

Using Python From Your Laptop: A Handy Guide

Written by Ciaradh Coomey

.....

If you want to download Python to your own computer:

1. Using a web browser, navigate to <https://www.python.org/downloads/>.
2. Select the yellow “Download Python 3.7.4” button (or whatever the current version is).
3. Once the download completes, click on the freshly downloaded [python-3.7.4.exe](#) file.
4. Important: if you intend to use the command line to run your Python programs, make sure to select the checkbox “Add Python 3.7 to PATH.”
5. Click “Install Now.”
6. Select “Yes” when asked whether you want to allow the app to make changes to your device.
7. Be patient as the installation completes. Make a cup of coffee/tea, start reading the next chapter in your book, watch a YouTube video, have a quick snack, etc.
8. Success! Continue on to the option of your choice below.

.....

If you want to write and execute Python code from your own computer, and you liked using the terminal in Lab 1:

If you haven’t done so already, download Python (see above). Just as in lab, you should be able to write and save .py files with the text editor of your choice (if you liked Atom, it is available for download here: <https://atom.io/>). The default text editor for Windows is Notepad and the default text editor for OSX is TextEdit, but there are lots of text editor options that you can download for free online <https://lifehacker.com/five-best-text-editors-1564907215>.

If you have a Mac:

1. Open the Terminal.
2. To navigate to the correct folder to run your program, you can use essentially the same commands as given in Lab 1 (`cd`, `ls`, `pwd`, `mkdir`, `mv`, `cp`, etc.).
3. To run a Python program, you can enter `python3 fileName.py`, just as in Lab 1.

If you are using Windows, and you don't mind learning new commands that are different from those learned in Lab 1:

- Open Windows Command Prompt.
- To change directories, use `cd DirectoryName` where `DirectoryName` is the name of your desired directory. Remember, this only works if you are currently in a directory that contains your desired directory. Other `cd` commands such as `cd ..` for moving up a directory and `cd ~` for returning to the home directory remain the same for the Windows command prompt.
- Another important command to be aware of is `dir`, which takes the place of Linux command `ls`.
- To run the Python interpreter in the Windows Command Line, you must use the command `py` instead of `python3`. As in Linux, you can keep the python interpreter open by adding `-i` after `py` when running a saved script. For example, if you are in a directory named `lab2` that contains a Python script called `drawsquare.py`, you can run the Python script with the line `py drawsquare.py`. To keep the Python interpreter open after running the `drawsquare.py` file, you can use `py -i drawsquare.py`.
- More information can be found here:
<http://www.cs.columbia.edu/~sedwards/classes/2015/1102-fall/Command%20Prompt%20Cheatsheet.pdf>

If you are using Windows, but want to use the same commands that you learned in Lab 1:

Git Bash is a command-line emulator that allows you to type Linux-style commands on your Windows computer. Besides the basic `ls`, `cd`, `mkdir`, and similar commands that we learned in Lab 1, Git Bash can also be used for SSH (see below) and to push files to GitHub (talk to a TA if you are interested in learning more about this).

1. First, download Git Bash from the following link, <https://git-scm.com/downloads>, selecting the “Windows” option.
2. Once the download is complete, you can use Git Bash to navigate through the files on your PC and run your Python program from the command line just as you would a terminal on one of the CSE Labs Machines (as in Lab 1).

.....

If you want to write and execute Python code from your own computer, but you'd rather not use the terminal (recommended for beginners):

If you haven't done so already, download Python (see above).

IDLE is a popular IDE (Integrated Development Environment) for Python development that is automatically installed with Python. To use IDLE, simply type "IDLE" in your computer's search bar (at the lower left of the screen in Windows 10). You can also choose to pin the application to your start menu or save it to your desktop.

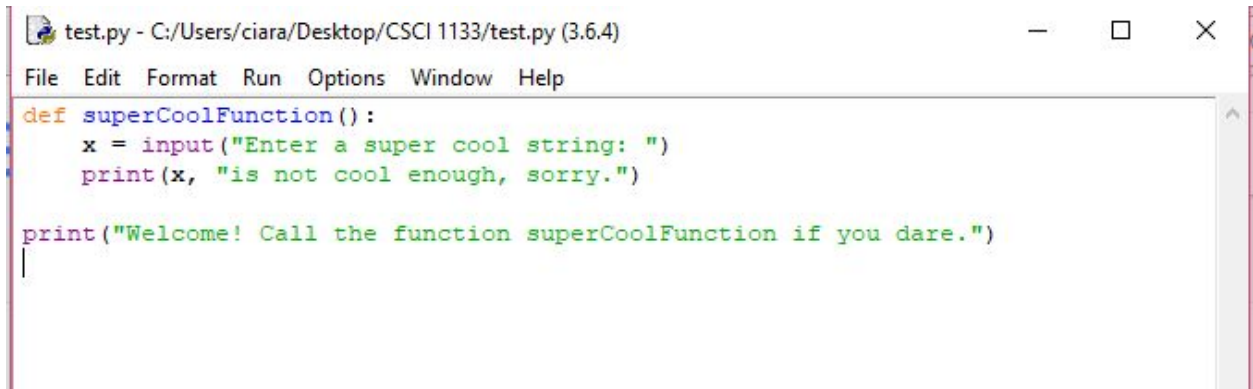
IDLE has two main components: a shell window for testing and running your code, and a code editor for writing and saving your programs. When you first open IDLE, the Python shell will appear.



```
Python 3.6.4 Shell
File Edit Shell Debug Options Window Help
Python 3.6.4 (v3.6.4:d48eceb, Dec 19 2017, 06:04:45) [MSC v.1900 32 bit (Intel)]
on win32
Type "copyright", "credits" or "license()" for more information.
>>> 3 + 4
7
>>> |
```

Just as in the command-line shell we used in Lab 1, you can use the shell in IDLE to evaluate expressions and call functions.

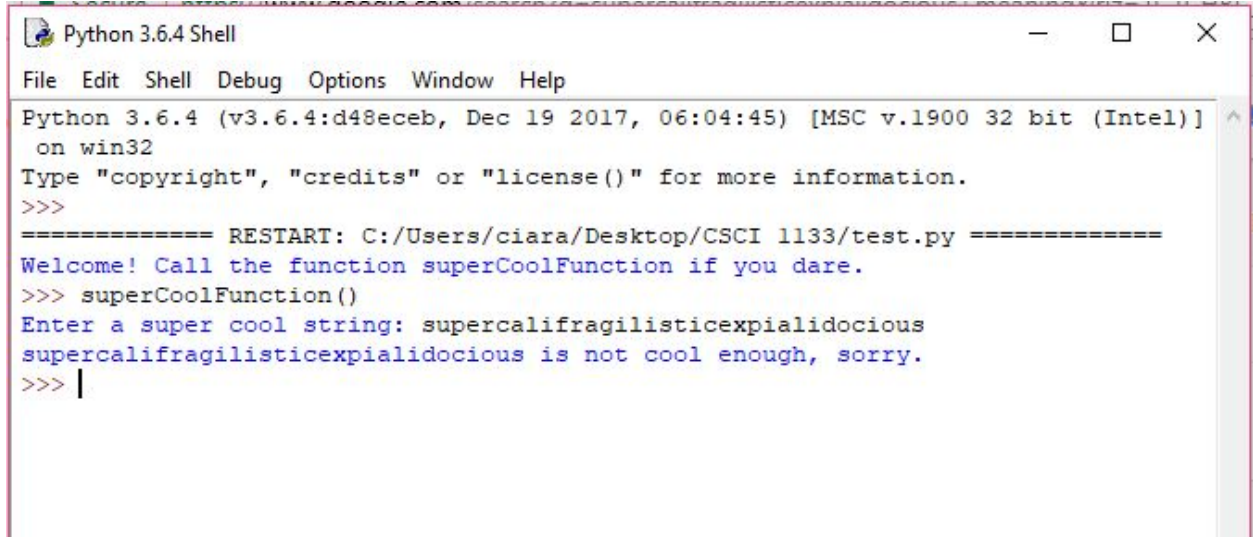
To open a new file, select File > New File from the upper left of the shell window. You can then write and save your Python code, just as you did in Atom or another text editor in lab. The .py files that you create with IDLE are equivalent to those created with a simple text editor, and can be submitted for your homework.



```
test.py - C:/Users/ciara/Desktop/CSCI 1133/test.py (3.6.4)
File Edit Format Run Options Window Help
def superCoolFunction():
    x = input("Enter a super cool string: ")
    print(x, "is not cool enough, sorry.")

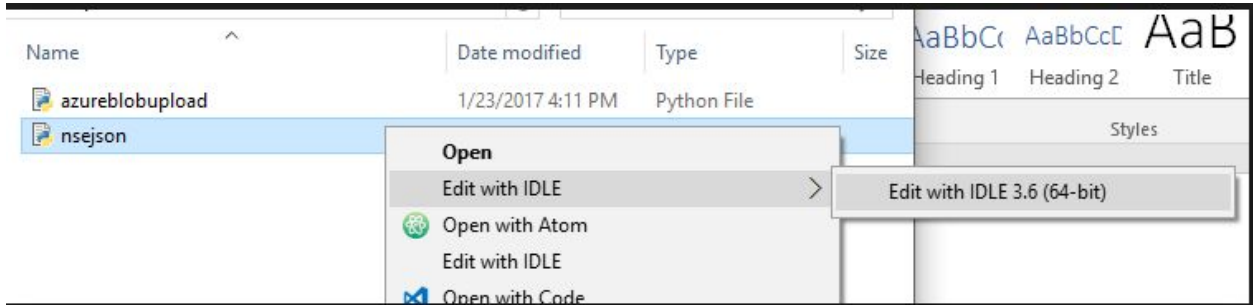
print("Welcome! Call the function superCoolFunction if you dare.")
|
```

To run your program, simply select Run > Run Module from the upper left of the shell window (or hit F5). You will then see your output in the shell window, and will be able to run functions from your file:



```
Python 3.6.4 Shell
File Edit Shell Debug Options Window Help
Python 3.6.4 (v3.6.4:d48eceb, Dec 19 2017, 06:04:45) [MSC v.1900 32 bit (Intel)]
on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/ciara/Desktop/CSCI 1133/test.py =====
Welcome! Call the function superCoolFunction if you dare.
>>> superCoolFunction()
Enter a super cool string: supercalifragilisticexpialidocious
supercalifragilisticexpialidocious is not cool enough, sorry.
>>> |
```

You can also open .py files that you prepared earlier by selecting File > Open from the upper right of the code editor, or by right-clicking directly on the file and selecting “Edit with IDLE.”

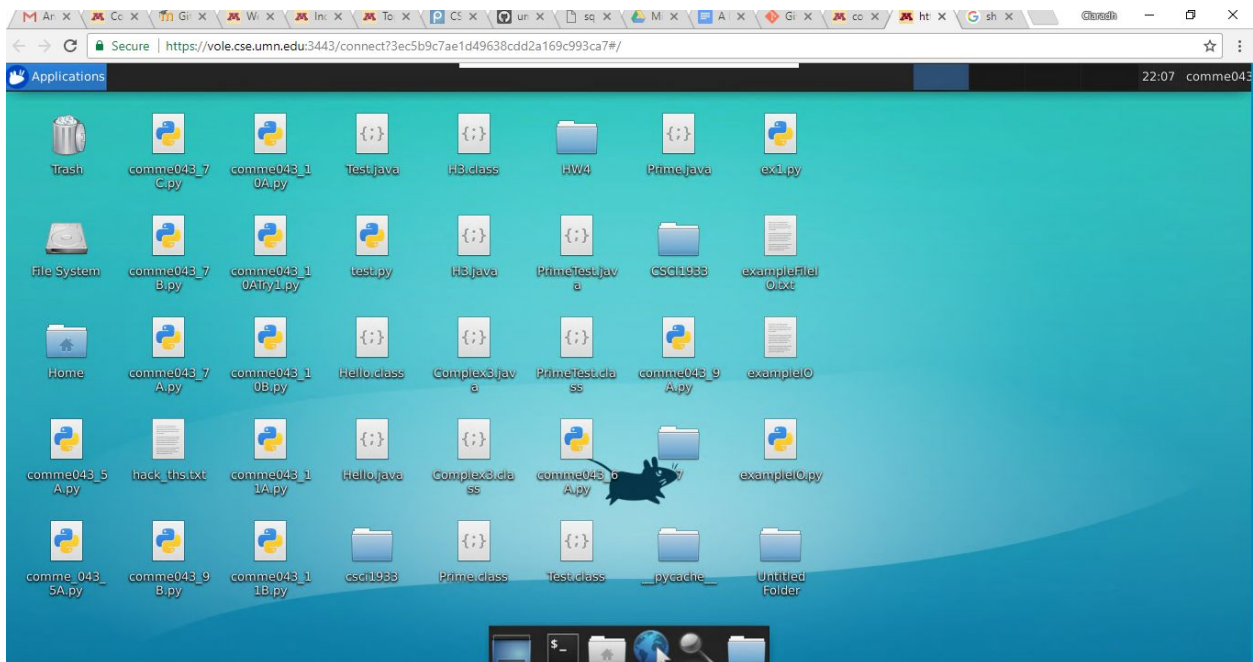


.....

If you want to connect to your lab account from your own computer, and want to see a graphical display of your lab desktop:

VOLE provides a way to visually connect to your CSE Labs Machines account. To use VOLE:

1. Go to <http://vole.cse.umn.edu/>
2. Click the “Connect Now!” button.
3. Log in with your UMN account.
4. If a “high usage” warning appears, hit the “Close” button.
5. Select the “+” symbol near the upper left (it may read “Launch Session” instead).
6. Click the image of the mouse (Vole? Rat? Unidentifiable rodent?) and select “Start.”
7. Though VOLE can be a little slow, especially at peak use times, you will now be able to view a virtual desktop that functions just as one of the lab computers. Any changes you make via VOLE will also appear when you log onto your account with SSH or in lab.



.....
If you want to connect to your lab account from your own computer, and are okay with only using the terminal (typing commands without clicking buttons with your mouse or seeing other graphics, such as turtle drawings):

SSH is another way to remotely connect to the CSE Labs Machines from your own laptop. Though SSH only allows you to interact with programs and files via the command line, SSH is significantly faster than VOLE. To link to SSH:

1. Open Git Bash or another terminal emulator (see above), and type

```
ssh comme043@cse-kh4250-16.cselabs.umn.edu
```

where “comme043” is replaced with your UMN username, and “4250-16” with the computer of your desire (for example, kh4250-18). If the computer you select is busy, you may receive a message that indicates you are unable to connect. In this case, simply replace the computer number with a different choice.

2. The first time you connect to a given computer, you may see the following message:

```
The authenticity of host 'cse-kh4250-18.cselabs.umn.edu  
(128.101.37.18)' can't be established.  
RSA key fingerprint is  
SHA256:dr/o2s0vGyJAhI6eckmthBvBQMknJx5FCTQEpUnqCyQ.  
Are you sure you want to continue connecting (yes/no)?
```

Simply type “yes” to continue connecting.

3. Each time you connect to one of the CSE Labs Machines via SSH, you will also be prompted for your UMN password.
4. Once the connection process is complete, you will be able to execute Linux terminal commands exactly as on the lab computers. For example, you can remotely run the python shell, or even open or edit files that you have saved on your CSE Labs Account. Any changes you make via SSH will also appear when you log onto your account with VOLE or in lab.

A few screenshots of an example SSH session:

```
$ ssh comme043@cse1-kh4250-16.cselabs.umn.edu
comme043@cse1-kh4250-16.cselabs.umn.edu's password:
-----
      COLLEGE OF SCIENCE AND ENGINEERING WORKSTATION
If you are not authorized to access this system, disconnect now.

      YOU SHOULD HAVE NO EXPECTATION OF PRIVACY

By continuing, you agree to the terms outlined in the Acceptable Use
Policy (AUP) governing use of this workstation.  The AUP may be found
online at

      https://cseit.umn.edu/knowledge-help/acceptable-use-policy

As a user of this system, it is YOUR responsibility to be familiar
with the information contained in the AUP.

Users requiring assistance should talk to the operator on duty:
They can be reached by any of the following means:
      email: operator@cselabs.umn.edu
      or call 612-625-0876  The phones in the labs directly call operator.
      You may also visit them in Keller Hall 1-201

-----

Message from the CSE Labs operator:
=====

      Due to high usage, all computers in Keller Hall
4-250 will automatically kill processes that have been
running for longer than 3 days.

Please email operator@cselabs.umn.edu with any concerns.

=====

Last login: Wed Dec 13 11:24:46 2017 from 10.139.22.220
comme043@cse1-kh4250-16:/home/comme043 $ python3
Python 3.5.2 (default, Nov 23 2017, 16:37:01)
[GCC 5.4.0 20160609] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> 3 + 4
7
>>> |
```

```

comme043@cse1-kh4250-16:/home/comme043 $ ls
1133Exam3/      Music/          comme043_3A.py  hack_ths.txt  lab3.6         lab7.2
2.5_day.csv    Pictures/       comme043_3B.2.py hello/         lab3.py        lab7.3
34534.py        Public/         comme043_3B.py  hello-world/  lab4.1.py      lab7.4
<comme043>1A.py README.dotfiles comme043_4A.py  higg.py       lab4.2.py      lab7.5
<comme043>_1A.py Templates/     comme043_4B.2.py higgins.py    lab4.3.py      lab8.py
CSCI/          Try2comme043_2A.py comme043_4B.py  hydrogen/     lab4.4.py      lab81.txt
CSCI 1133 Exam 2/ Untitled Document 1 comme043_4BTry1.py jfgd         lab4.5         lab0ne/
CSCI4041/      Videos/        comme043_4BTry2.py lab1/         lab5.1.py      myscripts/
Desktop/       __pycache__/   comme043_5B.py  lab11.py      lab5.2         nano decay.py
Documents/     ads/            computetax.py  lab13.1       lab5.3         nested_list.py
Downloads/     cDc-0200.txt   csci2041/      lab13a.py     lab5.4         show.sh*
Lab 11.1.py    challenge.py    dfd            lab3.1        lab6.1         simpletext.txt
Lab 11.py      comme043_1A.py exam-grieb032/ lab3.2        lab6.2         table.csv
Lab 12_1.py    comme043_2A.py exam-tamma008/ lab3.3        lab6.3         test.py
Lab 12_3.py    comme043_2B.py exam-wickx182/ lab3.4        lab6.py        www/
Lab12_2.py     comme043_3A.2.py f_to_c.py      lab3.5        lab7.1
comme043@cse1-kh4250-16:/home/comme043 $ python3 comme043_1A.py
Hello, Welcome to the Fahrenheit to Celsius Temperature Conversion Program

Please enter the temperature in Fahrenheit: 300
The temperature in Celsius is: 148.88888888888889
comme043@cse1-kh4250-16:/home/comme043 $

```

```

comme043@cse1-kh4250-16:/home/comme043 $ nano comme043_1A.py

```

```

# CSci 1133 HW 1
# Ciaradh Coomey
# HW Problem 1A
#Here is a python program to convert the Fahrenheit temperature to
# the temperature in Celsius

print('Hello, Welcome to the Fahrenheit to Celsius Temperature Conversion Program\n')
Tf = float(input('Please enter the temperature in Fahrenheit: '))
Tc = (Tf - 32) * 5/9
print ('The temperature in Celsius is: ', Tc)

```

[Read 10 lines]

⌘ Get Help	⌘ Write Out	⌘ Where Is	⌘ Cut Text	⌘ Justify	⌘ Cur Pos	⌘ Prev Page
⌘ Exit	⌘ Read File	⌘ Replace	⌘ Uncut Text	⌘ To Linter	⌘ Go To Line	⌘ Next Page