1.**What is List? How will you reverse a list?**

* It allows to create variable-length and mutable sequences of objects. In a list, one can store objects of any type. one can also mix objects of different types within the same list
* In Python, a built-in function called reverse() is used to reverse the list.

**2. How will you remove last object from a list?**

**Suppose list1 is [2, 33, 222, 14, and 25],what**

**is list1 [-1]?**

* for remove last object from list pop() method can be used.

e.g : list[2,33,222,14,25]

list.pop()

print(list)

* if there is a one list =[2,55,6,33,56,6,7], then in this list ‘7 ’ is on the -1 position as if one wants to search elements from list.

**3. Differentiate between append () and extend ()**

**methods?**

* The append() method in the Python programming language adds an item to a list that already exists whereas extend() method adds the specific list elements to the end of the curent list.

**4. How will you compare two lists?**

1. sort() method

2. “==” operator

* sort() method sorts the two lists and the == operator compares the two lists item by item which means they have equal data items at equal positions. This checks if the list contains equal data item values but it does not take into account the order of elements in the list.

**5. What is tuple? Difference between list and tuple**

* Mutability:

List: Lists are mutable, which means you can change their elements after they have been defined. You can add, remove, or modify elements in a list.

Tuple: Tuples are immutable, which means once they are created, you cannot change the elements. If you try to modify a tuple, you'll get an error.

* Syntax:

List: Lists are created using square brackets [].

Tuple: Tuples are created using parentheses ().

* Use Cases:

List: Use lists when you have a collection of items that may need to be modified. For example, a list of tasks, a list of names, etc.

Tuple: Use tuples when you want to group together related pieces of information that shouldn't be changed. For example, coordinates (x, y), dates (year, month, day), etc.

* Performance:

Tuple: Tuples are generally more memory efficient and faster to access than lists, because they are immutable. This means that Python can make certain optimizations.

List: Lists, being mutable, may require more memory and operations for certain operations like appending or modifying elements.

* Methods:

List: Lists have several built-in methods for manipulation, such as append(), extend(), remove(), etc.

Tuple: Tuples have fewer methods because they are immutable. They support operations like indexing, slicing, and counting.

* Iteration:

List: You can iterate over the elements of a list using a loop.

Tuple: You can also iterate over the elements of a tuple using a loop.

**6. How will you create a dictionary using tuples in**

**python?**

* Here is the step-by-step approach using dict() constructor and list comprehension: Create a list comprehension that generates a tuple of the form (key, value) for each tuple in the input list. Pass the list comprehension as an argument to the dict() constructor to create a dictionary.

1. **How Do You Traverse Through A Dictionary Object**

**In Python?**

There are several ways to traverse through a dictionary object in Python. Here are a few examples:

* Using the keys() method:

The keys() method returns a list of the keys in a dictionary. You can then use this list to iterate over the dictionary.

dictionary = {'a': 1, 'b': 2, 'c': 3}

keys = dictionary.keys()

for key in keys:

print(key)

* Using the items() method:

The items() method returns a list of tuples, where each tuple contains a key-value pair from the dictionary. You can then use this list to iterate over the dictionary and print the key-value pairs.

dictionary = {'a': 1, 'b': 2, 'c': 3}

items = dictionary.items()

for key, value in items:

print(key, value)

* Using the for loop:

You can also use a for loop to iterate over a dictionary. The for loop will iterate over the keys in the dictionary, and you can then use the get() method to get the value for each key.

dictionary = {'a': 1, 'b': 2, 'c': 3}

for key in dictionary:

value = dictionary.get(key)

print(key, value)

* Using the in operator:

You can also use the in operator to iterate over a dictionary. The in operator will return True if the key exists in the dictionary, and False if it does not.

dictionary = {'a': 1, 'b': 2, 'c': 3}

if 'a' in dictionary:

print('The key "a" exists in the dictionary.')

else:

print('The key "a" does not exist in the dictionary.')

1. **How Do You Check The Presence Of A Key In A**

**Dictionary?**

* using get() :

Using the Inbuilt method get() method returns a list of available keys in the dictionary. With the Inbuilt method keys(), use the if statement to check if the key is present in the dictionary or not. If the key is present it will print “Present” Otherwise it will print “Not Present”.

1. **Why Do You Use the Zip () Method in Python?**

* The zip() method in Python is used to combine two or more iterables into a single iterable, where elements from corresponding positions are paired together. This can be useful for a variety of tasks, such as merging two lists, creating a dictionary, or iterating over multiple iterables at once.

E.g : Merge two lists:

list1 = [1, 2, 3]

list2 = [4, 5, 6]

merged\_list = list(zip(list1, list2))

print(merged\_list)

Output : [(1, 4), (2, 5), (3, 6)]

E.g : Create a dictionary:

list1 = ["a", "b", "c"]

list2 = [1, 2, 3]

dictionary = dict(zip(list1, list2))

print(dictionary)

Output : {'a': 1, 'b': 2, 'c': 3}

1. **How Many Basic Types Of Functions Are Available In Python?**

* two types

Mainly, there are two types of functions: User-defined functions – These functions are defined by the user to perform a specific task. Built-in functions – These functions are pre-defined functions in Python.

1. **How can you pick a random item from a list or tuple?**

* In Python, you can use the choice() function from the random module to randomly select an item from a list or tuple. The choice() function takes one argument, which is a non-empty sequence like a list, tuple, string, or any iterable like range. It returns a single random element from the sequence.

To use the choice() function, you can:

Import the random module.

Create a tuple and add some dummy data to it.

Generate a random item from the tuple using random.

Print the generated random tuple item.

1. How can you pick a random item from a range?

* You can use the Python random module to pick a random item from a range. The random module defines functions for generating random numbers.

e.g : Using random.randrange() function.

Using random.randint() function.

Using random.random() function.

Using random.sample()

1. **How can you get a random number in python?**

* There are several ways to get a random number in Python. One way is to use the randint() function. This function takes two arguments: the minimum and maximum values of the random number. For example, the following code will generate a random number between 0 and 10:

E.g : import random

number = random.randint(0, 10)

print(number)

1. **How will you set the starting value in generating random numbers?**

* To set the starting value (seed) for generating random numbers, you can use the srand(x) function. The srand(x) function sets the seed of the random number generator algorithm used by the function rand( ). The default seed value is 1

 How will you randomizes the items of a list in place?