What is List? How will you reverse a list?

* A list is a data structure in Python that is a mutable, or changeable, ordered sequence of elements. Each element or value that is inside of a list is called an item. Just as strings are defined as characters between quotes, lists are defined by having values between square brackets [ ]
* reverse() Method. Every list in Python has a built-in reverse() method you can call to reverse the contents of the list object in-place. Reversing the list in-place means won't create a new list and copy the existing elements to it in reverse order. Instead, it directly modifies the original list object.

How will you remove last object from a list?

Suppose list1 is [2, 33, 222, 14, and 25], what is list1 [-1]?

* One way is to use the pop() method. This method removes the last element of a list by default, or you can specify the index of the element you want to remove.
* For example: my\_list =[2, 33, 222, 14, and 25]

-1 is a and 25

Differentiate between append () and extend () methods?

* In Python, the main difference between the append() and extend() methods is that append() adds a single item to the end of a list, while extend() adds multiple items from an iterable to the end of a list:
* **append()**

Adds a single item to the end of a list. For example, products.append("Apricots") adds the item "Apricots" to the end of the list products.

* **extend()**

Adds multiple items from an iterable to the end of a list. For example, products.extend(["Apricots", "Mangoes"]) adds the items "Apricots" and "Mangoes" to the end of the list products.

What is a tuple? Difference between list and tuple.

* **Tuple**: A tuple is an immutable ordered collection of elements. Once created, its values cannot be changed. **Difference**:
* **List**: Mutable, allows changes after creation, defined with [].
* **Tuple**: Immutable, does not allow changes after creation, defined with ().

Why use the zip() method?

* The zip() method is used to combine two or more iterables (e.g., lists or tuples) element-wise. It creates a zip object that pairs elements based on their position.

**1. Types of functions in Python**

* **Built-in Functions**: Predefined in Python, e.g., print(), len().
* **User-defined Functions**: Created by users to perform custom operations.
* **Lambda Functions**: Anonymous, single-expression functions defined using lambda.
* **Recursive Functions**: Call themselves to solve sub-problems, such as calculating factorials.