Writing your first program

Python's interactive mode is useful as a simple calculator, but you can create programs for more extensive functionality. A Python program is simply a plain text file script created with an editor, such as Windows' Notepad, that has been saved with a ".py" file extension. Python programs can be executed by stating the script file name after the **python** command at a terminal prompt.

Beware

Don't use a word processor to create program files as they add format information to the file.

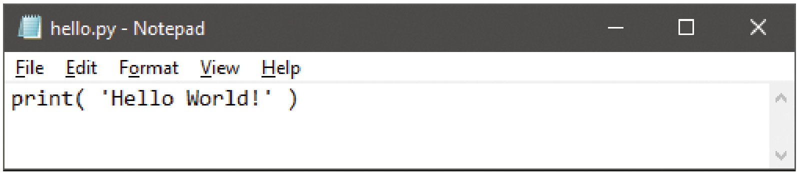
The traditional first program to create when learning any programming language simply prints out a specified greeting message. In Python, the **print()** function is used to specify the message within its parentheses. This must be a string of characters enclosed between quote marks. These may be **" "** double quote marks or **' '** single quote marks – but not a mixture of both.

hello.py

1. On Windows, launch any plain text editor such as the Notepad application

2. Next, precisely type the following statement into the empty text editor window
   print('Hello World!')

3. Now, create a new directory at **C:\MyScripts** and save the file in it as **hello.py**

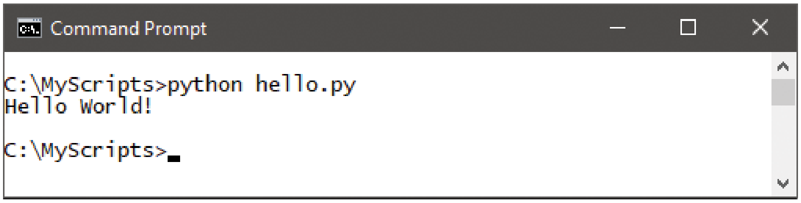


```
print( 'Hello World!' )
```

Hot tip

The directory created at **C:\MyScripts** will be used to contain all Windows examples in this book.

4. Finally, launch a Command Prompt window, navigate to the new directory and precisely enter the command **python hello.py** – to see the Python interpreter run your program and print out the specified greeting message



```
C:\MyScripts>python hello.py
Hello World!

C:\MyScripts>_
```

…cont'd

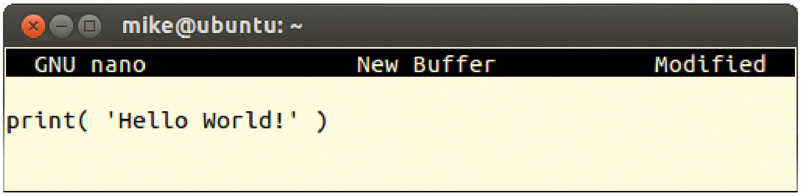
The procedure to create the traditional first Python program is identical on Linux systems to that on Windows systems. It is, however, important to be aware, on any platform where different versions of Python are installed: you must use the correct command to call upon the particular Python interpreter required. This is especially important on Linux systems that often ship with the Python 2.7 version installed as their default. This means that the command **python** will assume you want to call that interpreter. Where Python 3.3 is installed, and you want to call that particular interpreter to process a script, you must use the command **python3.3** to explicitly call upon that version's interpreter.

1. On Linux, launch any plain text editor such as the Nano application

2. Next, precisely type the following statement into the empty text editor window
   `print( 'Hello World!' )`

hello.py

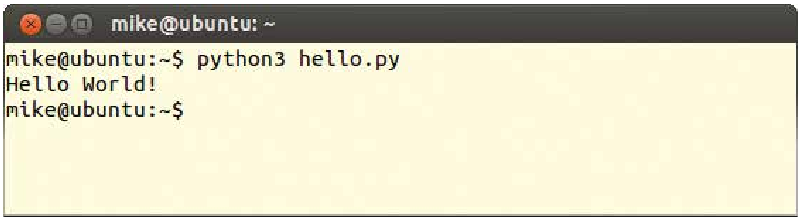
3. Now, save the file in your home directory as **hello.py**



```
  GNU nano                   New Buffer                Modified

print( 'Hello World!' )
```

4. Finally, launch a Terminal window and navigate to your home directory and precisely enter the command **python3.3 hello.py** – to see the Python interpreter run your program and print out the specified greeting message



```
mike@ubuntu:~$ python3 hello.py
Hello World!
mike@ubuntu:~$
```

Don't forget

All further examples in this book are illustrated on Windows (simply because that platform has the most users) but they can also be created and executed on Linux.