Session 5: Loops in Python

🎯 Topics Covered

- The for loop

- The while loop

- Why & When to use them

- Difference between for and while

- List methods with loops

- Practice problems

🔑 1. Introduction to Loops

Loops are used when we want to repeat a block of code multiple times.
Instead of writing the same code again and again, we use loops to save time and make programs shorter and cleaner.

🔁 2. The for Loop

Definition: A for loop is used to iterate over a sequence (like list, tuple, string, or range).

Why use a for loop?
- When you know how many times you want to run the loop.
- Best for fixed repetitions.

Example:

```
for i in range(3):
    print(f"For loop count: {i}")
```

Output:
For loop count: 0
For loop count: 1
For loop count: 2

🔂 3. The while Loop

Definition: A while loop keeps running as long as the condition is True.

Why use a while loop?
- When you don't know how many times you need to repeat.
- Best for condition-based repetition.

Example:

```
count = 0
while count < 3:
    print(f"While loop count: {count}")
    count += 1
```

Output:
While loop count: 0
While loop count: 1
While loop count: 2

⚖️ 4. Difference Between for and while

Feature	for loop 🌀	while loop 🔂
Use Case	When we know the number of iterations	When the number of iterations is unknown
Control	Controlled by sequence (range, list)	Controlled by a condition (True/False)
Risk	Usually safe, ends automatically	Risk of infinite loop if condition is wrong
Example	Printing numbers 1–10	Keep asking for password until correct

📋 5. Common List Methods with Loops

Method	Description
append()	Adds an element at the end of the list
clear()	Removes all the elements from the list
copy()	Returns a copy of the list
extend()	Adds elements of a list (or any iterable) to the end
insert()	Adds an element at the specified position

pop()	Removes the element at the specified position
remove()	Removes the item with the specified value

Example with List Methods and Loops

```
fruits = ["apple", "banana", "cherry"]

# Using append in for loop
for i in range(3):
    fruits.append(f"fruit_{i}")

print(fruits)

# Using pop in while loop
while fruits:
    print("Popped:", fruits.pop())
```

🎯 6. Practice Problems

Problem 1: Sum of Numbers

```
total = 0
for i in range(1, 11):
    total += i
print("Sum =", total)
```

Problem 2: Multiplication Table

```
num = 5
i = 1
while i <= 10:
    print(f"{num} x {i} = {num * i}")
    i += 1
```

Problem 3: Reverse a List using Loops

```
nums = [1, 2, 3, 4, 5]
reversed_list = []
for n in nums:
    reversed_list.insert(0, n)

print("Reversed:", reversed_list)
```

📝 7. Assignments for You

1. Write a for loop that prints only even numbers from 1 to 20.

2. Write a while loop that keeps asking the user for input until they type "stop".

3. Create a list of 5 numbers and use a loop with append() to store their squares in a new list.

4. Use pop() in a loop to empty a list and print each element.

📚 Extra Code Examples for List Methods

Example: copy()

```
fruits = ["apple", "banana", "cherry"]
copied_fruits = fruits.copy()
print("Original:", fruits)
print("Copied:", copied_fruits)
```

Example: extend()

```
fruits = ["apple", "banana"]
extra = ["cherry", "date"]
fruits.extend(extra)
print("Extended list:", fruits)
```

Example: insert()

```
fruits = ["apple", "banana", "cherry"]
fruits.insert(1, "orange")
print("After insert:", fruits)
```

Example: remove()

```
fruits = ["apple", "banana", "cherry"]
fruits.remove("banana")
print("After remove:", fruits)
```