

## The Python IDLE: Introduction

**NOTE:** All the Python exercises assume the use of the PYTHON 2.x version of the programming language. Current version at of this writing is Python 2.7.8

Python is available at <https://www.python.org/>

Why Python is awesome: <https://www.python.org/about/success/>

This tutorial teaches the basics of how to use the Python programming language's Integrated DeveLopment Environment (IDLE). The IDLE is a very powerful tool for learning Python as it can give instant feedback to the user and allows the user to rapidly test and debug their code.

INSTALLATION: (If Python and the IDLE are not already installed.)

<https://www.python.org/downloads/>

## The Python IDLE: Your first Python code!

Open the IDLE. A window should open that looks something like this:

```
Python 2.6.5 (r265:79359, Mar 24 2010, 01:32:55)
[GCC 4.0.1 (Apple Inc. build 5493)] on darwin
Type "copyright", "credits" or "license()" for more information.

*****
Personal firewall software may warn about the connection IDLE
makes to its subprocess using this computer's internal loopback
interface. This connection is not visible on any external
interface and no data is sent to or received from the Internet.
*****

IDLE 2.6.5
>>>
```

You can type in and run your very first Python program in the Python Interpreter. By Coder Law, it must print "Hello world!"

```
IDLE 2.6.5
>>> print "Hello world!"
Hello world!
>>>
```

```
IDLE 2.6.5
>>> print "Hello world!"
Hello world!
>>> 19.5 + .5
20.0
>>> |
```

Look! It's a calculator too!

## The Python IDLE: Instant feedback

Let's do something a little more interesting. Try this:

A programming variable called "my\_list"

```
>>> print "Make a list of numbers."
Make a list of numbers.
>>> my_list=range(5,10)
>>> print my_list
[5, 6, 7, 8, 9]
```

This is a programming function. Notice the ().

Here's a little bit of code. The "for loop" prints out all the number in your list:

```
>>> print "Let's print out the numbers:"
Let's print out the numbers:
>>> for number in my_list:
    print number,

5 6 7 8 9
>>> |
```

The IDLE let's you know if you've made a mistake.

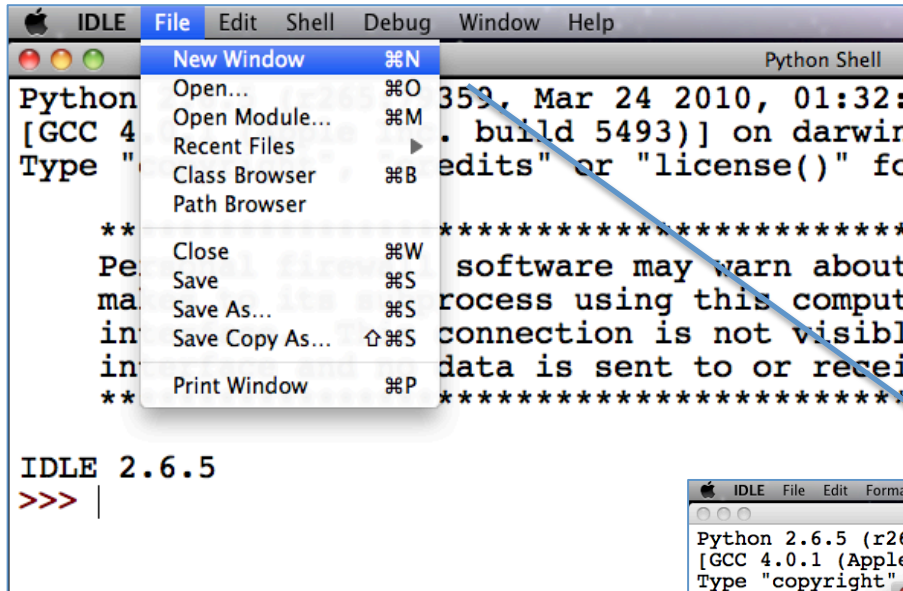
```
>>> for number in my_list:
    print num,

Traceback (most recent call last):
  File "<pyshell#10>", line 2, in <module>
    print num,
NameError: name 'num' is not defined
>>>
```

You made a name error. Oops!

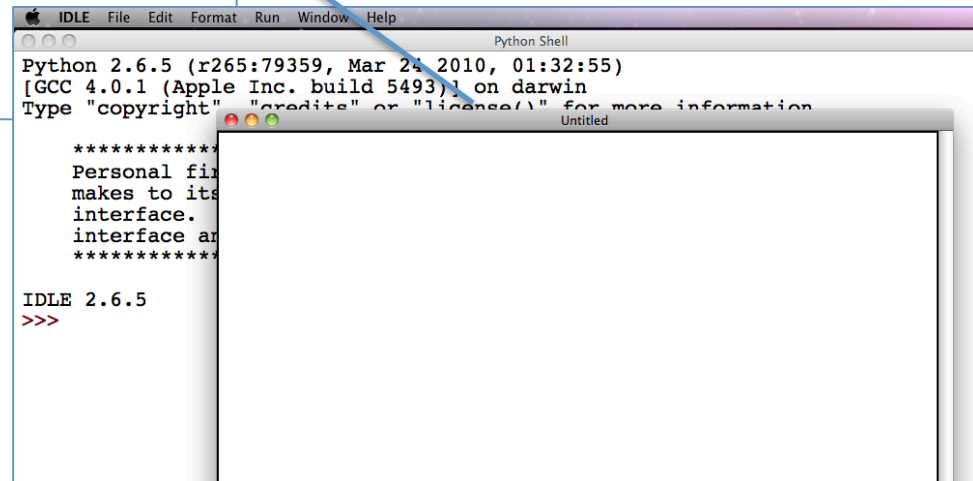
## The Python IDLE: Opening a file

Once you quit the IDLE, you lose your work. But, of course, you want to save your wonderful program and keep improving it.



### STEP 1: File -> New

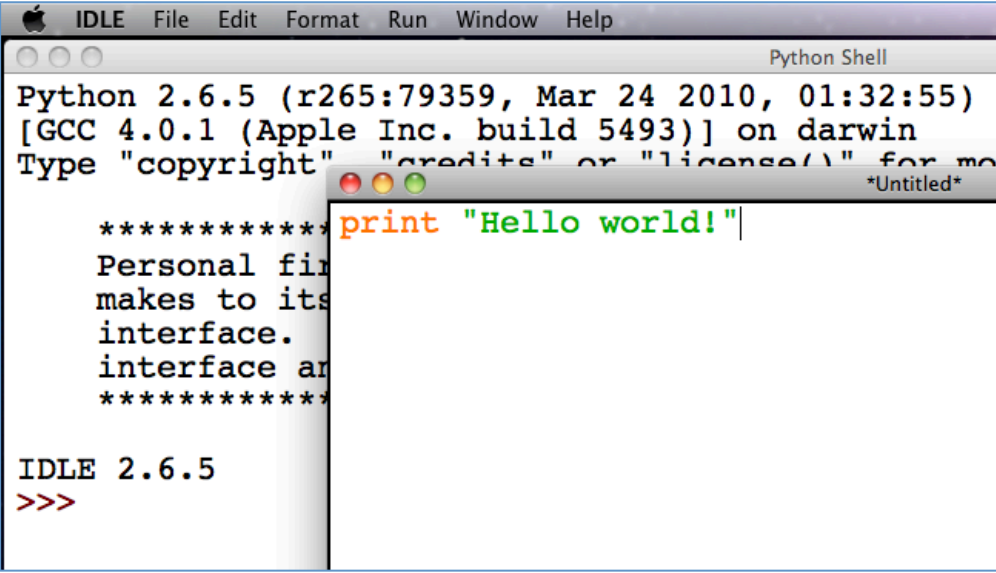
This opens a file window where you can type your wonderful code.



## The Python IDLE: Saving your program

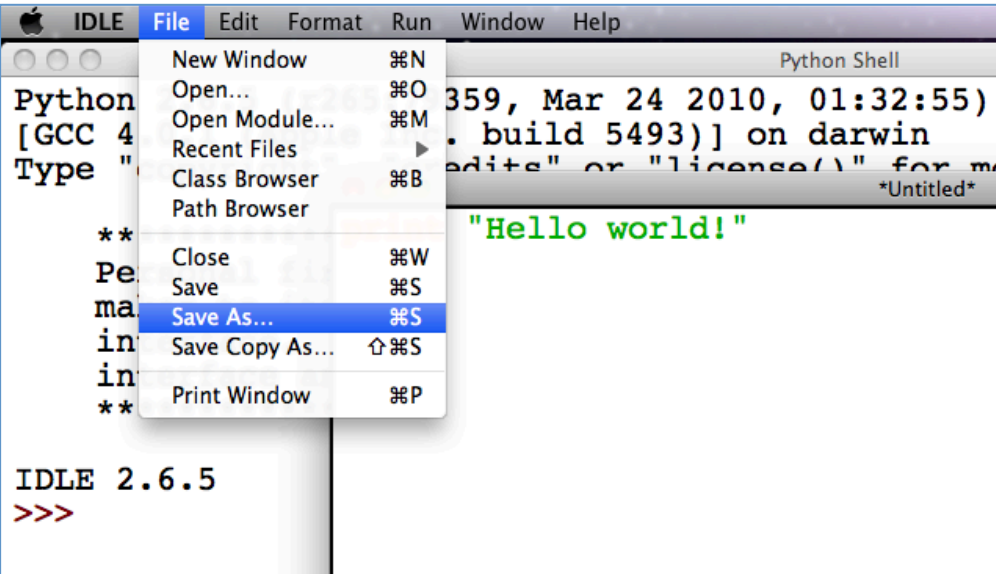
Write your code in the window and save the file.

STEP 2: Type in your code.



The screenshot shows the Python IDLE 2.6.5 interface. The top menu bar includes 'IDLE', 'File', 'Edit', 'Format', 'Run', 'Window', and 'Help'. A 'Python Shell' window is open, displaying the Python version and system information: 'Python 2.6.5 (r265:79359, Mar 24 2010, 01:32:55) [GCC 4.0.1 (Apple Inc. build 5493)] on darwin'. Below this, it prompts the user to type 'copyright', 'credits', or 'license()' for more information. An editor window titled '\*Untitled\*' is overlaid on the shell, containing the code: `print "Hello world!"`. The IDLE 2.6.5 logo and the prompt `>>>` are visible at the bottom of the shell window.

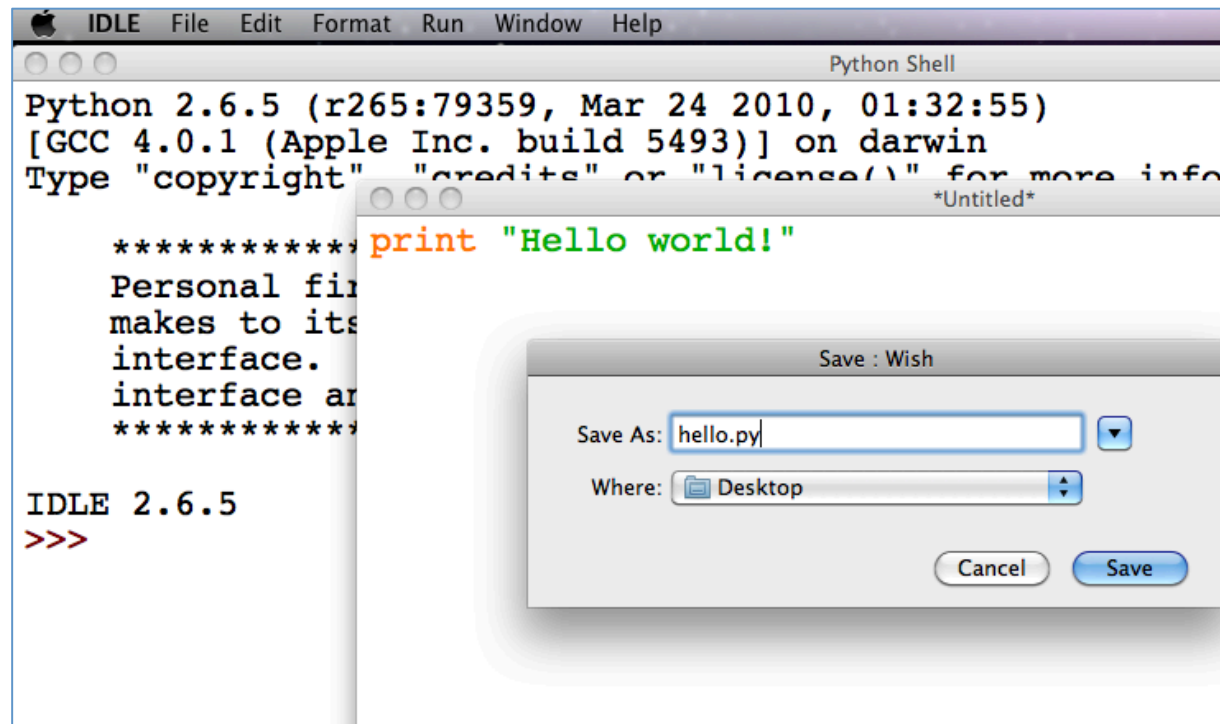
STEP 3: Save it.



The screenshot shows the Python IDLE 2.6.5 interface with the 'File' menu open. The menu items are: 'New Window' (⌘N), 'Open...' (⌘O), 'Open Module...' (⌘M), 'Recent Files', 'Class Browser' (⌘B), 'Path Browser', 'Close' (⌘W), 'Save' (⌘S), 'Save As...' (⌘S), 'Save Copy As...' (⇧⌘S), and 'Print Window' (⌘P). The 'Save As...' option is highlighted. The background shows the same Python Shell and editor window as in the previous screenshot, with the code `print "Hello world!"` visible in the editor.

## The Python IDLE: Name your file

I recommend putting a “.py” at the end of the file name. The IDLE will recognize it as a Python file and keep the nice color coding.



## The Python IDLE: Changing and Saving your file.

**SAGE ADVICE:** The instant feedback in the interpreter is nice, but it is MUCH BETTER to do your work in a file. Then you can quickly save your changes and run the file to see if it works.

The screenshot shows the Python IDLE 2.6.5 interface. The 'File' menu is open, with 'Save' (⌘S) highlighted. The Python Shell window displays the output 'Hello world!'. The code editor window shows the following code:

```
output="Hello world!"  
  
for i in range(5):  
    print output
```

Two red callout boxes provide additional information:

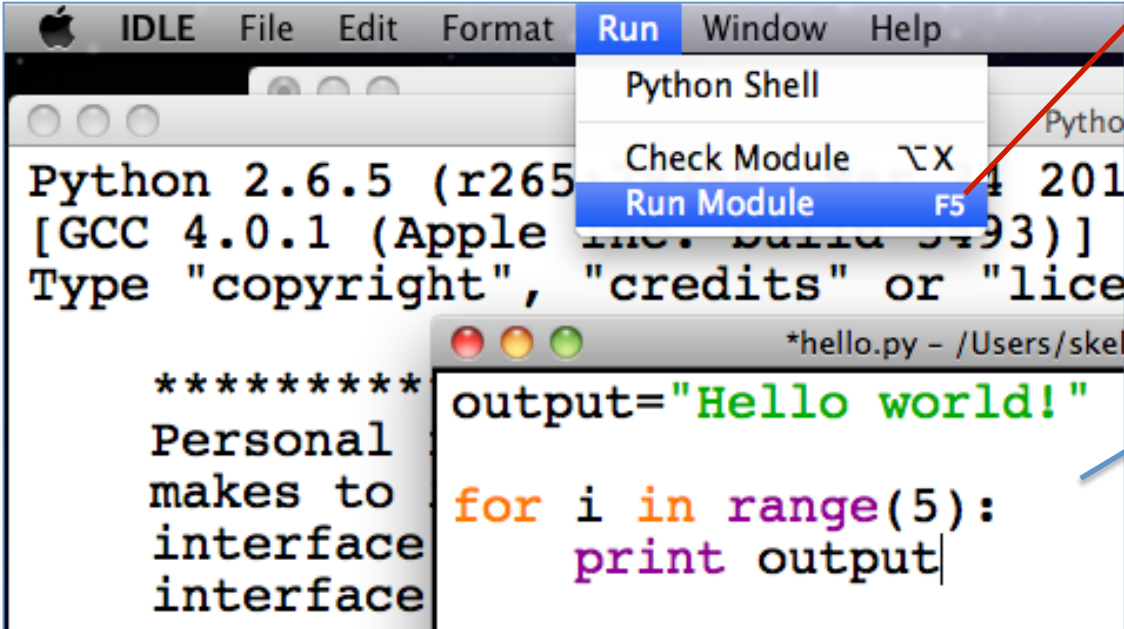
- A box pointing to the 'Save' menu item contains the text: "Save Changes! Always have to do this. The fastest (shortcut) way is to type **CMD-S** on a Mac, or **CNTRL-S** on a PC."
- A box pointing to the code editor contains the text: "New code."

## The Python IDLE: Running your file – Output in the Python Interpreter

To RUN the file, just type the F5 button on your keyboard or go to the RUN dropdown menu.

YOUR BEST FRIENDS: CMD-S (CTRL-S) and F5

To quickly save and test your code, use the shortcut keys. So FAST!



The screenshot shows the Python IDLE 2.6.5 interface. The 'Run' menu is open, highlighting 'Run Module' with the shortcut key F5. The code editor contains the following code:

```
*****  
Personal  
makes to  
interface  
interface  
  
output="Hello world!"  
  
for i in range(5):  
    print output
```

A red box highlights the 'Run Module' option in the menu, with a red arrow pointing to it. A text box next to it says: "Runs your code. Shortcut key: F5".

A blue box highlights the output window, with a blue arrow pointing to it. The output window shows the following text:

```
CODE OUTPUT  
IDLE 2.6.5  
>>> =====  
=====  
>>>  
Hello world!  
Hello world!  
Hello world!  
Hello world!  
Hello world!  
>>> |
```