

Looping Concepts - The while Loop in Python

Introduction:

Loops are essential in programming **for executing a block of code repeatedly**. The while loop in Python **allows** you to *iterate as long as a certain condition holds true*.

The Basic Syntax of the while Loop:

The syntax of the while loop is as follows:

```
while condition:
```

```
    # Code to be executed while the condition is True
```

The loop continues executing the indented block of code as long as the condition remains true.

Example: Simple Countdown with while Loop:

Let's create a simple program that counts down from 5 to 1 using a while loop.

```
count = 5
while count > 0:
    print(count)
    count -= 1
```

Observe how the loop continues until the condition `count > 0` becomes false.

User-Driven Loop:

You can use a while loop to repeatedly prompt the user until a specific condition is met.

```
user_input = ""
while user_input.lower() != "exit":
    user_input = input("Enter a command (type 'exit' to end): ")
    print(f"You entered: {user_input}")
```

This loop keeps asking the user for input until the user types "exit."

Conclusion:

In this lab, we explored the while loop in Python, a powerful tool for repetitive execution. Understanding and mastering looping concepts is crucial for writing efficient and dynamic programs. Experiment with different conditions and scenarios to solidify your understanding.

Assignment:

Create a Python program that uses a while loop to do the following:

- Ask the user to enter a number.
- Keep asking for input **until the user enters a negative number**.

