Python Programming
Module 1: Getting Started

Learning objectives

1. Installing Python to your system.
2. Installing a text editor to make it easier to write Python code.
3. Executing snippets of Python code in a terminal session.

Setting up your Programming Environment

Here, we'll look at the two major versions of Python that are currently in use and outline the steps to set up Python on your system.

Python 2 and Python 3

Two versions of Python are available: Python 2 and the newer Python 3. Most changes are incremental and hardly noticeable, but in some cases code written for Python 2 may not work properly on systems with Python 3 installed. If both versions are installed on your system or if you need to install Python, use Python 3.

Python on Different Operating Systems

Python is a cross-platform programming language, which means it runs on all the major operating systems. Any Python program you write should run on any modern computer that has Python installed. However, the methods for setting up Python on different operating systems vary slightly.

Python on Linux

Linux systems are designed for programming, so Python is already installed on most Linux computers.

Checking Your Version of Python

Open a terminal window by running the terminal application on your system (in Ubuntu, you can press CTRL-ALT-T). To find out whether Python is installed, enter `python` with a lowercase `p`. You should see output telling you which version of Python is installed and a >>> prompt where you can start entering Python commands, like this:

```
$ python
Python 2.7.6 (default, Mar 22 2014, 22:59:38)
[GCC 4.8.2] on linux2
Type "help", "copyright", "credits" or "license" for more
information.
>>>
```

This output tells you that Python 2.7.6 is currently the default version of Python installed on this computer. When you've seen this output, press CTRL-D or enter `exit()` to leave the Python prompt and return to a terminal prompt.

To check for Python 3, you might have to specify that version; so even if the output displayed Python 2.7 as the default version, try the command python3:

```
$ python3
Python 3.5.0 (default, Sep 17 2015, 13:05:18)
[GCC 4.8.4] on linux
Type "help", "copyright", "credits" or "license" for more
information.
>>>
```

This output means you also have Python 3 installed, so you'll be able to use either version. Whenever you see the python command in this book, enter `python3` instead.

Installing a Text Editor

Geany is a simple text editor: it's easy to install, will let you run almost all your programs directly from the editor instead of through a terminal

You can install Geany in one line using the terminal on most Linux systems:

```
$ sudo apt-get install geany
```

If this doesn't work, see the instructions at *http://geany.org/Download/ThirdPartyPackages/*.

Running the Hello World Program

To start your first program, open Geany and save an empty Python file (*File* > *Save As*) called `hello_world.py` in your `python_work` folder. The extension `.py` tells Geany your file will contain a Python program.

After you've saved your file, enter the following line:

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

If multiple versions of Python are installed on your system, you need to make sure Geany is configured to use the correct version. Go to *Build* > *Set Build Commands*. You should see the words `Compile` and `Execute` with a command next to each. Geany assumes the correct command for each is python, but if your system uses the python3 command, you'll need to change this.

If the command python3 worked in a terminal session, change the Compile and Execute commands so Geany will use the Python 3 interpreter. Your Compile command should look like this:

```
python3 -m py_compile "%f"
```

You need to type this command exactly as it's shown. Make sure the spaces and capitalization match what is shown here. Your Execute command should look like this:

```
python3 "%f"
```

Again, make sure the spacing and capitalization match what is shown here. Figure below shows how these commands should look in Geany's configuration menu.

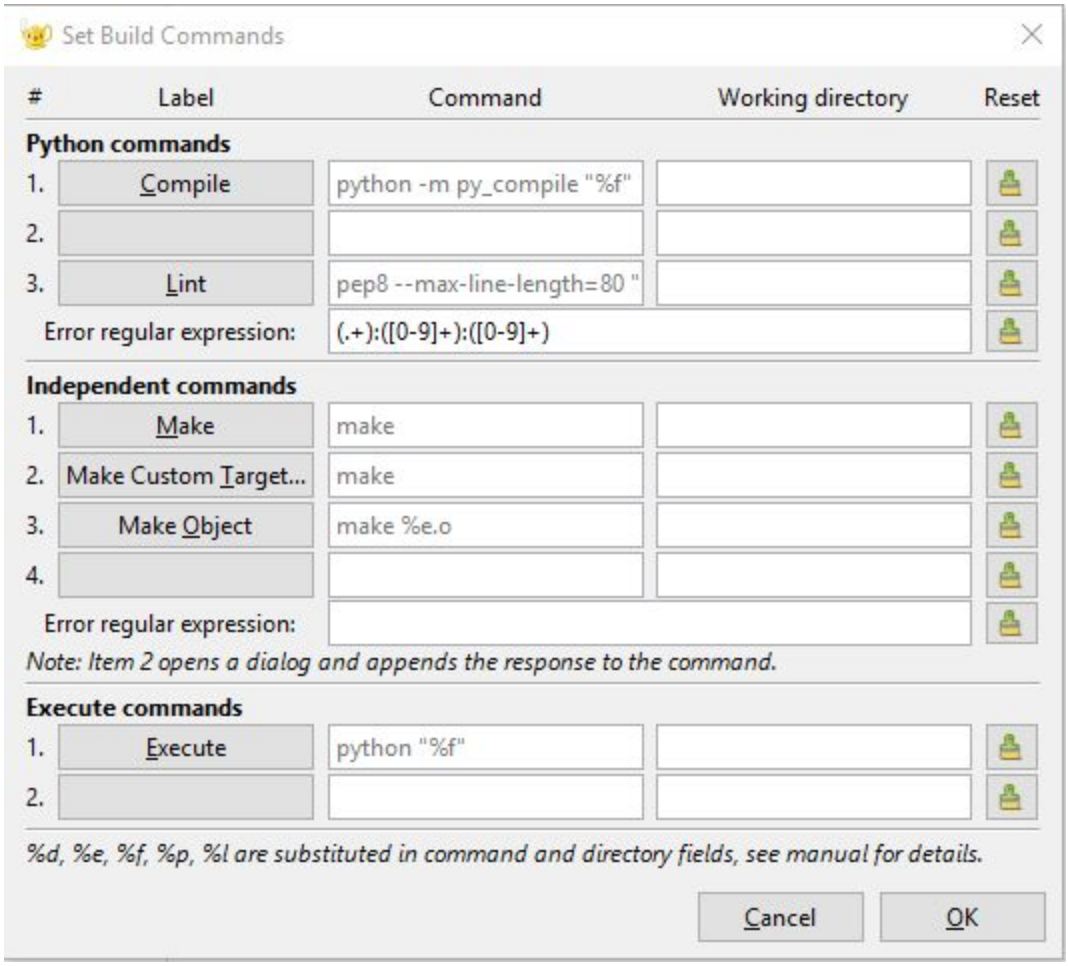

Figure 1-1: Here, Geany is configured to use Python 3 on Linux.

Now run `hello_world.py` by selecting *Build* > *Execute* in the menu, by clicking the Execute icon (which shows a set of gears), or by pressing F5. A terminal window should pop up with the following output:

```
Hello Python world!

------------------
(program exited with code: 0)
Press return to continue.
```

If you don't see this, check every character on the line you entered. Did you accidentally capitalize print? Did you forget one or both of the quotation marks or parentheses? Programming languages expect very specific syntax, and if you don't provide that, you'll get errors. If you can't get the program to run, see "Troubleshooting Installation Issues" discussed later in the document.

Running Python in a Terminal Session

You can try running snippets of Python code by opening a terminal and typing python or python3, as you did when checking your version. Do this again, but this time enter the following line in the terminal session:

```
>>>print("Hello Python interpreter!")
Hello Python interpreter!
>>>
```

You should see your message printed directly in the current terminal window. Remember that you can close the Python interpreter by pressing CTRL-D or by typing the command `exit()`.

Python on OS X

Python is already installed on most OS X systems. Once you know Python is installed, you'll need to install a text editor and make sure it's configured correctly.

Checking Whether Python Is Installed

Open a terminal window by going to *Applications* > *Utilities* > *Terminal*. You can also press COMMAND-SPACEBAR, then type `terminal`, and then press ENTER. To find out whether Python is installed, enter `python` with a lowercase `p`. You should see output telling you which version of Python is installed on your system and a `>>>` prompt where you can start entering Python commands, like this:

```
$ python
Python 2.7.5 (default, Mar 9 2014, 22:15:05)
[GCC 4.2.1 Compatible Apple LLVM 5.0 (clang-500.0.68)] on darwin
Type "help", "copyright", "credits", or "license" for more
information.
>>>
```

This output tells you that Python 2.7.5 is currently the default version installed on this computer. When you've seen this output, press CTRL-D or enter `exit()` to leave the Python prompt and return to a terminal prompt.

To check for Python 3, try the command `python3`. If `python3` works on your system, whenever you see the python command in this book, make sure you use `python3` instead.

Running Python in a Terminal Session

You can try running snippets of Python code by opening a terminal and typing `python` or `python3`. Enter the following line in the terminal session:

```
>>>print("Hello Python interpreter!")
Hello Python interpreter!
```

You can close the Python interpreter by pressing CTRL-D or by typing the command `exit()`.

Installing a Text Editor

Sublime Text is a simple text editor: it's easy to install on OS X, will let you run almost all of your programs directly from the editor instead of through a terminal, uses syntax highlighting to color your code, and runs your code in a terminal session embedded in the Sublime Text window to make it easy to see the output.

You can download an installer for Sublime Text from *http://sublimetext .com/3*. Click the download link and look for an installer for OS X. After the installer has been downloaded, open it and then drag the Sublime Text icon into your `Applications` folder.

Configuring Sublime Text for Python 3

If you use a command other than python to start a Python terminal session, you'll need to configure Sublime Text so it knows where to find the correct version of Python on your system. Issue the following command to find out the full path to your Python interpreter:

```
$ type -a python3
python3 is /usr/local/bin/python3
```

Now open Sublime Text, and go to *Tools* > *Build System* > *New Build System*, which will open a new configuration file for you. Delete what you see and enter the following:

```
{
      "cmd": ["/usr/local/bin/python3", "-u", "$file"],
}
```

This code tells Sublime Text to use your system's python3 command when running the currently open file. Make sure you use the path you found when issuing the command type `-a python3` in the previous step. Save the file as `Python3.sublime-build` in the default directory that Sublime Text opens when you choose Save.

Running the Hello World Program

To start your first program, launch Sublime Text by opening the *Applications* folder and double-clicking the Sublime Text icon. You can also press command-spacebar and enter `sublime text` in the search bar that pops up.

Make a folder called `python_work` somewhere on your system for your projects. (It's best to use lowercase letters and underscores for spaces in file and folder names, because these are Python naming conventions). Save an empty Python file (*File* > *Save As*) called `hello_world.py` in your `python_work` folder. After you've saved your file, enter the following line:

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

If the command python works on your system, you can run your program by selecting **Tools > Build** in the menu or by pressing COMMAND-B.

A terminal screen should appear at the bottom of the Sublime Text window, showing the following output:

```
Hello Python world!
[Finished in 0.1s]
```

If you can't get the program to run, see "Troubleshooting Installation Issues" discussed later in the document.

Python on Windows

Windows doesn't always come with Python, so you'll probably need to download and install it, and then download and install a text editor.

Installing Python

Download a Python installer for Windows. Go to *http:// python.org/downloads/*. You can select one from various available versions to download, which should automatically start downloading the correct installer for your system after the click. After you've downloaded the file, run the installer. Make sure you check the option Add Python to PATH, which will make it easier to configure your system correctly. Figure 1-2 shows this option checked.

Figure 1-2: Make sure you check the box labeled Add Python to PATH.

Starting a Python Terminal Session

Open a command window and enter **python** in lowercase. If you get a Python prompt (>>>), Windows has found the version of Python you just installed:

```
C:\>python
Python 3.5.0 (v3.5.0:374f501f4567, Sep 13 2015, 22:15:05) [MSC v.1900
32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license" for more
information.
>>>
```

If this worked, you can move on to the next section, "Running Python in a Terminal Session."
However, you may see output that looks more like this:

```
C:\>python
'python' is not recognized as an internal or external command,
operable program or batch file.
```

In this case, you need to tell Windows how to find the Python version you just installed. Your system's python command is usually saved in your *C:* drive, so open Windows Explorer and open your *C:* drive. Look for a folder starting with the name `Python`, open that folder, and find the `python` file (in lowercase). For example, I have a `Python35` folder with a file named `python` inside it, so the path to the python command on my system is *C:\Python35\python*. Otherwise, enter `python` into the search box in Windows Explorer to show you exactly where the python command is stored on your system.
When you think you know the path, test it by entering that path into a terminal window. Open a command window and enter the full path you just found:

```
C:\>C:\Python35\python
Python 3.5.0 (v3.5.0:374f501f4567, Sep 13 2015, 22:15:05) [MSC v.1900
32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license" for more
information.
>>>
```

If this worked, you know how to access Python on your system.

Running Python in a terminal Session
Enter the following line in your Python session, and make sure you see the output:

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

Any time you want to run a snippet of Python code, open a command window and start a Python terminal session. To close the terminal session, press CTRL-Z and then press ENTER, or enter the command `exit()`.

Installing a Text Editor

Geany is a simple text editor: it's easy to install, will let you run almost all of your programs directly from the editor instead of through a terminal, uses syntax highlighting to color your code, and runs your code in a terminal window so you'll get used to using terminals.

You can download a Windows installer for Geany from *http://geany.org/*. Click *Releases* under the Download menu, and look for the ***geany-1.25_setup.exe*** installer or something similar. Run the installer and accept all the defaults.

To start your first program, open Geany. Make a folder called `python_work` somewhere on your system for your projects. (It's best to use lowercase letters and underscores for spaces in file and folder names, because these are Python naming conventions.) Go back to Geany and save an empty Python file (*File* **>** *Save As*) called `hello_world.py` in your `python_work` folder. The extension `.py` tells Geany that your file will contain a Python program. It also tells Geany how to run your program and to highlight the text in a helpful way.

After you've saved your file, type the following line:

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

If the command python worked on your system, you won't have to configure Geany; skip the next section and move on to "Running the Hello World Program" on page 14. If you needed to enter a path like `C:\Python35\python` to start a Python interpreter, follow the directions in the next section to configure Geany for your system.

Running the Hello World Program

Run `hello_world.py` by selecting **Build > Execute** in the menu, by clicking the Execute icon (which shows a set of gears), or by pressing F5. A terminal window should pop up with the following output:

```
Hello Python world!
------------------
(program exited with code: 0)
Press return to continue
```

If you don't see this, check every character on the line you entered. Did you accidentally capitalize print? Did you forget one or both of the quotation marks or parentheses? Programming languages expect very specific syntax, and if you don't provide that, you'll get errors. If you can't get the program to run, see the next section for help.

Till above step you have successfully installed Python to your system if it wasn't already there, and a text editor named Geany. You can also have other text editor of your choice, but at start choosing a

simple editor is a good idea and then you can move on to a more complex one as your experience with the language increases.

Troubleshooting Installation Issues

If you've been unable to run `hello_world.py`, here are a few remedies you can try:

1. When a program contains a significant error, Python displays a `traceback`. Python looks through the file and tries to report the problem. The traceback might give you a clue as to what issue is preventing the program from running.

2. Take a step away from your computer, take a short break, and then try again. Remember that syntax is very important in programming, so even a missing colon, a mismatched quotation mark, or mismatched parentheses can prevent a program from running properly.

3. Start over again. You probably don't need to uninstall anything, but it might make sense to delete your `hello_world.py` file and create it again from scratch.

Running Python Programs from a Terminal

Most of the programs you write in your text editor you'll run directly from the editor, but sometimes it's useful to run programs from a terminal instead. For example, you might want to run an existing program without opening it for editing.

You can do this on any system with Python installed if you know how to access the directory where you've stored your program file. To try this, make sure you've saved the `hello_world.py` file in the `python_work` folder on your desktop.

On Linux and OS X

Running a Python program from a terminal session is the same on Linux and OS X. The terminal command `cd`, for *change directory*, is used to navigate through your file system in a terminal session. The command `ls`, for *list*, shows you all the non-hidden files that exist in the current directory.

Open a new terminal window and issue the following commands to run `hello_world.py`:

```
1 ~$ cd Desktop/python_work/
2 ~/Desktop/python_work$ ls
hello_world.py
3 ~/Desktop/python_work$ python hello_world.py
Hello Python world!
```

At 1, we use the `cd` command to navigate to the `python_work` folder, which is in the `Desktop` folder. Next, we use the `ls` command to make sure `hello_world.py` is in this folder 2. Then, we run the file using the command python hello_world.py 3.

It's that simple. You just use the python (or python3) command to run Python programs.

On Windows

The terminal command `cd`, for *change directory*, is used to navigate through your file system in a command window. The command `dir`, for *directory*, shows you all the files that exist in the current directory.
Open a new terminal window and issue the following commands to run `hello_world.py`:

```
1 C:\> cd Desktop\python_work
2 C:\Desktop\python_work> dir
hello_world.py
3 C:\Desktop\python_work> python hello_world.py
Hello Python world!
```

At 1, we use the cd command to navigate to the `python_work` folder, which is in the *Desktop* folder. Next, we use the dir command to make sure `hello_world.py` is in this folder 2. Then, we run the file using the command `python hello_world.py` 3.
If you haven't configured your system to use the simple command python, you may need to use the longer version of this command:

```
C:\$ cd Desktop\python_work
C:\Desktop\python_work$ dir
hello_world.py
C:\Desktop\python_work$ C:\Python35\python hello_world.py
Hello Python world!
```

Most of your programs will run fine directly from your editor, but as your work becomes more complex, you might write programs that you'll need to run from a terminal.

In couple of steps above you have executed a snippet of Python code in a terminal session which says Hello python world!.

Exercises

1. **Do the following.**

   Go to terminal, start python, and do:

   1. Print your name and address in two different lines.
   2. Print Hello and World in two different lines.

   Open geany editor and do:

   3. Write a program to print following:

      ```
      Hello,
      I am <name>,
      From <place>,
      Working for <employer>
      ```

   4. Can you print the above lines with only one print statement?

2. **Practice the following operations.**

   Go to terminal, start python, and do:

   1. Addition 5 + 5
   2. Subtraction 5 - 5
   3. Multiplication 3 * 5
   4. Division (5 + 5) / 2
   5. Exponentiation 2 ^ 5
   6. Modulo 28 %% 6
   7. Print the value of pi.
   8. Find the area of circle with radius 3 cm.

   Open geany editor and do:

   1. Write a program which calculates the area of a circle with radius 3.
   2. Write a program which calculates the volume of a cylinder with base radius 3 cm and height 10 cm.

3. **Write the following lines of code. You will get an error. Try and understand what the error is and correct it.**

   Go to the terminal, start python, and do:

   1. Print(“hello world”)
   2. print(“hello world’)
   3. print “hello world”
   4. Divide 0 by 0