

- **created by:** Sudip Ghimire
- **URL:** <https://www.sudipghimire.com.np>
- **Github:** <https://github.com/ghimiresdp>

# Chapter 1: Python Basics

---

## Table of Contents

- [Chapter 1: Python Basics](#)
  - [What is python?](#)
  - [What is an interpreter](#)
  - [Features of python](#)
    - Python is easy to code
    - Among programming languages, Python grammar is closest to English grammar.
    - Some notable features of python for programmers coming from other programming languages
  - [Why we choose python?](#)
  - [Installing Python](#)
    - [Windows](#)
    - [Linux and Mac OS](#)
  - [Getting Started](#)
  - [Hello World with IDLE](#)
  - [Creating, editing, and running python files.](#)

## What is python?

- An interpreted, high-level, general-purpose programming language
- Created by *Guido Van Rossum*, First released in 1991
- It can be used in:
  - Software and Web Development
  - Mathematics, Data Science and Artificial Intelligence
  - Games and Simulation
  - Scripting

## What is an interpreter

- A computer program that directly executes instructions without requiring to be compiled to a machine level code.
- The code executes line-by-line when required
- Unlike compiler, it runs instructions until the line where error is spotted
- It does not generate intermediary code, hence, is highly efficient in terms of memory
- Drawback: Needs to be interpreted every time
- Programming languages such as Perl, Python, MATLAB, QBasic are interpreted programming languages

## Features of python

Python is easy to code

Python syntax is easy and the code is cleaner so that anyone can understand the python code easily.

## Hello world program in C++ and Java vs Python

### C++

```
#include <iostream>

int main() {
    std::cout << "Hello World!";
    return 0;
}
```

### Java

```
// Java code
class Main{
    public static void main(String args[]){
        System.out.println("Hello World!")
    }
}
```

### Python

```
print('Hello World!')
```

Among programming languages, Python grammar is closest to English grammar.

Example: Checking if a string is in an array/list of strings **C++**

```
bool in_array(const std::string &value, const std::vector<std::string> &array)
{
    return std::find(array.begin(), array.end(), value) != array.end();
}

int main() {
    std::vector<std::string> tab {"Jon", "Jane"};

    if (in_array(n, tab)) {
        cout << "The name is probably a placeholder"
    }
}
```

### python

```
if name in ['Jon', 'Jane']:
    print("The name is probably a placeholder")
```

other examples in python

```
# checking if condition is true or not
if condition is not this_value:
    do_this_task()

value = 'Even' if x%2==0 else 'Odd'
```

Some notable features of python for programmers coming from other programming languages

Unlike other programming languages, python do not need to end the statement with semicolons.

- When we insert a new line, python automatically identifies whether statement is complete or not.
- If the brackets are open, it do not terminate the statement with new line.

```
x = 5      # terminates the statement

person = {                                # does not terminate the statement
    'name': 'Jon Doe'
}      # statement will be terminated here
```

- Python do not need curly brackets or braces to identify a block of code however *indentation* is used.
- standard indentation type in python is space however tab character in modern IDEs and code editors are automatically converted into spaces by default.
- standard indentation size is 4.

```
for value in range(1,10):
    print(f"I am inside the loop and the value is {value}")
print('I am outside the loop now')
```

- We can use single or double quotation marks for strings.
- we can even use one kind of quotation mark inside another without escaping the character.
- however using similar kind of quotation inside a string requires escaping the character

```
print("I'm Jon Doe.")  
print('He said, "I\'m Jon Doe".')
```

- Python variables are dynamically typed. Unlike other statically typed languages, we do not need to specify the data type in python.

```
x = 5          # it automatically assigns data type as int  
y = 5.5        # it automatically assigns data type as float  
x = "John"     # it replaces the type of data to string from int (previously  
               defined: x = 5).
```

## Why we choose python?

- Compatible with all major operating systems
- Larger community support
- Easier Syntax, cleaner code, shorter instructions
- Can be used in both procedural and Object-Oriented approach
- Easier to develop, maintain and support

## Installing Python

### Windows

Installing python in windows is so easy. You can just browse <https://python.org> and download the latest/stable version of your choice and follow the instructions.

Note: It is recommended to **add python to path** while installing.

### Linux and Mac OS

Linux and Mac os comes with python pre-installed so we do not need to download the one. If we want to install the latest version, then we can follow instructions from <https://python.org>

## Getting Started

To verify the python is installed in your machine, you can run the following in your command prompt or the terminal

```
$ python --version
```

## Hello World with IDLE

IDLE is a built-in editor for python. It comes bundled with python so if you install python you should have IDLE installed on your system.

once you open IDLE, you should see the python terminal with the shell ready to take the input.

The window should show something similar to the following box

```
>>>
```

Now you can write the following and press `Enter` the output will be shown in the shell below

```
>>> print("Hello world!!")  
Hello world!!
```

## Creating, editing, and running python files.

We can create a new file and save with `.py` extension to create a new python file. Since python file is a normal text file, you should be able to open with any text editor. If you want better syntax highlighting and code completion, you can use IDEs or code editors such as VSCode.

Once opened, you can directly type into the document and save it. The following is an example of a python file `hello_world.py` that adds 2 numbers and prints out the result.

```
# filename: hello_world.py  
x = 5  
y = 10  
print('The sum is:', x + y)
```

To run the python file, we can now open our favorite terminal and run the command below:

```
$ python hello_world.py
```

If you're not in the location where the python file exists, you can give the full path of the file.

### Bash Example:

```
$ python /home/john/hello_world.py  
  
The sum is: 15
```

If you're using CMD or powershell, The command would be like:

```
C:\Users\john> python C:\Users\john\Documents\hello_world.py
```

The sum is: 15