You’ve explored **lists**, **slicing**, **looping**, **membership**, **methods**, and **list comprehensions** in Python. Here's a **full explanation**, part-by-part, of everything you typed:

**🧠 1. List Basics**

python

CopyEdit

language\_varities = ["C" , "C++" , "JAVA" , "Python"]

Creates a list of strings.  
Access elements using **indexes**:

* language\_varities[0] → 'C'
* language\_varities[-1] → 'Python' (last element)

**✂️ 2. List Slicing**

python

CopyEdit

language\_varities[1:3] → ['C++', 'JAVA']

language\_varities[:2] → ['C', 'C++']

language\_varities[2:] → ['JAVA', 'Python']

* [start:stop] → Includes start, excludes stop.

**🔁 3. Changing List Elements**

python

CopyEdit

language\_varities[3] = "C#"

Changes 'Python' to 'C#'.

python

CopyEdit

language\_varities[1:2] = ["net core"]

Replaces only 'C++' with 'net core'.

python

CopyEdit

language\_varities[1:3] = ["Html", "Javascript"]

Replaces 'net core' and 'JAVA' with 2 new elements.

python

CopyEdit

language\_varities[1:1] = ["test", "test"]

Inserts **without replacing** (between index 1 and 1).

python

CopyEdit

language\_varities[1:3] = []

Removes 'test' and 'test'.

**🔁 4. Looping Through a List**

python

CopyEdit

for language in language\_varities:

print(language)

Prints each element.

python

CopyEdit

print(language, end="-")

Prints all items in one line with - separator.

**🔍 5. Membership Testing (in)**

python

CopyEdit

if "CSS" in language\_varities:

Checks whether "CSS" exists in the list. If yes, runs the print statement.

🛑 You got an error earlier because of a **typo**:

python

CopyEdit

if "Html" in langauge\_varities: # wrong spelling

**🧰 6. List Methods**

python

CopyEdit

language\_varities.append("JavaScript")

Adds item to the **end**.

python

CopyEdit

language\_varities.pop()

Removes and **returns the last item**.

python

CopyEdit

language\_varities.remove("C")

Removes the item "C".

python

CopyEdit

language\_varities.insert(1, "Numpy")

Inserts "Numpy" at index 1.

python

CopyEdit

language\_varities.copy()

Creates a **shallow copy** of the list.

**🧮 7. range() and Comprehensions**

python

CopyEdit

range(10) → range object from 0 to 9

**🔢 List Comprehension**

python

CopyEdit

squared\_num = [x\*\*2 for x in range(10)]

* Squares of 0–9 → [0, 1, 4, ..., 81]

python

CopyEdit

cube\_num = [y\*\*3 for y in range(5)]

* Cubes of 0–4 → [0, 1, 8, 27, 64]

**✅ Summary**

| **Concept** | **Example** | **Output** |
| --- | --- | --- |
| List access | x[0] | First item |
| Slicing | x[1:3] | Sub-list |
| Modify list | x[1] = val | Change |
| Insert | x[1:1] = [val] | Insert |
| Remove | pop(), remove() | Delete items |
| Add | append(), insert() | Add items |
| Copy | copy() | New list |
| Loop | for x in list: | Iteration |
| Comprehension | [x\*\*2 for x in range(5)] | [0, 1, 4, 9, 16] |