



9

Introduction to Loops

by Gladden Rumao

C02: Fundamentals of Programming

Join the lecture online on your dashboard

Start Lecture Recording

Recap of Previous Lecture!

Quick Recap :

- Defining Function
- Calling Functions
- Types of Arguments



Scope of Variables

Scope : Hostel analogy

Local Scope: Items inside your own personal room (e.g., clothes, books, suitcase etc).

Global Scope: Items in the common living room accessible to everyone in the hostel (TV, Sofa, Board Games etc).

Scope refers to the region of the program where a variable is recognized. If a variable is out of scope, it cannot be accessed or used.



Local scope :

Variables defined inside a function or a block are said to have a local scope.

These variables can only be accessed within the function or block where they are defined.



Global scope

Variables defined at the top level of a script or module, outside any function or block, are said to have a global scope.

These variables can be accessed anywhere in the module.



Global Keyword

Resolving scope: Example

*Local scope/variable will
be given priority over
global variable/scope.*

```
# Global variable declaration
x = "I am a global variable"

def my_function():
    # Local variable declaration with the same name as the global variable
    x = "I am a local variable"
    print(x) # Output: I am a local variable

# Calling the function to demonstrate local variable usage
my_function()

# Accessing the global variable outside the function
print(x) # Output: I am a global variable
```

Resolving scope: Error

Resolving scope: Error

```
num = 10 # Global variable

def check_scope():
    num2 = 10 + num # Gives error as num used before declaration
    num = 5 # Local variable num
    print(num2)

check_scope()
```

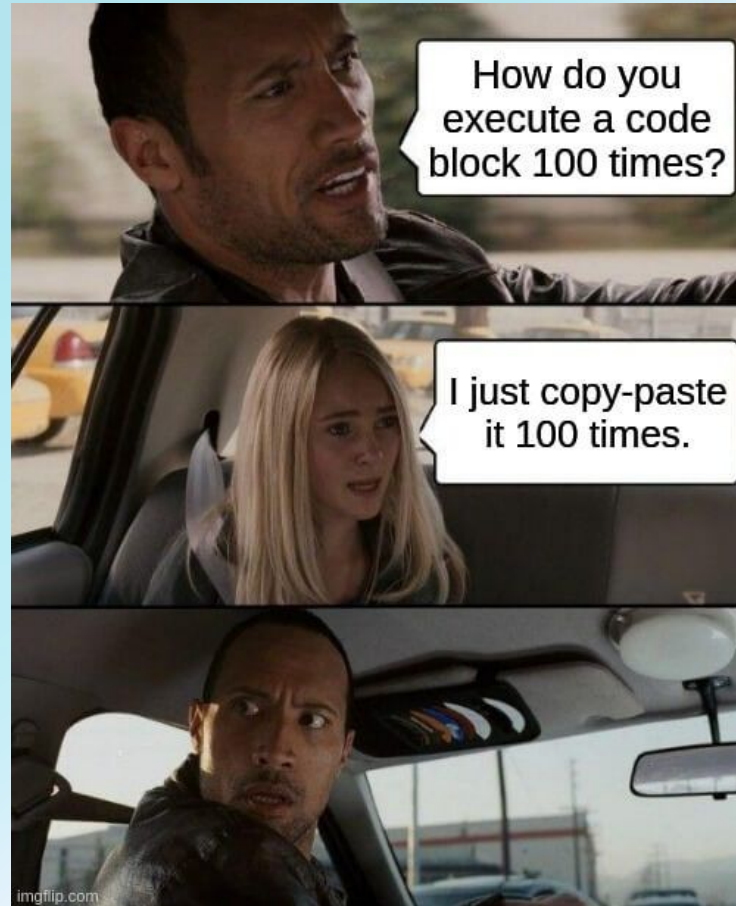
Resolving Scope: Fix for the problem

Ideally you should avoid variable with same names in global and local scope.

```
num = 10 # Global variable

def check_scope():
    global num # Declare num as global
    num2 = 10 + num # Accessing global variable num
    num = 5 # Modifying global variable num
    print(num2) # Output: 20

check_scope()
print(num) # Output: 5
```



Loops

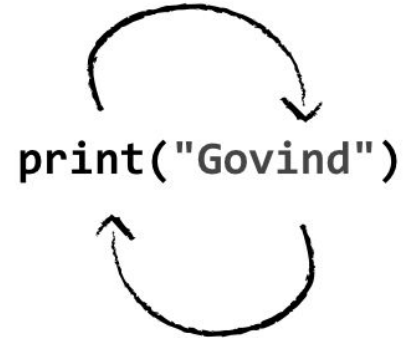
What are Loops ?

Consider I want to print a name 10 times.

```
print("Govind")  
print("Govind")  
print("Govind")  
print("Govind")  
print("Govind")  
print("Govind")  
print("Govind")  
print("Govind")  
print("Govind")  
print("Govind")
```

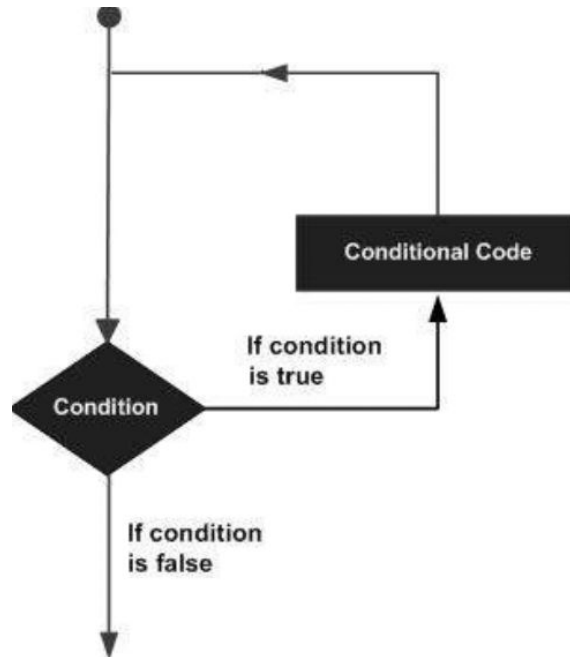


OR



What are Loops ?

A loop is used to repeat an instruction multiple times until a condition goes wrong.



Types of loop in Python

There are two primitive loop types in Python



for
Loop



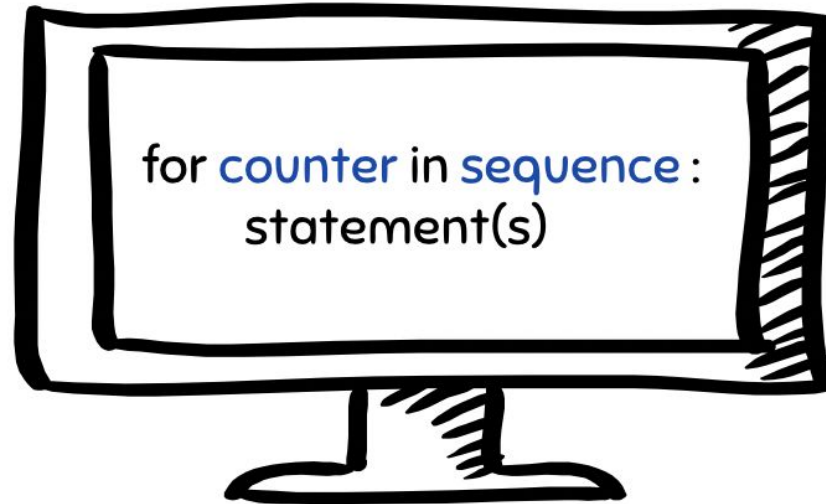
while
Loop

For Loop

For Loops

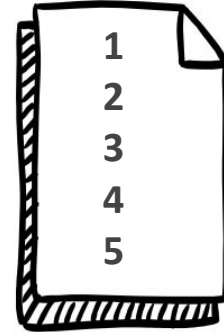
A **for** loop is used for iterating over a sequence.

Syntax :-



Range Function :

```
# for loop  
for i in range(1,6):  
    print(i)
```



Range Function :

Variations:

`range(stop)`

`range(start, stop)`

`range(start, stop, step)`



using a unique
variable that makes
sense in a for loop



i, j, k

ProgrammerHumor.it

Multiplication Table

Print Numbers – 1 to N

Sum of Numbers – 1 to N

Factorial of Number

While Loop

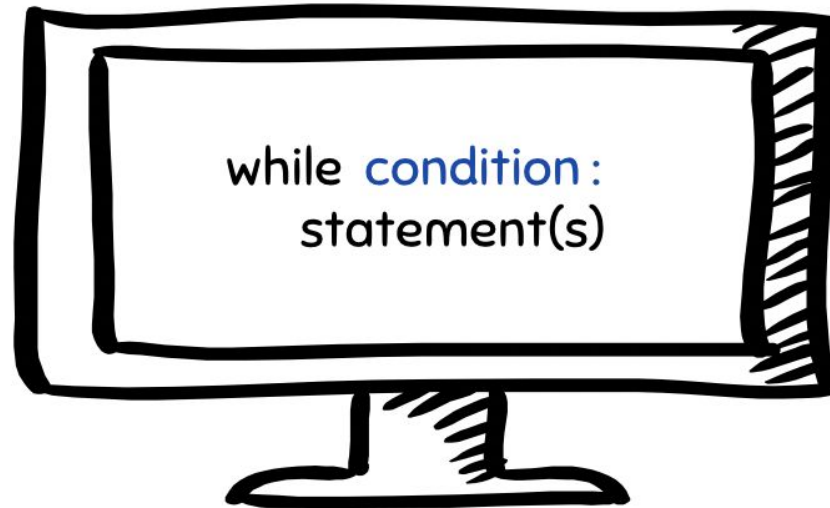
While Loop :

```
while(alive)
{
    eat();
    sleep();
    code();
}
```

While Loop :

A **while** loop is used to execute a set of statements as long as a condition is true.

Syntax :-



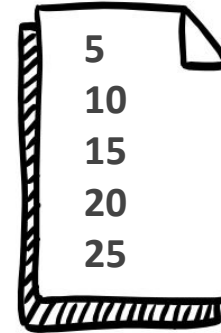
Infinite Loop :

```
while(true){  
    I love you;  
}
```

For the non-geeks: this means **I love you**
infinitely forever or until the system crashes

Examples

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



5
10
15
20
25



Thank You!