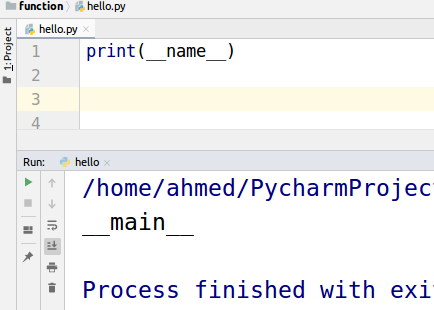
Python Main Function with Examples: Understand main

## What is the main function?

The first function which is called when you run a program in any programming language is the main function. Python specifically does not have the main function as it is an interpreter language but we can create the main function and run it when the python file is run using special variable \_\_name\_\_. Let us first see what is \_\_name\_\_

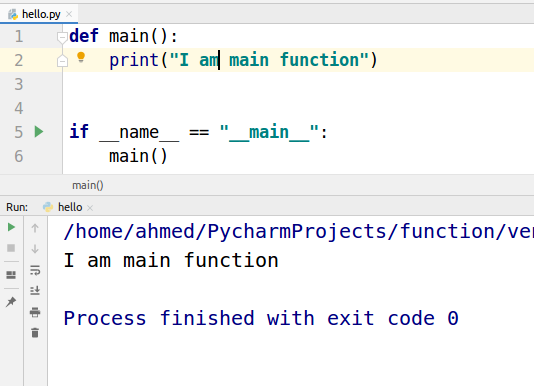
\_\_name\_\_ is a special variable

Python automatically makes \_\_name\_\_ = “\_\_main\_\_” as you can see below

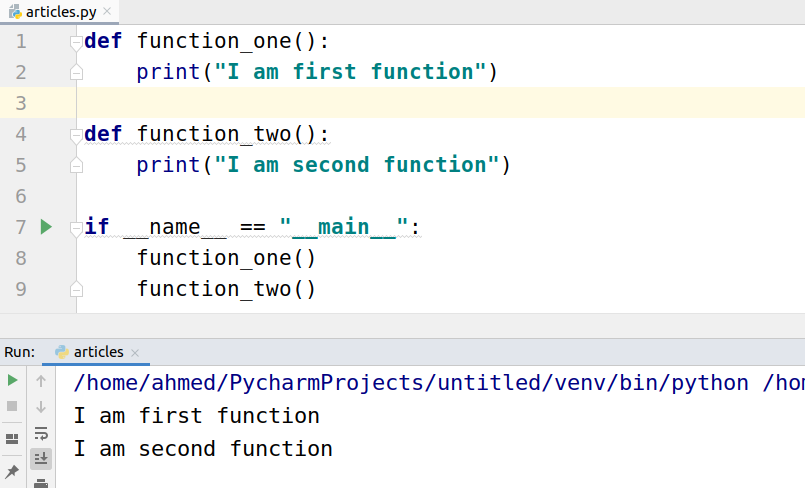


So we will write a condition to check whether \_\_name\_\_ == \_\_main\_\_ then call the main function. This checking will make sure that this hello.py is running this script. We will take a deep look into this when we will discuss importing other files than you will be able to see the importance of this.

|  |
| --- |
| def main():  print("I am main function")   if \_\_name\_\_ == "\_\_main\_\_":  main() |



We can call different functions from the main functions. Let’s have a look at the example.



## Understanding \_\_name\_\_ == “\_\_main\_\_”

Take a look at the screenshot below. When we run the articles.py file \_\_main\_\_ is set to “\_\_name\_\_”

|  |
| --- |
| def function\_one():  print("I am first function")   if \_\_name\_\_ == "\_\_main\_\_":  function\_one() print("Value in built variable name is: ", \_\_name\_\_) |

## 

But when we import the articles.py file in secondFile \_\_main\_\_ is set to “articles”

This \_\_main\_\_ function gives us control over when to run the main function.

|  |
| --- |
| import articles print("Hello") |

