Introduction to Loops in Python

- Loops are fundamental constructs in programming that allow repeated execution of code blocks.
- They help in automating repetitive tasks efficiently.
- Python primarily uses for loops and while loops to perform iterations.
- Understanding loops is crucial for writing efficient and concise code.

Range Function

- The range function generates a sequence of numbers.
- Syntax: range(start, stop, step)
- Example: range(0, 5) produces [0, 1, 2, 3, 4]
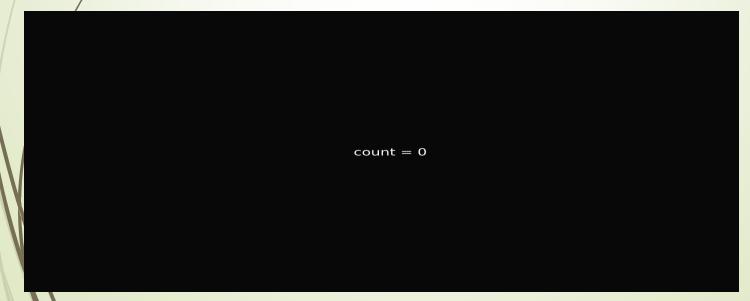- Used in for loops to iterate over a sequence of numbers.

For Loop

- A for loop is used to iterate over a sequence (list, tuple, string) or other iterable objects.
- Syntax: for item in iterable:
- Example: for i in range(5): print(i)
- Executes the block of code repeatedly for each item in the sequence.

```
numbers = [1, 2, 3, 4, 5]
```

While Loop

- A while loop repeatedly executes a block of code as long as a condition is true.
- Syntax: while condition:
- Example: while i < 5: i += 1
- Executes the block of code until the condition is false.

```
count = 0
```

Nested Loops

- A nested loop is a loop inside another loop. The inner loop runs completely whenever the outer loop runs once.
- Syntax:

```
for i in range(outer_loop_count):
    for j in range(inner_loop_count):
```

- Example:

```
for i in range(3):
    for j in range(2):
        print(f"i = {i}, j = {j}")
```

Visual Demonstration of Nested loop

```python
for i in range(3):
    for j in range(2):
        print(f"i = {i}, j = {j}")
```

Quiz on Loops

1. What is the difference between for loop and while loop?
2. How does the range function work in for loops?
3. Provide an example of a while loop in Python.
4. What are the use cases of for loop and while loop?