Python

→ What is Python?

Python is a popular programming language.
It is used for:

- web development (server_side)
- software development
- mathematics
- system scripting

→ Syntax

↳ Python Indentation

Indentation refers to the spaces at the beginning of a code line.

Python uses indentation to indicate a block of code.

Example:

```
if 5 > 2:
    print("Five is greater than two!")
```

the space here is important

We have to use the same number of spaces in the same block of code, otherwise Python will give us an error

Python Variables

In python variables are created when we assign a value to it.

Example:

```
x = 5
y = "Hello World!"
```

Comments

Python has commenting capability for purpose of in-code documentation.

Comment start with a #

Example:

```
# This is a comment
print("Hello world")
```

For multiline comments we use:

```
"""
This is a comment
written in more
than one line
"""
```

→ Variables

Variables are containers for storing data values

↳ Creating Variable

Python has no command for declaring variables
Variables do not need to be declared with any particular type, and can even change type after they have been set

Example:

```
x= 4   # x is of type int
x="Sally" # x is now of type str
print(x) # output : sally
print (type(x))  # we can get the type of
                   a variable with the
                   type() function
```

↳ Variable Names

A variable can have a short name like (x and y) or a more descriptive name (age, total_price...)

- A variable name must start with a letter or the underscore character (_)
- A variable name cannot start with a number
- A variable name can only contain alpha-numeric characters and underscores (A-z, 0-9, and _)
- Variable names are case-sensitive (age, Age, and AGE are 3 different variables)

Multi Words Variable Names

There are several techniques we can use to write multi words variable names :

Camel Case

```
myVariableName = "John"
```

Pascal Case

```
MyVariableName = "Jhon"
```

Snake Case

```
my_variable_name = "Jhon"
```

Output Variables

The python print() function is used to output variables.

In the print() function, we output multiple variables, separated by a comma ",", or the operator +

Example:

```
x = "Python "
y = "is "
z = "awesome"
print (x+y+z)
print (x, y, z)
```

} some Output: Python is awesome

For numbers, the + characters work as mathematical operator

Example:

```
x = 5
y = 10
print (x + y)  # output : 15
```

In the print() function, when we try to combine a string and a number with the + operator, python will give an error.:

```
x = 5
y = "John"
print (x + y)  # Type Error
```

So the best way to output multiple variables in the print() function is to separate them with commas, wich even support different data types:

```
x = 5
y = "John"
print (x, y)  # output: 5 John
```

↳ Global Variables

Variables that are created outside a function are known as global variables

Example:

```
x = "awesome"

def myfunc():
  print("Python is" + x)

myfunc() # output: Pythoon is awesome
```

If we create a variable with the same name inside a function, this variable will be local and can only be used inside the function. The global variable with the same name will remain as it was, global and with the original value

Example:

```
x = "awesome"
def myfunc():
  x = "fantastic"
  print("Python is" + x)
myfunc() # output: python is fantastic

print("Python is" + x) # output: python is awesom
```

→ The global Keyword

Normally, when we creat a variable inside a function, the variable is local, and can only be used inside the function.

To creat a global variable inside a function, we can use the global Keyword.

Example:

```
def myfunc():
  global x
  x = "fantastic

myfunc()
print("Python is" + x)  # Output: Python is fantastic)
```