# Functions

- Block of lines of codes that can be repeated more than once in the code

- Save lines of codes

- Avoid repetitive task

- speeds up the process

- Modular Approach : CODE Blocks

- Update the codes will make it easier to be completed

- Whatever is written inside the function is JUST saved at some MEMORY location

- The code inside the function RUNS, only when you CALL the Function

**#### Return**

- return --> Helps to take the value outside the function

- A function can have multiple Returns

- Whenever a return is called inside the function, then function is STOPPED and value is taken out

- We can have single as well as multiple return values

**### Syntax**

\*\*Funtion Declaration\*\*

def functionName(a1, a2) : --- Formal Arguments

statements

statements

statements

statements

value = \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

return value #### To take the value out of the funtion

\*\*Calling the Funtion\*\*

functionName(a1, a2) --- Actual Arguments

**Global and Local Variable**

Global variable ::

Any variable that is written inside the Python code.

It can be used anywhere in the program. LIfecycle of variable is limited to Entire Program

Local variable ::

Any variable that is written inside the Python Function

It can ONLY be used in the FUNCTION. LIfecycle of variable is limited to Function

### Arguments

##### Syntax

\*\*Funtion Declaration\*\*

def functionName(a1, a2) : --- Formal Arguments

statements

statements

statements

statements

value = \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

return value #### To take the value out of the funtion

\*\*Calling the Funtion\*\*

functionName(a1, a2) --- Actual Arguments