

1 Introduction to Loops

In Python, **loops** are used to **repeat a block of code multiple times**.

They help perform tasks like printing a message several times, iterating over lists, or generating patterns.

Example:

```
print("Hello Saumya!")  
print("Hello Saumya!")  
print("Hello Saumya!")  
print("Hello Saumya!")
```

Instead of writing this three times, we can use a **loop**.

2 while Loop

A **while** loop runs **as long as a condition is True**.

Syntax:

```
while condition:  
    # code block
```

Example:

```
i = 1  
while i <= 5:  
    print("Hello, Saumya Singh!")  
    i += 1
```

Output:

```
Hello, Saumya Singh!  
Hello, Saumya Singh!  
Hello, Saumya Singh!  
Hello, Saumya Singh!  
Hello, Saumya Singh!
```

Practice Questions

1. Write a Python program to print numbers from 1 to 10 using a **while** loop.
2. Write a program to print numbers from 10 down to 1 using a **while** loop.
(Hint: start from 10 and decrease the counter each time.)

Example Output: 10 9 8 ... 1

3. Write a program to print all even numbers between 1 and 50 using a while loop.
(Hint: Use the modulus operator % to check for even numbers.)

Example Output: 2 4 6 8 ... 50

4. Write a program that prints the sum of first n natural numbers.
For example, if $n = 5$, then output should be $1 + 2 + 3 + 4 + 5 = 15$.
(Hint: Keep a running total inside the loop.)
5. Write a program to print this pattern using a while loop:

```
*  
  
* *  
  
* * *  
  
* * * *
```

(Hint: use one while loop and string multiplication like '*' * count)

6. Saumya wants to print her name 5 times, but each time with a number in front of it. Write a program using a while loop that prints:

```
1. Saumya Singh  
2. Saumya Singh  
3. Saumya Singh  
4. Saumya Singh  
5. Saumya Singh
```

7. Write a program to print the multiplication table of any number using a while loop.

(Hint: Start $i = 1$ and run the loop until $i \leq 10$.)

Example Output:

Enter a number: 3

3 x 1 = 3

3 x 2 = 6

...

3 x 10 = 30

3 for Loop

A **for** loop is used to **iterate (go through)** sequences like lists, tuples, or strings.

Syntax:

```
for element in sequence:  
    # code block
```

Example:

```
foods = ["Pizza", "Samosa", "GulabJamun"]  
for item in foods:  
    print("Saumya likes", item)
```

Output:

```
Saumya likes Pizza  
Saumya likes Samosa  
Saumya likes GulabJamun
```

for Loop with range()

The **range()** function generates a sequence of numbers.
It is often used with loops.

Syntax:

`range(start, stop, step)`

- `start` → beginning number (default = 0)
- `stop` → end limit (excluded)
- `step` → increment value (default = 1)

Example:

```
for i in range(1, 6):  
    print("Iteration:", i)
```

Output:

```
Iteration: 1  
Iteration: 2  
Iteration: 3  
Iteration: 4  
Iteration: 5
```

Practice Questions

8. Write a program using `for` and `range()` to print all even numbers between 1 and 20.
9. Write a program to print numbers from 1 to 50, but print "Saumya Singh" instead of numbers that are multiples of 5.
Example Output: 1 2 3 4 Saumya Singh 6 7 8 9 Saumya Singh ...
10. Write a program to print the square of each number from 1 to 10 using a for loop.
Example Output: 1 4 9 16 25 36 49 64 81 100
11. Write a program that prints the multiplication table of any number entered by the user using a for loop.

Example Output:

```
Enter number: 6
6 x 1 = 6
6 x 2 = 12
...
6 x 10 = 60
```

12. Write a program that prints all numbers from 100 to 1 using for and range().

Hint:

Use a negative step: range(100, 0, -1)

13. Saumya wants to print her username five times in uppercase letters. Write a program to print:

```
SAUMYA1SINGH
SAUMYA1SINGH
SAUMYA1SINGH
SAUMYA1SINGH
SAUMYA1SINGH
```

Hint:

Use a simple for loop and .upper() method.

14. You are given a list of Saumya's favorite foods. Write a Python program to print each food item using a for loop.

15. Saumya has created a tuple of countries she has already traveled to. Write a Python program to print each country using a for loop.

```
countries_traveled = ("Malaysia", "Vietnam", "Switzerland", "Italy", "Bhutan")
```

4 break, continue, and pass

break Statement

The **break** statement **stops** the loop immediately when it is encountered.

Example:

```
for num in range(1, 10):  
    if num == 5:  
        break  
    print(num)
```

Output:

```
1  
2  
3  
4
```

(Loop stops when `num == 5`)

continue Statement

The `continue` statement **skips** the current iteration and moves to the next one.

Example:

```
for num in range(1, 6):  
    if num == 3:  
        continue  
    print(num)
```

Output:

```
1  
2  
4  
5
```

(3 is skipped)

pass Statement

The `pass` statement **does nothing** — it's used as a placeholder when you want to keep a block empty.

Example:

```
for i in range(5):  
    pass # future code will go here
```

Practice Questions

Write a program that prints numbers 1 to 10, but skips the number 7 using the `continue` statement.

5 Nested Loops

A **loop inside another loop** is called a nested loop.

Example:

```
for i in range(1, 4):  
    for j in range(1, 4):  
        print(i, j)
```

Output:

```
1 1  
1 2  
1 3  
2 1  
2 2  
2 3  
3 1  
3 2  
3 3
```

Practice Question 4

Write a program using nested loops to print this pattern:

```
*  
* *  
* * *
```

Mini Project – Countdown Timer (with 1-second gap)

Goal:

Print a countdown before something “exciting” happens (like “Launching...” or “Happy New Year!”).

Concepts Used: for loop, range(), and the time module.

Mini Project – Multiplication Table

Goal: Print the multiplication table of a number using a loop.

Sample Run:

```
Enter a number: 5
5 x 1 = 5
5 x 2 = 10
...
5 x 10 = 50
```

Mini Project – Guess the Number Game

Concepts Used: while loop and user input.

Sample Run:

```
Guess a number between 1 and 10: 4
Wrong guess! Try again.
Guess again: 7
Congratulations, Saumya! You guessed it right 🎉
```

Summary

Concept	Description	Example
---------	-------------	---------

while loop	Repeats while condition is true	<code>while i <= 5:</code>
for loop	Iterates over sequence	<code>for item in list:</code>
range()	Generates number sequence	<code>range(1, 6)</code>
break	Exits the loop	<code>if x == 5: break</code>
continue	Skips one iteration	<code>if x == 3: continue</code>
pass	Placeholder (does nothing)	<code>pass</code>

Practice Set (Quick Recap)

1. Print numbers from 1 to 100 using a `for` loop.
2. Print numbers from 100 to 1 using a `while` loop.
3. Print all numbers between 1 and 50 except multiples of 5.
4. Create a program that asks the user for 5 favorite foods and prints them one by one.
5. Print the sum of first 10 natural numbers using a `while` loop.