

INSTALLING THIRD-PARTY MODULES IN PYTHON

WHAT ARE MODULES?

Modules are basically files which contain definitions that provide helpful functionality. Theoretically, all of you could write any program you wanted to write on your own, without any modules or outside help. But why reinvent the wheel if you don't have to? This is the general gist behind modules. You all have experience using modules such as `tkinter` and `csv`. These modules are part of a library of standard modules that came packaged with your install which the developers have deemed useful or necessary for your daily python usage. However, they can't give you everything. This is where third-party modules come in. When you come across a problem that the standard modules can't solve, there's a good chance somebody else had the same problem before you and created his own module to solve it. The good news? We can take that module (if the developer is kind enough to publish it) and install it for our own use. Cool, right? The bad news, however, is that it's a little more complicated than your average one-click install...

More reading: <http://docs.python.org/3/tutorial/modules.html>

HOW TO INSTALL MODULES

Before we get into the details of actually installing `pymysql`, it'll be good to learn a little more about installing packages to Python in general. This will be useful to know if you ever start using Python more seriously as there are many packages out there which provide you with some functionality you couldn't otherwise get. In this class, we will be working with `pymysql`, a library written purely in Python, to interact with MySQL database.

The traditional way to install `pymysql` in this class has been to download the source and run its `setup.py` script to install the package on our system. However, in the interest of giving you the know-how to install other third party libraries should you need them; we will be looking at a more versatile method using the `setuptools` module. You will also need to know the very basics of using the command line.

THE OS X TERMINAL

1. Open up the terminal. You should be able to find the terminal in Applications > Utilities. Alternatively, you can type "terminal" into the Spotlight search bar. You should end up with a prompt something like this (yours will probably look a little different in terms of cosmetics, but the important details are our prompt and the command line. on the left of the dollar sign, you'll see your prompt, which contains usually the name of the owner of the computer and the current working directory. Yours could be in a different order than mine):

```
Terminal - michael@march:~
File Edit View Terminal Tabs Help
michael ~ $
```

2. Navigating your file system through the command line is going to feel different from what you're used to. If you look at your prompt, you should see your present working directory (for Mac/Unix systems you'll probably see some variation of the name of your computer along with the present working directory which will probably be '~' a shortcut for 'home directory'). This is where you currently are in your filesystem. You can use the command "ls" shown below to see the contents of this directory. Input commands by typing them into the prompt and hitting the return key.

```
Terminal - michael@march:~
File Edit View Terminal Tabs Help
michael ~ $
michael ~ $ ls
builds coursera Desktop Downloads Images School setup
code cs2316 Documents gba R sch_readings workspace
michael ~ $
```

3. We won't be going into great detail with the command line in this class, but we will need to know how to change directories and execute scripts (we'll talk about this in a bit) to install pymysql. To change our directory (or folder), type the cd command into the command line followed by the path to the directory you wish to change to. If we downloaded a file to our Downloads folder in ~/Downloads for example, we could type 'cd ~/Downloads' into the command line to change to the Downloads directory. (Alternatively, we could just type in Downloads, and since the Downloads folder is in our current directory, we change to it by relative path):

```
Terminal - michael@march:~/Downloads
File Edit View Terminal Tabs Help
michael ~ $
michael ~ $ ls
builds coursera Desktop Downloads Images School setup
code cs2316 Documents gba R sch_readings workspace
michael ~ $ cd Downloads
michael ~/Downloads $
```

STANDARD INSTALL (INSTALLING SETUPTOOLS)

1. First, download and extract the setuptools zip file located on the class webpage.
2. Now, use the cd command to change the directory to the folder you just unzipped.

```
Last login: Tue Mar  4 12:21:54 on ttys000
lawn-128-61-115-184:~ Insomnio$ cd /Users/Insomnio/Python/Packages/setuptools-2.2
```

3. Now you can run the setup.py file using following command “sudo python3.3 setup.py install”:

```
lawn-128-61-115-184:setuptools-2.2 Insomnio$ sudo python3.3 setup.py install
Password:
running install
running bdist_egg
running egg_info
writing top-level names to setuptools.egg-info/top_level.txt
writing requirements to setuptools.egg-info/requirements.txt
```

Using the “sudo” command grants you administrator privileges, which are necessary to do the install. You will be asked for your password after you hit return. Type your password and hit return. You will have to type your password blind, meaning you will not be able to see it, or any markers for the keys you have pressed.

4. You will see a lot of text scrolling down your screen. Wait for it to stop and then type “python3.3 -m easy_install --version” into your command line.

```
Last login: Tue Mar  4 12:19:45 on ttys000
lawn-128-61-115-184:~ Insomnio$ python3.3 -m easy_install --version
setuptools 2.2
lawn-128-61-115-184:~ Insomnio$
```

If it displays something similar to above, you’ve installed setuptools correctly. If you're still having problems, contact your nearest TA.

INSTALLING USING EASY_INSTALL

That took a lot of work, didn’t it? First you had to download the module’s source distribution and extract it, then change the directory in the command prompt to the directory you extracted it to, and, only after that was done, could you run the installation command. Now that you have setuptools installed you will be able to download any third party library in the Python Packaging Index: <https://pypi.python.org/pypi> and install it with one simple command.

Before you do anything else, please close and reopen the terminal. In order to install pymysql, just run following command “sudo python3.3 -m easy_install pymysql” and enter your password.

```
Last login: Tue Mar  4 12:28:09 on ttys000
lawn-128-61-115-184:~ Insomnio$ python3.3 -m easy_install pymysql
Searching for pymysql
Best match: PyMySQL 0.6.1
Processing PyMySQL-0.6.1-py3.3.egg
PyMySQL 0.6.1 is already the active version in easy-install.pth

Using /Library/Frameworks/Python.framework/Versions/3.3/lib/python3.3/site-packages/PyMySQL-0.6.1-py3.3.egg
Processing dependencies for pymysql
Finished processing dependencies for pymysql
lawn-128-61-115-184:~ Insomnio$
```

That's it! If everything worked properly, you should be able to pull up an IDLE shell (Python window) and type in "import pymysql" without error statements. If it says pymysql could not be found re-run the command. If the issue persists or you're having any other trouble, ask the nearest TA.

More Info on Easy Install: http://pythonhosted.org//setuptools/easy_install.html