

If-else Statement

If
Marks > 70:
Get Ice Cream



else
Give Practice Test



Pseudo-Code: if-else

```
If(condition):
```

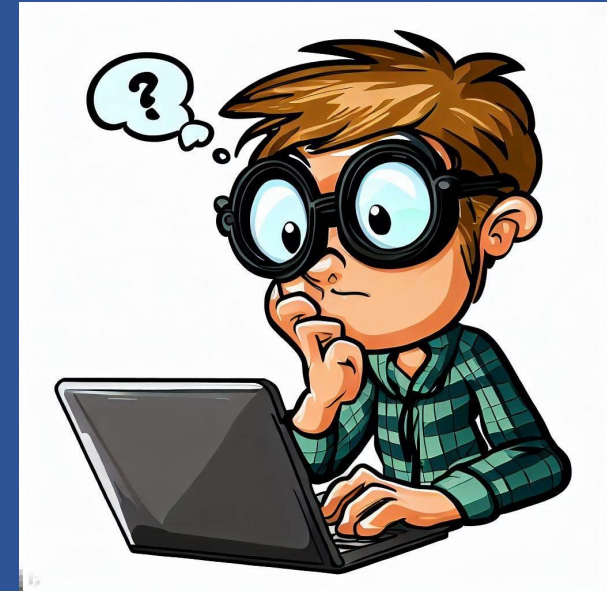
```
    Statements to be executed....
```

```
else:
```

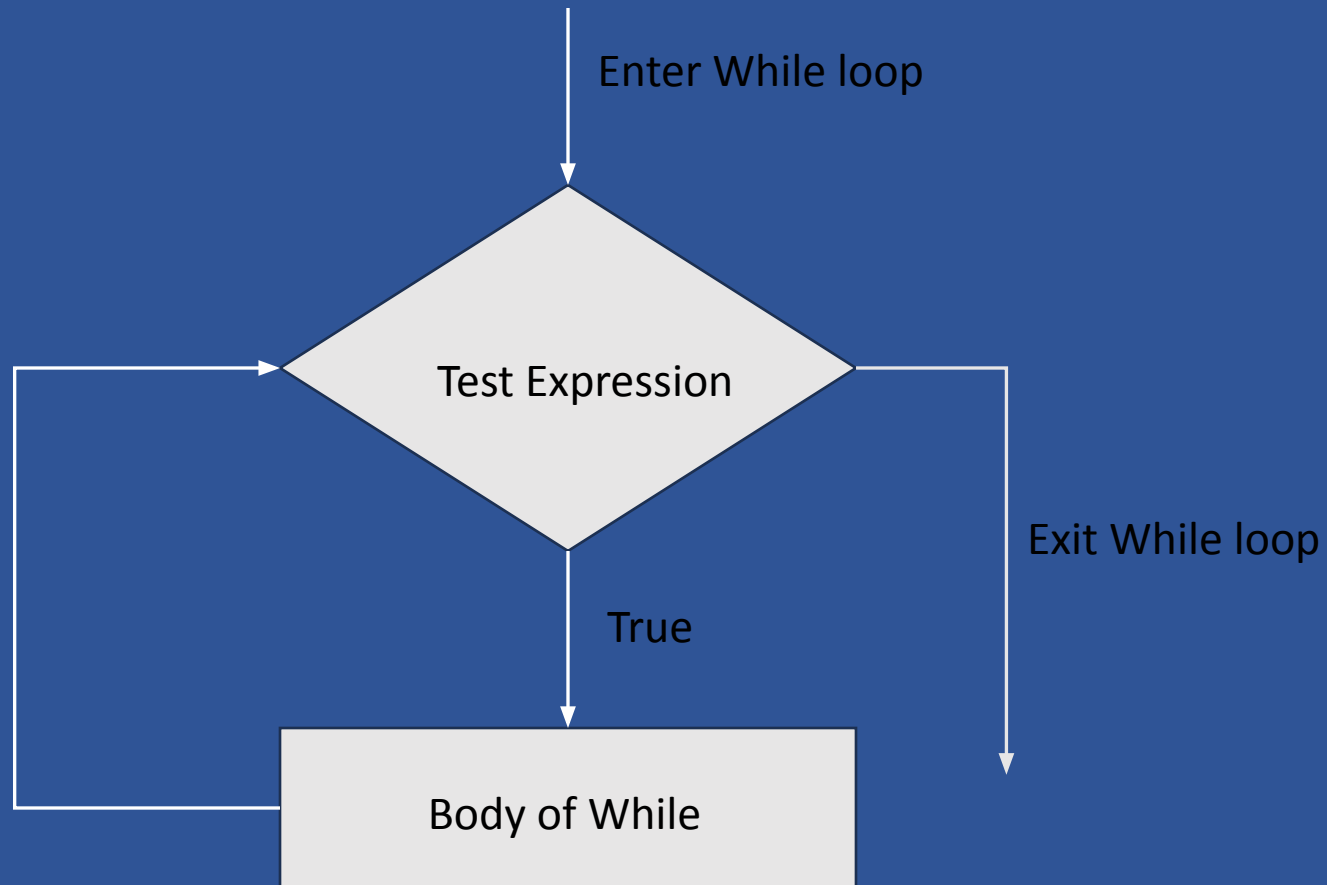
```
    Statements to be executed.....
```

Looping Statements

Get your salary credited
at the end of **each** month



While Loop



SYNTAX

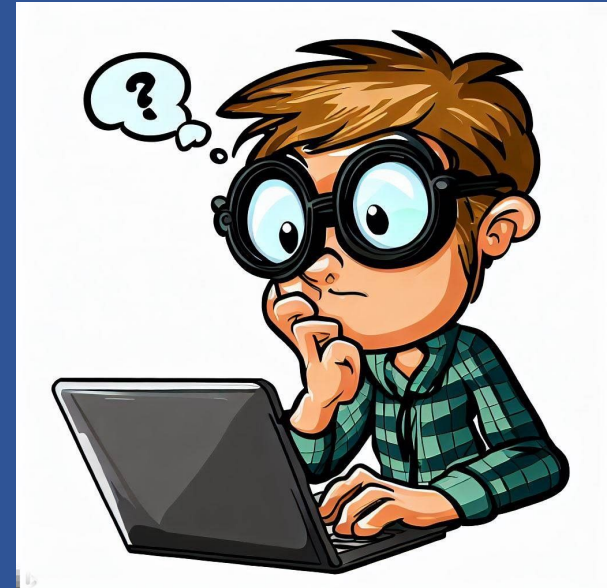
```
while condition:  
    Execute Statements
```

For Loop

For Loop is used to iterate over a sequence (tuple, list, dictionary.....)

For loop syntax

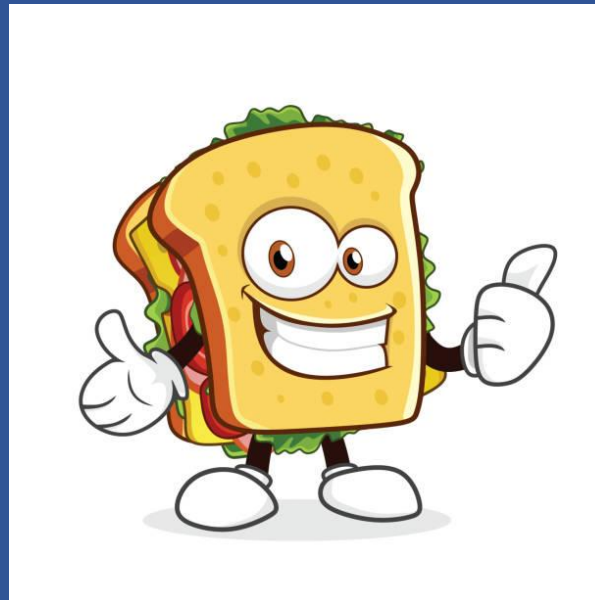
```
for val in sequence:  
    body of for loop
```



Functions

Function (Recipe): A function like a recipe is a set of instructions that perform a specific task or operation. Just like a recipe, a function has a name and can take inputs (ingredients) and produce an output (result).

Input(Ingredients): The values given to a function like bread, tomato etc.

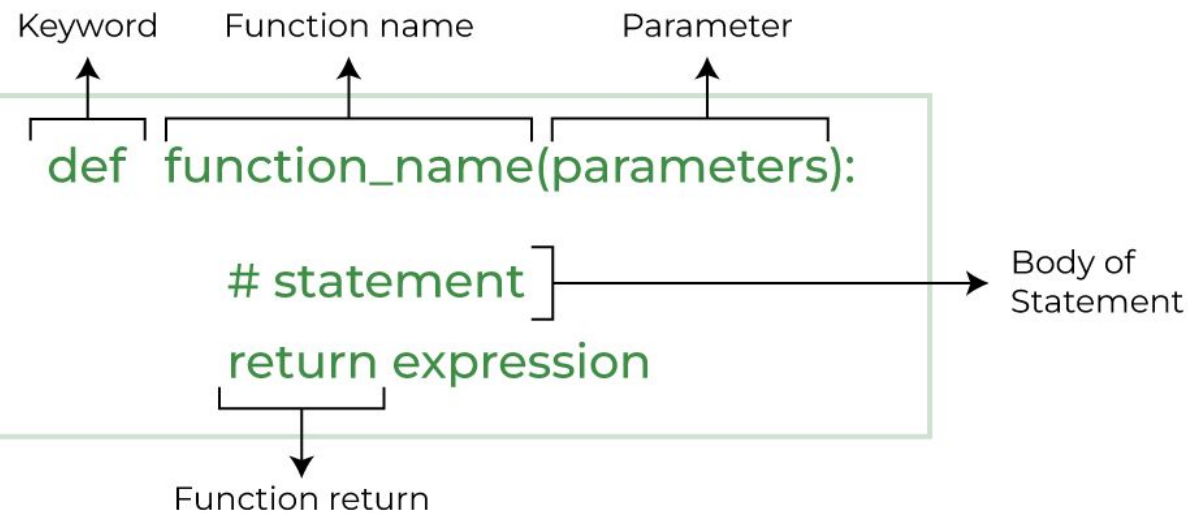


Output (Result): It is the result obtained from the function like Sandwich.

Functions can be reused multiple times, just like a recipe can be used to make the same dish repeatedly. They allow us to write code once and use it in different parts of a program.

Python Functions

Function is a block of code which performs a specific task



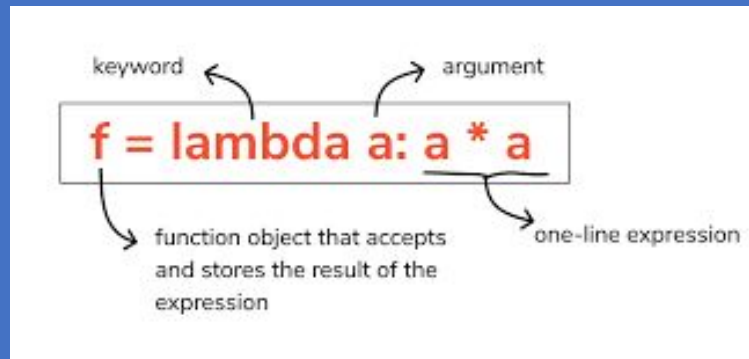
```
def function_name(parameters):
    # statement
    return expression
```

The diagram illustrates the syntax of a Python function definition. It shows the following components:

- Keyword:** The word `def` is labeled as the keyword.
- Function name:** The identifier `function_name` is labeled as the function name.
- Parameter:** The text `(parameters)` is labeled as the parameter.
- Body of Statement:** The indented lines `# statement` and `return expression` are grouped together and labeled as the body of the statement.
- Function return:** The `return` statement is specifically labeled as the function return.

Python Lambda Functions

A small anonymous function , taking any number of arguments, having one expression



The diagram shows the code `f = lambda a: a * a` with several annotations. An arrow points from the word 'keyword' to 'lambda'. Another arrow points from the word 'argument' to 'a'. A third arrow points from the text 'function object that accepts and stores the result of the expression' to 'f'. A fourth arrow points from the text 'one-line expression' to the expression 'a * a'.

```
f = lambda a: a * a
```

keyword

argument

function object that accepts and stores the result of the expression

one-line expression