Class 10: while Loops in Python

Instructor: Md Rasel Sarker

What is a while loop?

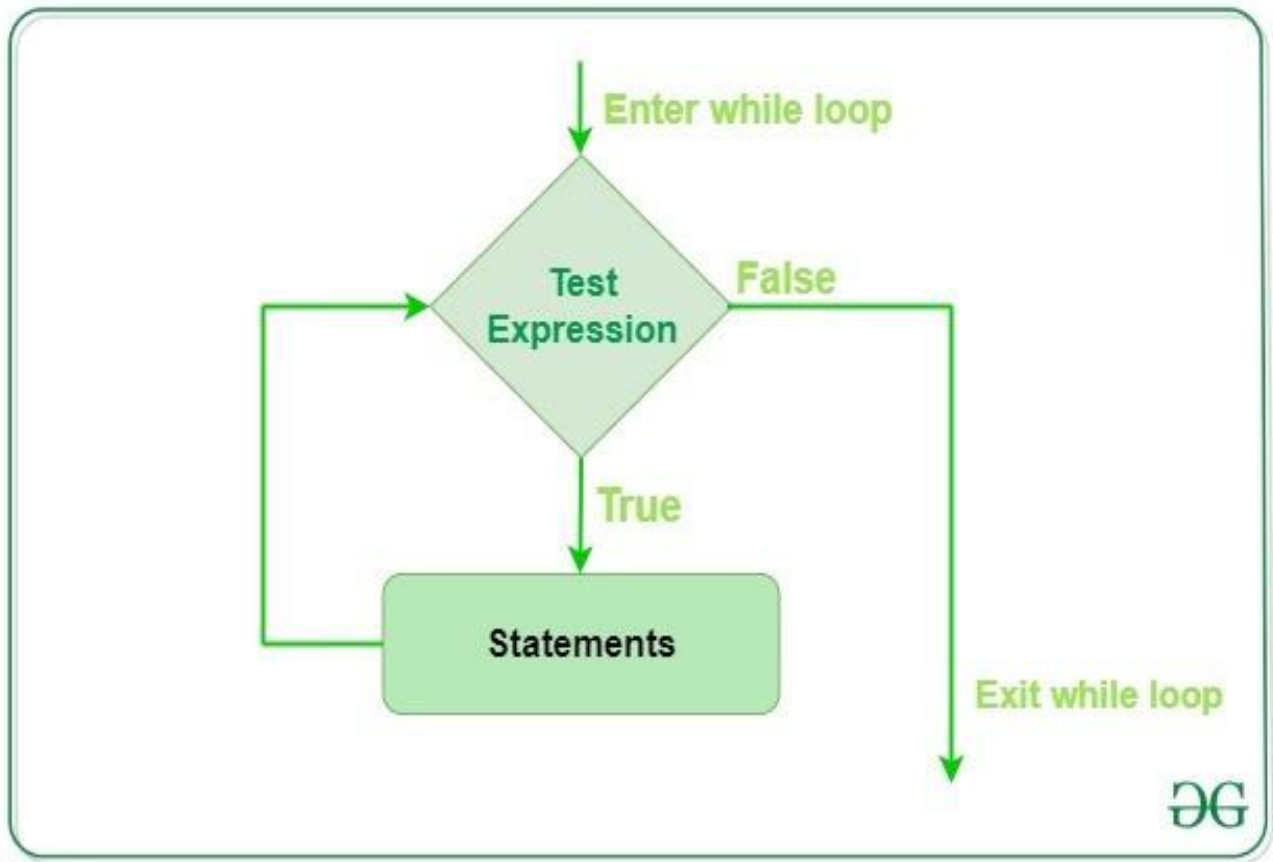
- Loops are used to repeat instructions.
- A while loop is used to repeatedly execute a block of code as long as a certain condition is true.
- Use a while loop when you don't know beforehand how many times you need to execute the loop.

While Loops:

While conditions:

    #some work

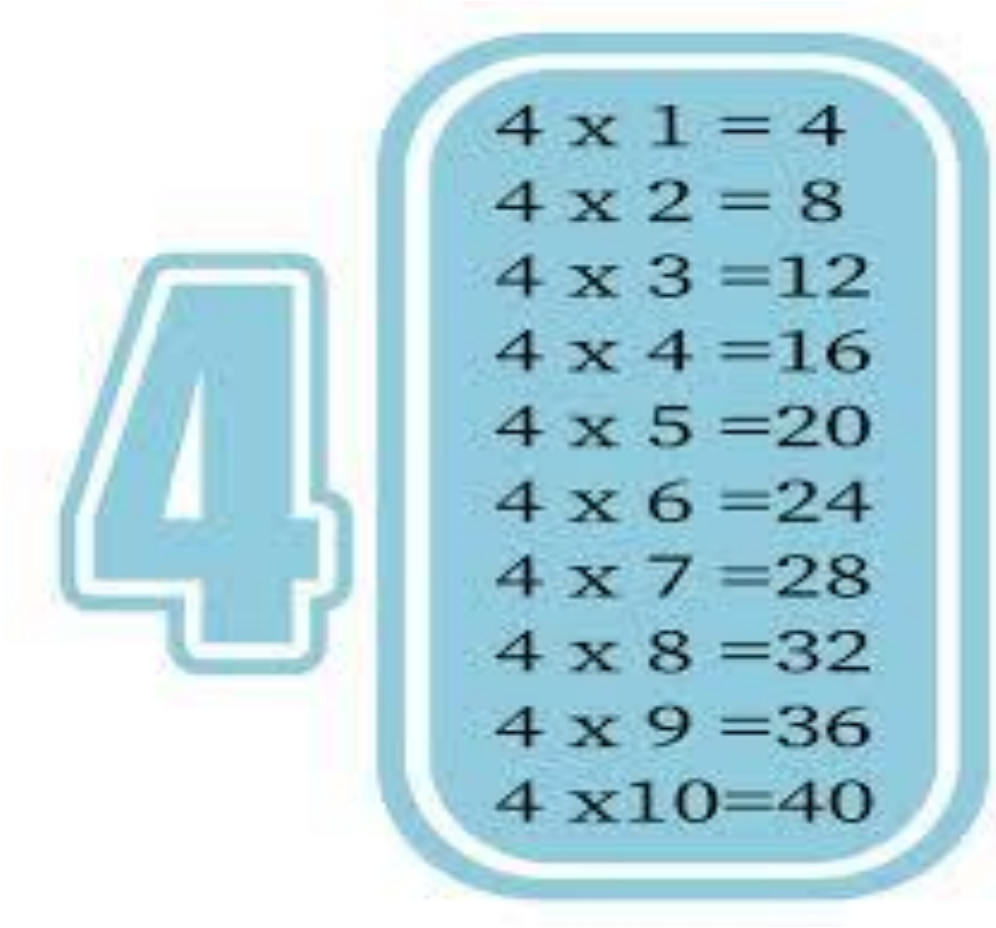
```
i = 1
while i < 5:
    print(i)
    i += 1
```


Enter while loop
Test Expression
False
True
Statements
Exit while loop

Let’s Practice

- Print numbers from 1 to 100.
- Print numbers from 100 to 1.
- Print the multiplication table of a number n.
- Print the elements of the following list using a loop:
  [1, 4, 9, 16, 25, 36, 49, 64, 81, 100]

Multiplication Table



Thanks for Watching

Class 11: 8+ problem solving
using while Loops

Instructor: Md Rasel Sarker

Email: rasel.sarker6933@gmail.com

Let’s Practice

1. WAP to find the sum of first n numbers (using while)
2. WAP to print even numbers from 1 to n (using while)
3. WAP to count digits in a number (using while)
4. WAP to reverse a number (using while)
5. WAP to find the sum of digits of a number (using while)
6. WAP to check if a number is palindrome or not (using while)
7. Write a program to calculate the factorial of n using a while loop.
8. WAP to check if a number is prime (bonus: not in original list)

Class 12: Break & Continue in Python

Instructor: Md Rasel Sarker

Email: rasel.sarker6933@gmail.com

Break & Continue in Python

- **break** – Immediately exits the loop when it is encountered. The loop stops running, and control moves to the first line after the loop.

- **continue** – Skips the rest of the code in the current loop iteration and jumps to the next iteration of the loop.

Let’s Practice

1: Print numbers from 1 to 20, but stop the loop when it reaches 10.

2: Continuously take names from the user and stop the loop when "stop" is entered.

3: From 1 to 50, find and print the first number divisible by 7, then break the loop.

4: Print numbers from 1 to 10, but skip 5.

5: From a list, print only even numbers (skip the odd ones).

6: In a sentence, skip the vowels and print only consonants.

Class 13: For Loops in Python

Instructor: Md Rasel Sarker

Email: rasel.sarker6933@gmail.com

For Loops in Python

- A for loop is used to iterate over a sequence (such as a list, tuple, or string).
- Use a for loop when you know exactly how many times you want to execute the loop.
- Loops are used for **sequential traversal**, meaning you can iterate through elements in **lists, strings, tuples**, etc.

```
list = [1, 2, 3]
for el in list:
    # some work
    print(el)
```

Various functions & keywords can control a for loop's behavior.

Function / Keyword	Purpose
break	Exits the loop immediately.
continue	Skips the current iteration.
pass	Does nothing (used as a placeholder).
range()	Generates numbers for iteration.
else	Runs if the loop completes without a break.
reversed()	Iterates over a sequence in reverse order.
sorted()	Iterates over a sorted version of the sequence.
zip()	Iterates over multiple sequences in parallel.
enumerate()	Provides index and item during iteration.

For Loop with else

- The else block executes if the loop finishes normally.
- If the loop is broken with break, the else block is skipped.

```
list = [1, 2, 3]
for el in list:
    # some work
    print(el)

else:
    # work when loop ends normally (not by break)
    print("END")
```

Let’s Practice – Using for Loop

Task 1: Print all elements of the following list using a loop:

numbers = [1, 4, 9, 16, 25, 36, 49, 64, 81, 100]

Task 2: Search for a number x in this tuple using a loop:

values = (1, 4, 9, 16, 25, 36, 49, 64, 81, 100)

Task 3: Search for a prime number in a range; if none is found, print "No prime found" from the else block.

range()

- The range function returns a sequence of numbers. It starts from 0 by default, increments by 1, and stops before the specified number.
- Commonly used to run iterations in a for loop.

range(start, stop, step)

```
num = [1, 10, 19, 22, 25, 36, 49, 64, 81, 100]

for el in range(8):
    print(num[el])

for el in range(1, 8):
    print(num[el])

for el in range(1, 8, 2):
    print(num[el])
```

Let’s Practice – using for & Range

1. Print numbers from 1 to 100.
2. Print numbers from 100 to 1.
3. Print the multiplication table of 5 using range().

Pass statements

- Pass is a null statement that does nothing. It is use as a placeholder for future code.
- Used to leave a loop or code block empty without causing an error.
- Acts only as a placeholder, does nothing.

Example

```
for el in range(10):
    pass
```

Let’s Practice

1. WAP to find the factorial of first n numbers (using for):
2. WAP to check if a number is prime (using for)
3. WAP to print multiplication table of a number (using for)
4. WAP to find the largest number from n inputs (using for)

5: Write a loop that skips the word "banana" using pass.

enumerate()

- when you need access to both index and value in a loop.

Let’s Practice

1: Print each item of a list along with its index.

2: Show the index of each word in a sentence.

3: Find the index of the first item in a list that is "apple".

zip()

- Helps iterate through two or more lists in pairs.
- Returns data as tuples from lists of equal length.

Let’s Practice

1: Print the items of two lists together (e.g., names and ages).

2: Create a new list by adding elements of two number lists.

3: Pair two lists (countries and capitals) and print the pairs.

Reversed()

- Used to iterate over a sequence in reverse order.
- Returns a reverse iterator without modifying the original list.

Let’s Practice

1: Print all items of a list in reverse order.

2: Print a string in reverse.

3: Count down from 10 to 1 using reversed().

Sorted()

- Sorts the elements of an iterable in order.
- By default, sorts in ascending order, but can be changed to descending.

Let’s Practice

1: Print a list of numbers in ascending order.

2: Sort a list of words alphabetically.

3: Sort a list of numbers in descending order using reverse=True.

Thanks for Watching