

# CS001X Introduction to Computing

## Lab 1

### Overview

In this lab, we will install the software we will need to write and run Python programs, gain familiarity with basic command line tools, and write and run our first Python program.

This lab provides detailed instructions for Windows 10 and MacOS 10.9 or higher.

### 1 Activity: Install Python

If you have an Apple computer, complete Section 1.1. If your operating system is Windows, complete Section 1.2.

#### 1.1 macOS

**Download Python installer** Use the following link to download the Python installer:

<https://www.python.org/ftp/python/3.7.4/python-3.7.4-macosx10.9.pkg>

**Install Python** Double click on the installer to run it (should be called `python-3.7.4-macosx10.9.pkg`). Follow the directions on the installer screens to install Python.

- On the screen titled “Welcome to the Python Installer”, click “Continue”.
- On the next screen, titled “Important Information”, click “Continue”.
- On the next screen, titled “Software License Agreement”, click “Continue” and then “Agree” on the window that pops up.
- Then select the hard drive onto which you wish to install Python. In most cases, you should only have one option. Then click “Continue”.
- Finally, you will then see a screen with title “Standard Install on Macintosh HD”, click “Install”. There is no need to customize the installation or change the install location.

## 1.2 Windows

**Check Windows version** First, check whether you have 32-bit or 64-bit Windows installed. Do this by going to the “Start” menu in the bottom left, then click “Settings”, then “System”. In the menu on the left side of the window, click “About”. Whether the operating system is 32 or 64-bit will be shown under the heading “Device specifications”. See Figure 1.2 for an example.

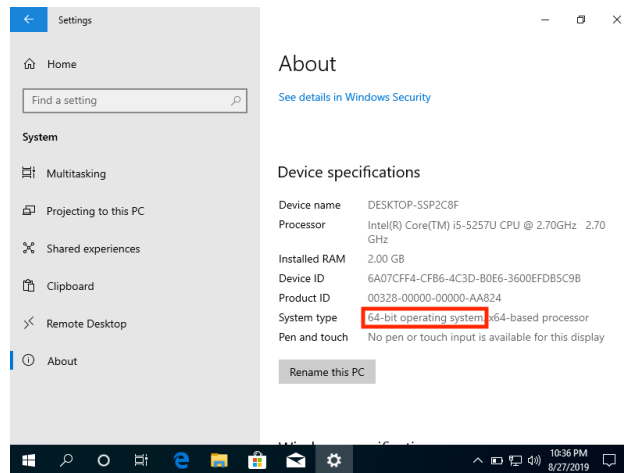


Figure 1: Displaying the Windows version

**Download Python installer** If you have a 32-bit version of Windows installed, use the following link to download the Python installer:

<https://www.python.org/ftp/python/3.7.4/python-3.7.4.exe>

If you have the 64-bit version of Windows, use this link:

<https://www.python.org/ftp/python/3.7.4/python-3.7.4-amd64.exe>

**Install Python** Next, double click on the downloaded installer file (should be `python-3.7.4.exe` if 32-bit or `python-3.7.4-amd64.exe` if 64-bit). You will then see a screen like the one in Figure 1.2.

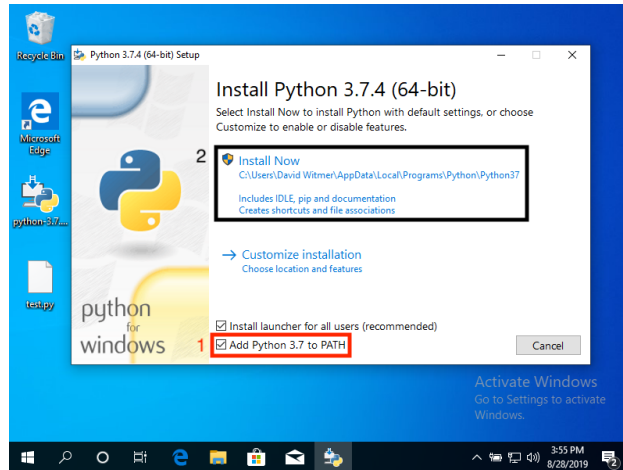


Figure 2: Windows installation screen

**IMPORTANT** On this screen, make sure you check the box “Add Python 3.7 to PATH” marked in red in Figure 1.2. AFTER YOU HAVE CHECKED THIS BOX, complete the installation by clicking on the “Install Now” option marked in black in Figure 1.2. This will be important both for running programs from the command line (see below for more on this) and installing additional Python packages (later in the course).

### 1.3 Other operating systems

To install Python for other operating systems, use a package manager (e.g., `apt` for Debian or Ubuntu, `yum` for Red Hat or Fedora, `dnf` for Fedora).

If you have no package manager or wish to compile from source, you can download source files from here:

<https://www.python.org/downloads/release/python-374/>

## 2 Activity: Install Atom

### 2.1 macOS

You can download Atom using this link:

<https://github.com/atom/atom/releases/download/v1.40.1/atom-mac.zip>

The downloaded file `atom-mac.zip` will either be automatically unzipped or you can double click on it to unzip. Inside, will be a file called `Atom` or `Atom.app`. You may move this file to wherever you would like to put it (e.g., your Applications folder). Then double click on it to open it.

### 2.2 Windows

**Download Atom installer** If you a 32-bit version of Windows, download the Atom installer using this link:

<https://github.com/atom/atom/releases/download/v1.40.1/AtomSetup.exe>

If you have a 64-bit version, use this link to download the installer:

<https://github.com/atom/atom/releases/download/v1.40.1/AtomSetup-x64.exe>

Double click on the downloaded installer file (`AtomSetup.exe` for 32-bit or `AtomSetup-x64.exe` for 64-bit). Atom will be installed and then open automatically.

## 2.3 Other operating systems

Atom may be available on package management systems. Also, see the webpage below for additional installation options:

<https://github.com/atom/atom/releases/tag/v1.40.1>

## 3 Activity: Write a first program

Open Atom. Click the “File” menu in the upper left corner. Then select the “New File” option. You can close the “Welcome” and “Telemetry Consent” tabs.

In this file, write a program that uses the `print` function described in lecture to print “Hello, world!”.

Make a new folder with the name `CS001X` . Save the file as `lab1.py`.

## 4 Activity: Use the command line

### 4.1 Paths

Paths specify locations of files and directories in the file system.

**macOS** In macOS, paths look like this:

```
/mydirectory/mysubdirectory/mysubsubdirectory/myfile
```

Directory and filenames are separated using `/`.

**Windows** In Windows, paths look like this:

```
C:\mydirectory\mysubdirectory\mysubsubdirectory\myfile
```

The drive is named at the front of the path and directory and filenames are separated using `\`.

**Absolute and relative paths** A filesystems has a root directory containing all other directories and files. Paths like the above give the location of a file based on the root directory and are called *absolute* paths. On the other hand, *relative paths* give the location of a file from some other directory. In the above example, if we are in the directory named `mysubdirectory`, the relative paths

`mysubsubdirectory/myfile` (macOS)

or

`mysubsubdirectory\myfile` (Windows)

give the location of the file with name `myfile`.

## 4.2 Open the shell

**Windows** Open the Windows Command Prompt. You can do this by typing “command prompt” in the search bar at the bottom of the desktop screen and clicking on the resulting program.

**macOS** Open the Terminal. You can do this by using Spotlight to search for “terminal” and opening the program that comes up.

## 4.3 Navigate through the file system

When working on the command line, you always have a current or working directory. Think about this as the directory in the file system that you are currently looking at.

**macOS** In macOS, you can display the path to the current directory using the `pwd` command. Try this now. Your terminal should look something like this:

```
dwitmer-macbook-pro:~ dwitmer$ pwd
/Users/dwitmer
```

From here, you can use the `cd` command to navigate to the directory containing your program. The `cd` command takes a path and changes the current directory accordingly.

```
dwitmer-macbook-pro:~ dwitmer$ cd Documents
dwitmer-macbook-pro:Documents dwitmer$ pwd
/Users/dwitmer/Documents
dwitmer-macbook-pro:Documents dwitmer$ cd CS001X
dwitmer-macbook-pro:CS001X dwitmer$ pwd
/Users/dwitmer/Documents/CS001X
```

You could also have done this in one step using the entire relative path as follows.

```
dwitmer-macbook-pro:~ dwitmer$ cd Documents/CS001X
```

```
dwitmer-macbook-pro:CS001X dwitmer$ pwd
/Users/dwitmer/Documents/CS001X
```

**Windows** In Windows, you can print the path to the current directory using the `cd` command.

```
C:\Users\dwitmer> cd
C:\Users\dwitmer
```

From here, you can use the `chdir` command to navigate to the directory containing your program. The `chdir` command takes a path and changes the current directory accordingly.

```
C:\Users\dwitmer> chdir Documents
C:\Users\dwitmer\Documents> chdir CS001X
C:\Users\dwitmer\Documents\CS001X>
```

Note that Command Prompt prints the full absolute path. You could also have done this in one step using the entire relative path as follows.

```
C:\Users\dwitmer> chdir Documents\CS001X
C:\Users\dwitmer\Documents\CS001X>
```

## 4.4 List directory contents

**macOS** Now, use the `ls` command to list the contents of the current directory:

```
dwitmer-macbook-pro:CS001X dwitmer$ ls
```

**Windows** Use the `dir` command to list the contents of the current directory:

```
C:\Users\dwitmer\Documents\CS001X> dir
```

## 4.5 Output the contents of a file

**macOS** Use the `more` command to display the contents of the `lab1.py` file.

```
dwitmer-macbook-pro:CS001X dwitmer$ more lab1.py
```

**Windows** Use the `more` command to display the contents of the `lab1.py` file.

```
C:\Users\dwitmer\Documents\CS001X> more lab1.py
```

In some cases, you may have to use `more < lab1.py` instead.

## 4.6 Make a directory

**macOS** Next, use `mkdir` to make a new directory called `lab1` as follows:

```
dwitmer-macbook-pro:CS001X dwitmer$ mkdir lab1
```

**Windows** Next, use `mkdir` to make a new directory called `lab1` as follows:

```
C:\Users\dwitmer\Documents\CS001X> mkdir lab1
```

## 4.7 Copy a file

**macOS** Use the `cp` command to make a copy of your `lab1.py` file called `lab1copy.py`. Then use `ls` to see the new file.

```
dwitmer-macbook-pro:CS001X dwitmer$ cp lab1.py lab1copy.py
dwitmer-macbook-pro:CS001X dwitmer$ ls
```

Use `more` to verify that it is indeed a copy.

**Windows** Use the `copy` command to make a copy of your `lab1.py` file called `lab1copy.py`. Then use `dir` to see the new file.

```
C:\Users\dwitmer\Documents\CS001X> copy lab1.py lab1copy.py
C:\Users\dwitmer\Documents\CS001X> dir
```

Use `more` to verify that it is indeed a copy.

## 4.8 Delete a file

**macOS** Use `rm` to delete the copy and then confirm that it is gone using `ls`.

```
dwitmer-macbook-pro:CS001X dwitmer$ rm lab1copy.py
```

**Windows** Use `del` to delete the copy and then confirm that it is gone using `dir`.

```
C:\Users\dwitmer\Documents\CS001X> del lab1copy.py
```

## 4.9 Move a file

**macOS** Last, move the `lab1.py` file to the `lab1` directory using `mv`:

```
dwitmer-macbook-pro:CS001X dwitmer$ mv lab1.py lab1/lab1.py
```

To confirm that the move was successful, use the `ls` command. The file should be gone. Then use `ls` again with the name of the new directory:

```
dwitmer-macbook-pro:CS001X dwitmer$ ls lab1
```

The file should be there.

**Windows** Move the `lab1.py` file to the `lab1` directory using `move`:

```
C:\Users\dwitmer\Documents\CS001X> move lab1.py lab1\lab1.py
```

To confirm that the move was successful, use the `dir` command. The file should be gone. Then use `dir` again with the name of the new directory.

```
C:\Users\dwitmer\Documents\CS001X> dir lab1
```

The file should be there.

## 5 Activity: Run a first program

Finally, we can run our program from the command line.

### 5.1 macOS

Use `cd` to change your current directory to `lab1`. Then run the following:

```
python3 lab1.py
```

Your program should print “Hello, world!”.

Using the command `python` (rather than `python3`) runs an older version of Python that comes preinstalled with macOS. We will not use this older version of Python in this class.

### 5.2 Windows

Use `chdir` to change your current directory to `lab1`. Then run the following:

```
python lab1.py
```

Your program should print “Hello, world!”.