from a set. It modifies the set in-place and raises a KeyError if the element is not found in the set.
- .symmetric_difference() : The symmetric_difference() method is used to find the symmetric difference between two sets. It returns a new set that contains elements that are present in either of the sets but not in both
- .union() : The union() method is used to combine two or more sets into a new set that contains all the unique elements from the individual sets.

DICTIONARY

- Dictionary is always enclosed with {}
- It is in the form of key: value pairs
- Keys are unique and duplicate values are allowed
- Multiple data types are allowed in dictionary
- Dictionary is Growable
- Slicing and Indexing is not allowed

DICTIONARY FUNCTIONS

- .clear() : The clear() method is used to remove all elements from a set. It modifies the dict in-place and leaves it empty.
- .copy(): It returns a new set with the same elements as the original set
- .fromkeys() : The fromkeys() method is a class method available for dictionaries in Python. It creates a new dictionary with specified keys and a default value for each key.
- .get() : It allows you to retrieve the value associated with a specified key in a dictionary.
- .items() : It allows you to retrieve a list of key-value pairs from the dictionary as tuples.
- .keys() : It allows you to retrieve a view object that contains all the keys in the dictionary.
- .pop() : The pop() method is primarily used to remove and return an element from a dictionary based on its key.
- .values() : It allows you to retrieve a view object that contains all the values in the dictionary
- .update() : It is used to merge the key-value pairs from one dictionary into another.

AUTHOR: V. N. V. PRAVEEN

vetsapraveen0098@gmail.com

www.linkedin.com/in/praveenvetsa

https://github.com/Praveenvetsa