

Hello World! Your First Python Program

In this exercise you will create a file containing the simplest Python program there is, and then you will run your program and save the output.

It is a long and venerable tradition that whenever you learn a new computer programming language or system, the first thing you do is cause the computer to display the phrase “Hello, World!” This dates back to the book “*The C Programming Language*” by Brian W. Kernighan and Dennis M. Ritchie, but has been applied to learning many other computer languages. This is not as trivial as it sounds. You have to accomplish these major steps:

1. Create a new file, with the correct file extension.
2. Edit the file to put into it the correct instructions.
3. Cause the instructions in the file to be executed (*i.e.* “run it”).

These are the things you need to be able to do in order to run *any* Python program, so these are what you need to learn first.

Python “print”

A Python program is called a “script,” because when you run it the computer reads instructions from the file, line by line, and does what each line tells it to do, similar to an actor reading lines from a script. One simple but important Python instruction is “**print**”. Thus, to print the desired text requires a script with the single line:

```
print("Hello, World!")
```

That’s all.

Learning Python involves first learning more of these basic instructions, and then learning how to put them together to tell the computer how to do something useful.

Creating and Editing Scripts

There are many ways to put this instruction into a script. You can use a simple text editor, such as Notepad (but not Microsoft Word or Wordpad!). You can use a more powerful text editor such as Emacs (which works the same on Windows, Mac, and Linux) or nano (on Raspberry pi, Mac, or Linux). But the easiest way to get started is to use Thonny or IDLE. Each not only lets you edit your programs, it also provides you an easy way to run and test your programs. Both tools provide an “*Integrated Development Environment*,” also known as an “IDE”.

So to get started, do the following:

1. Launch IDLE or Thonny, as you learned to do in the previous exercise.
2. If you are using Thonny, a window will open with a pane at the top marked “<untitled>”.

If you are using IDLE, the window that opens will likely be the “Shell” window, not the file editor. To open the file editor window pull down the “File” menu to “New File.” The new window will be labeled “untitled”.

3. Type the text you wish to add to your script in the “untitled” editing pane or window.
4. Pull down the “File” menu to “Save as...” and save your file to your Desktop (or a folder you choose) with the name `hello.py`.

It is important that the file end with `.py`, because that is what distinguishes it to the computer as a Python script.

Comments

One of the most important elements in any computer language does nothing. It’s called a “comment”. A comment is text which is intended for a human reader, but which is ignored by the computer. Do not underestimate the importance of comments. You will want to leave information in your programs for others who read them. You can also leave information for yourself later on, when you’ve forgotten the details of what you intended. Well written programs always contain appropriate and instructive comments.

In Python comments are indicated by the “#” character. A line beginning with # is simply ignored by the computer. You can also put comments on the same line as an instruction; anything on a line past the #, to the end of the line, is ignored by the computer.

You can put anything in the comments of your program, but some information is so important that you should always include it. For every program you write you should begin the file with comments that include:

- The name of the author
- How to contact the author (*e.g.* your e-mail address or your GitHub ID)
- The date the program was written, or at least started (or finished)
- A brief description of the purpose of the program

Other optional information could also be included, such as:

- A version number, if this is a revision of earlier work.
- A brief description of changes for a new version.

- Citing sources from which this work is derived. (Copying from someone and improving on their work is a very common thing, but give them credit for their contribution.)
- The name of the project or class for which it was written.
- The author’s organizational affiliation (your school or company).
- The name of the file (but this might change some day)
- Licensing restrictions, if any. As the author of the work, you hold the copyright to the contents of the file (unless it is a copy of somebody else’s work, or a work for hire), and you can decide if anybody else is allowed to copy it. You might wish to distribute it freely under an open source license, or you may wish to state how distribution is restricted. You can declare your intentions in a comment at the beginning of the file.

You can think of the comments at the beginning of your program as acting like the cover sheet on an essay or paper, in that – in the least – it should identify the author and date of the work, and possibly the class and purpose as well.

Another useful thing to know is that blank lines are also ignored by the computer. Why is this useful? Because you can add space to your programs to improve readability, just as blank lines between paragraphs or sections can make a research paper or essay easier to read.

Running it

To run your script from within IDLE simply pull down the “Run” menu (at the top of the window) to “Run Module”. You can also press the function key F5, by holding down the “fn” key like a shift key and pressing the button marked “F5”.

Thonny is similar. Pull down the “Run” menu at the top of the window to “Run current script,” or hold down “fn” and press “F5”. Thonny also provides a green “Play” button at the top of the window which is an even easier way to run your script.

Saving the Output

When you run a Python script, the output appears in the Shell window (for IDLE) or the Shell pane of the window (Thonny). As in the previous exercise, you could copy and paste the output to an email or direct message sent to your instructor, but for future work you will want to save the output of your scripts to a text file. So we will save the output from this script to a text file.

In IDLE this is easy: select the Shell window and pull down the “File” menu to “Save as...” Be sure that the name of the file you save ends in `.txt` to indicate that it is a text file. If it ends in `.py` (the default) it could be mistaken for a python script file.

With Thonny saving the output is a little bit harder, because there is no direct way to save the contents of the Shell pane of the window.* As you did before, select and copy the text in the Shell window (Control-C in Windows, ⌘-C on a Mac). Then pull down the “File” menu to “New”, and paste what you copied into that window (using Control-V in Windows, ⌘-V on a Mac). Then pull down the “File” menu to “Save as...” and save the file with a `.txt` file extension. Be sure it has the `.txt` extension.

Python Documentation

There are many useful sources of information on the Internet for new Python programmers. Perhaps too many for beginners – it can seem overwhelming at first. Here are some suggestions for how to learn more about Python:

- Very organized and complete information about Python can be found on the website of the Python Software Foundation at <http://docs.python.org>.
- If you are a complete beginner, you may wish to start with the “*Python for Beginners*” page at <https://www.python.org/about/gettingstarted/> .
- There is also a Python Tutorial, at <https://www.python.org/tutorial/> .
- The website known as “Real Python” at <https://realpython.com/> has a large number of tutorials and other resources for learning python, including different learning paths based on your interest and previous experience.
- You can often find answers to particular questions using Google, which in turn will often take you to the StackOverflow website.
- There are videos on YouTube and on LinkedIn Learning that can help you learn Python, or at least parts of it.

There are also literally hundreds of books written about Python. The set of exercises you are now beginning does not depend on using any one book or tutorial or website. Pick whichever works best for you.

The problem you will have is not to find information about Python – gobs of it are available to you on the Internet. The problem is sorting through it to find the resources that are easiest for you to use and address the questions you have as you increase your skills. Your instructor may be able to give you some useful suggestions and answer any particular questions you might have.

* Not yet, at least. The Thonny developers have been receptive to adding it, so watch for that in a future release.

Assignment

Your goal for this exercise is to create a Python script (a `.py` file) containing the simplest Python program ever, and then to run it to display the famous phrase “Hello, World!”.

To complete this exercise you must do the following:

1. Create a Python script which uses `print()` to display the text “Hello, World!” (exactly as shown).
2. Your script should include comments at the beginning of the script (before any commands) which contain at least the minimal information listed above, and hopefully more than that.
3. Save your script to a file with a name that includes your last name and the exercise number – something like “Baumfree01.py” if your name is Isabella Baumfree. In both IDLE and Thonny you can easily do this by pulling down the “File” menu to “Save as...”.
4. Run your script, and save the output to a text file with the same name, but with the “.txt” extension. For example: “Baumfree01.txt”
5. Submit to your instructor copies of both your script (the `.py` file) and the output it produces (the `.txt`) file. You can do this by sending the files via email, or by direct message, or by uploading to a course website. Use whichever method suggested by your instructor.