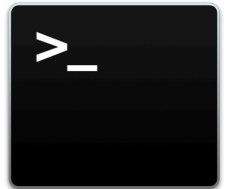


Working Remotely for CS134

What You Need

- Text editor for writing programs
 - Ideal: **Atom**, the application we have been using so far
- **Python3** for executing programs
 - via Terminal in Mac
 - via Command prompt in Windows
- **Git** for submitting work and downloading starters
 - If using Atom, can commit/push/pull as usual
- **VPN client CiscoAnyConnect** to log on to the CS server **evolene**
 - Information on downloading available at



<https://oit.williams.edu/help-guides/wifi-and-wired-connections/vpn/>

VPN (Virtual Private Network)

- Very important step, get it installed now!

VPN (Virtual Private Networking)

Williams VPN

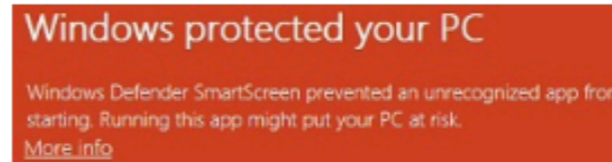
Running the VPN client allows you to establish a secure connection to the college network. Your computer acts as though it were here on campus. Among other things this allows a keyserver connection to run Williams licensed software from off-campus. The VPN connection is limited to Faculty, Staff and Students who are doing academic or administrative work.

- **macOS X:**
In the Applications folder open "Self Service" and run the Cisco AnyConnect VPN client. Once installed you'll need to enter the Server Address:

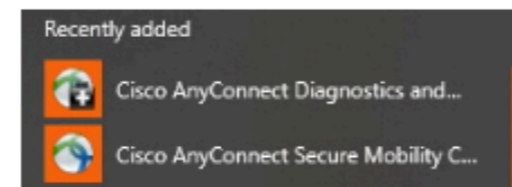
`ssl-vpn.williams.edu` (*this is a one time entry*)

- You will then be presented with a login screen. Use your short username and the password you use for email. If you have any questions call the HelpDesk: 413-597-4090 or email: itsupport@williams.edu.

- **Windows:**
Download the VPN client, called Cisco AnyConnect, from the [software download site](#). When you run it you will be warned that this is an unknown application



- This is OK! Click the More info link and choose Run Anyway. You will be prompted at least one more time as to whether you want to install the software. Keep clicking Allow.
- You will have to reboot. After rebooting you will have the Cisco AnyConnect Secure Mobility Client in your program list from the Start menu.



- When you first run it enter the Server Address:

`ssl-vpn.williams.edu` (*This is a one time entry*)

- You will then be presented with a login screen. Use your short username and the password you use for email.

Getting Set Up: Windows

Windows Commands vs Unix

Commands on Mac

Equivalent on Windows

(list contents of directory) `ls` ↔ `dir`

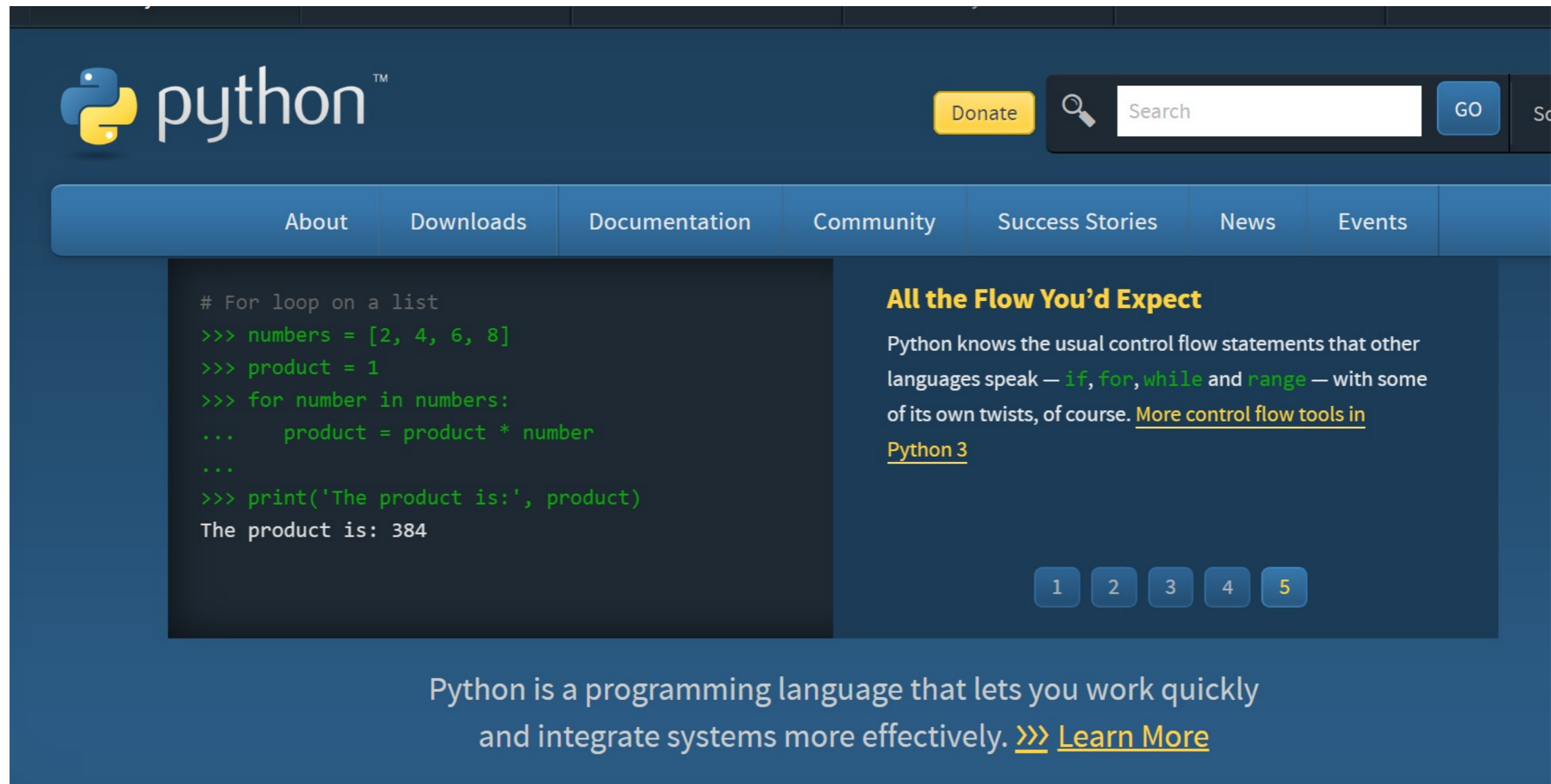
(change directory) `cd` ↔ `cd`

(make directory) `mkdir` ↔ `mkdir`

(move to root) `cd /` ↔ `cd \`

Python3 Setup: Windows

- Log on to www.python.org



The screenshot shows the Python.org website with a dark blue header. The Python logo and 'python™' are on the left. A yellow 'Donate' button is next to a search bar with a magnifying glass icon and a 'GO' button. Below the header is a navigation menu with links for 'About', 'Downloads', 'Documentation', 'Community', 'Success Stories', 'News', and 'Events'. The main content area features a code block on the left and an article titled 'All the Flow You'd Expect' on the right. The code block shows a Python script for calculating the product of a list of numbers. The article text discusses control flow statements like 'if', 'for', 'while', and 'range'. At the bottom of the page, there is a footer with the text: 'Python is a programming language that lets you work quickly and integrate systems more effectively. >>> [Learn More](#)'.

```
# For loop on a list
>>> numbers = [2, 4, 6, 8]
>>> product = 1
>>> for number in numbers:
...     product = product * number
...
>>> print('The product is:', product)
The product is: 384
```

All the Flow You'd Expect

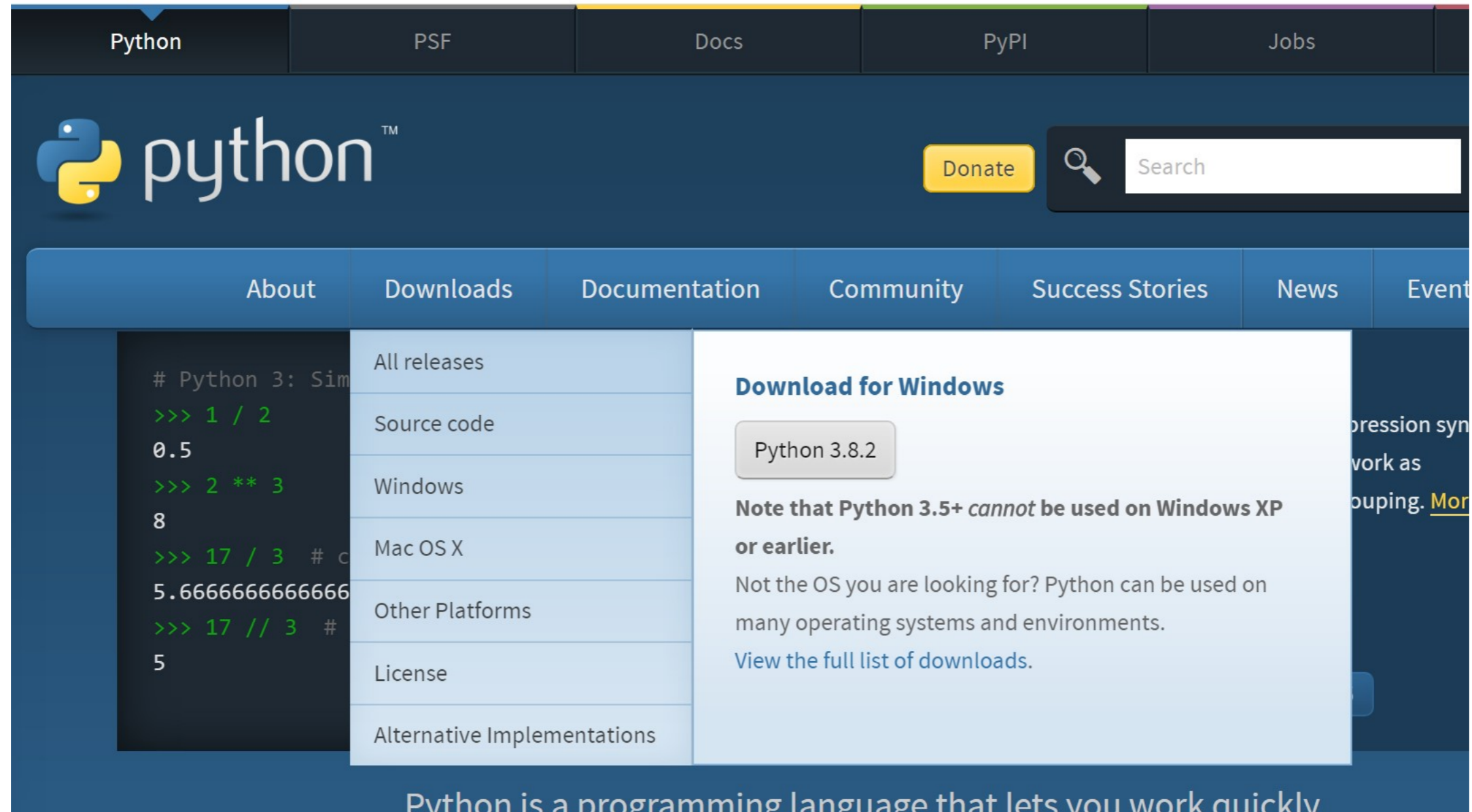
Python knows the usual control flow statements that other languages speak — `if`, `for`, `while` and `range` — with some of its own twists, of course. [More control flow tools in Python 3](#)

1 2 3 4 5

Python is a programming language that lets you work quickly and integrate systems more effectively. >>> [Learn More](#)

Python3 Setup: Windows

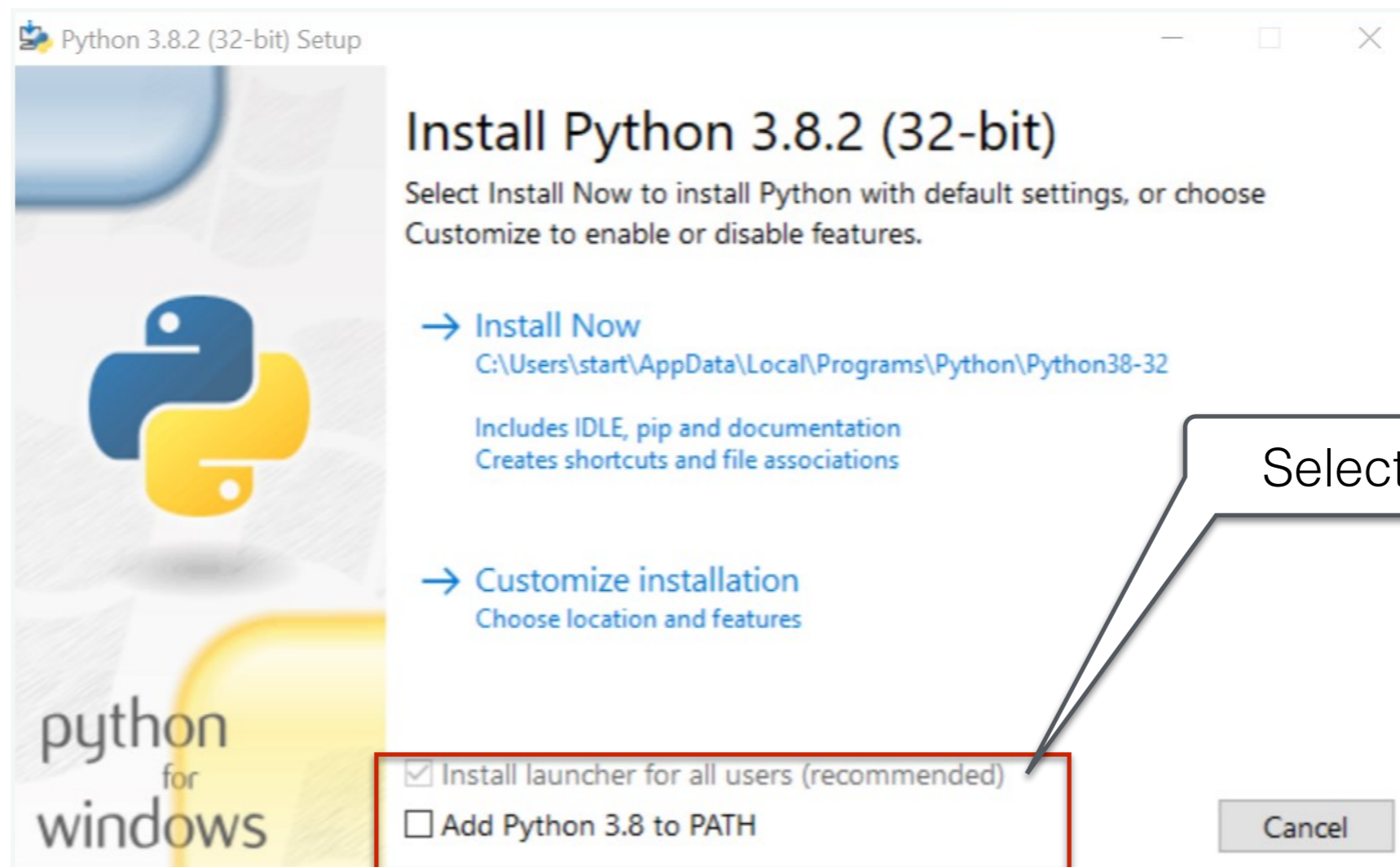
- Go to Downloads -> Download Python 3.8.2 for Windows



The screenshot shows the Python website's navigation menu with 'Downloads' selected. A dropdown menu is open, listing options: All releases, Source code, Windows, Mac OS X, Other Platforms, License, and Alternative Implementations. The 'Windows' option is highlighted, and a sub-menu is visible for 'Download for Windows'. This sub-menu includes a button for 'Python 3.8.2', a note stating 'Note that Python 3.5+ cannot be used on Windows XP or earlier.', and a link to 'View the full list of downloads.' The background of the website shows a terminal window with Python code and the Python logo.

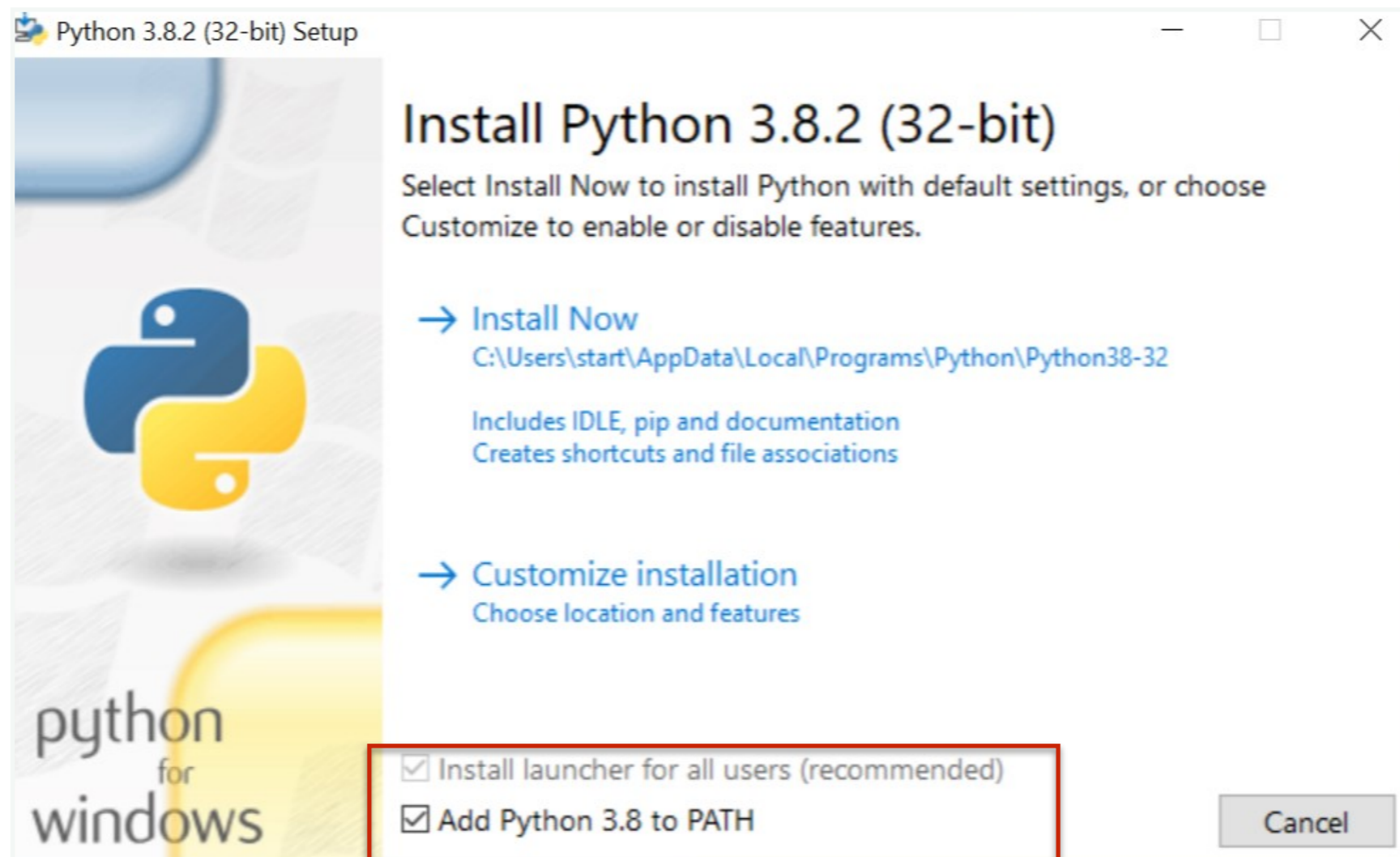
Python3 Setup: Windows

- The executable will download automatically and afterwards show the following pop up for installation



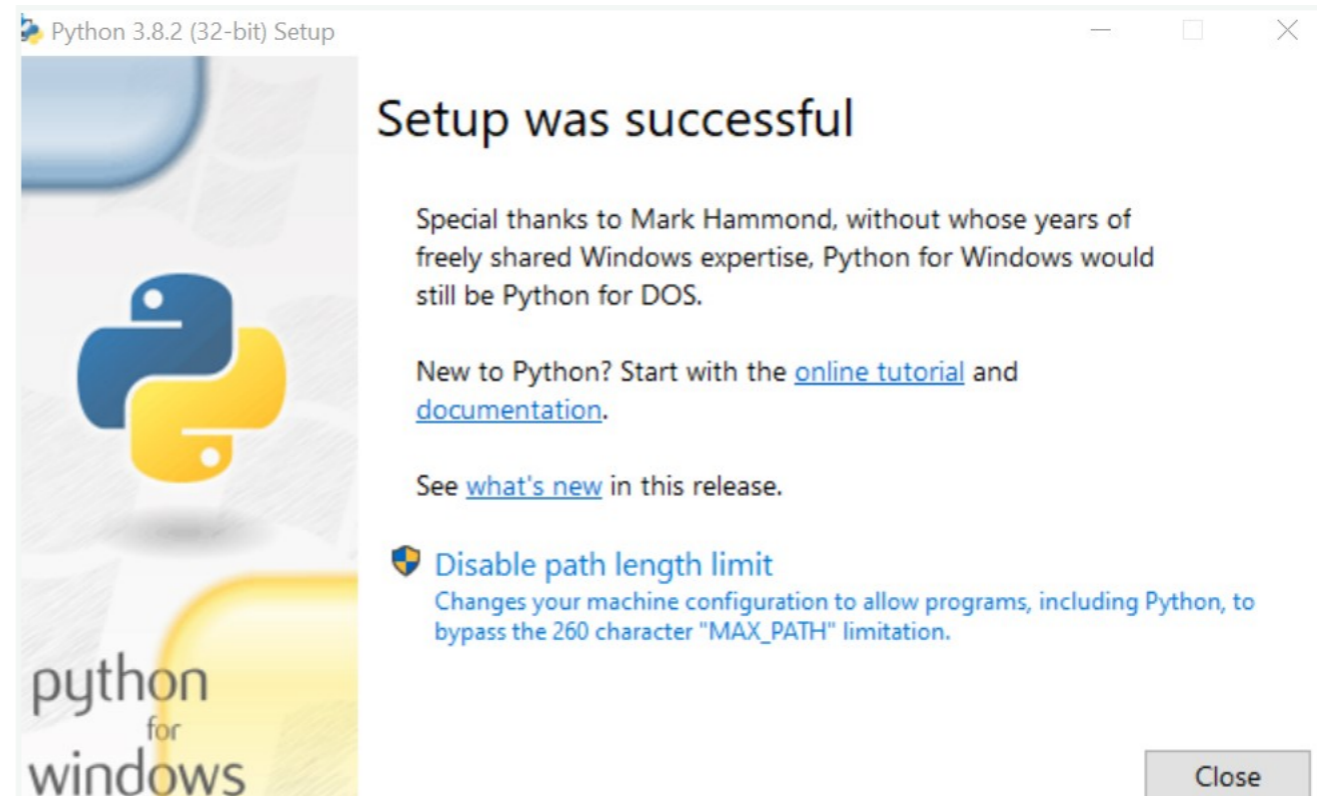
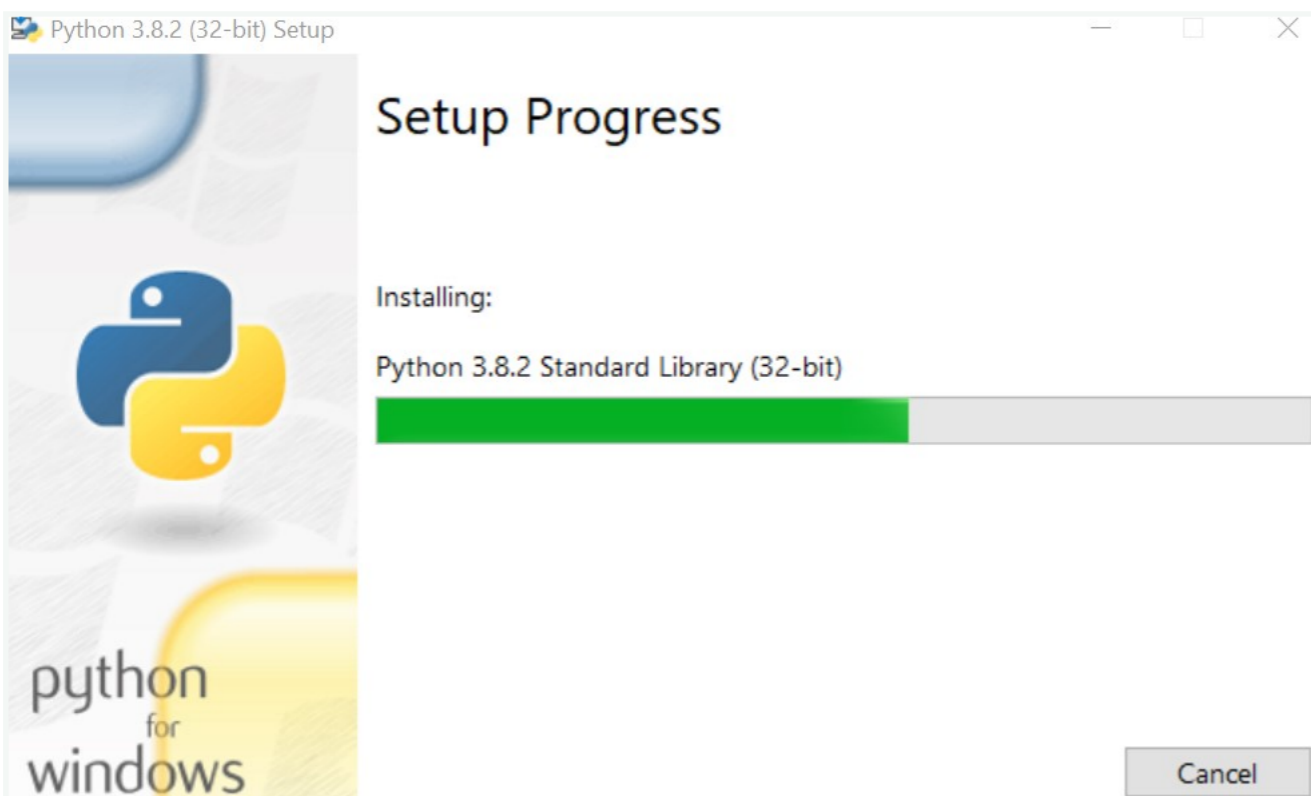
Python3 Setup: Windows

- Make sure to select the Add Python 3.8 to PATH option



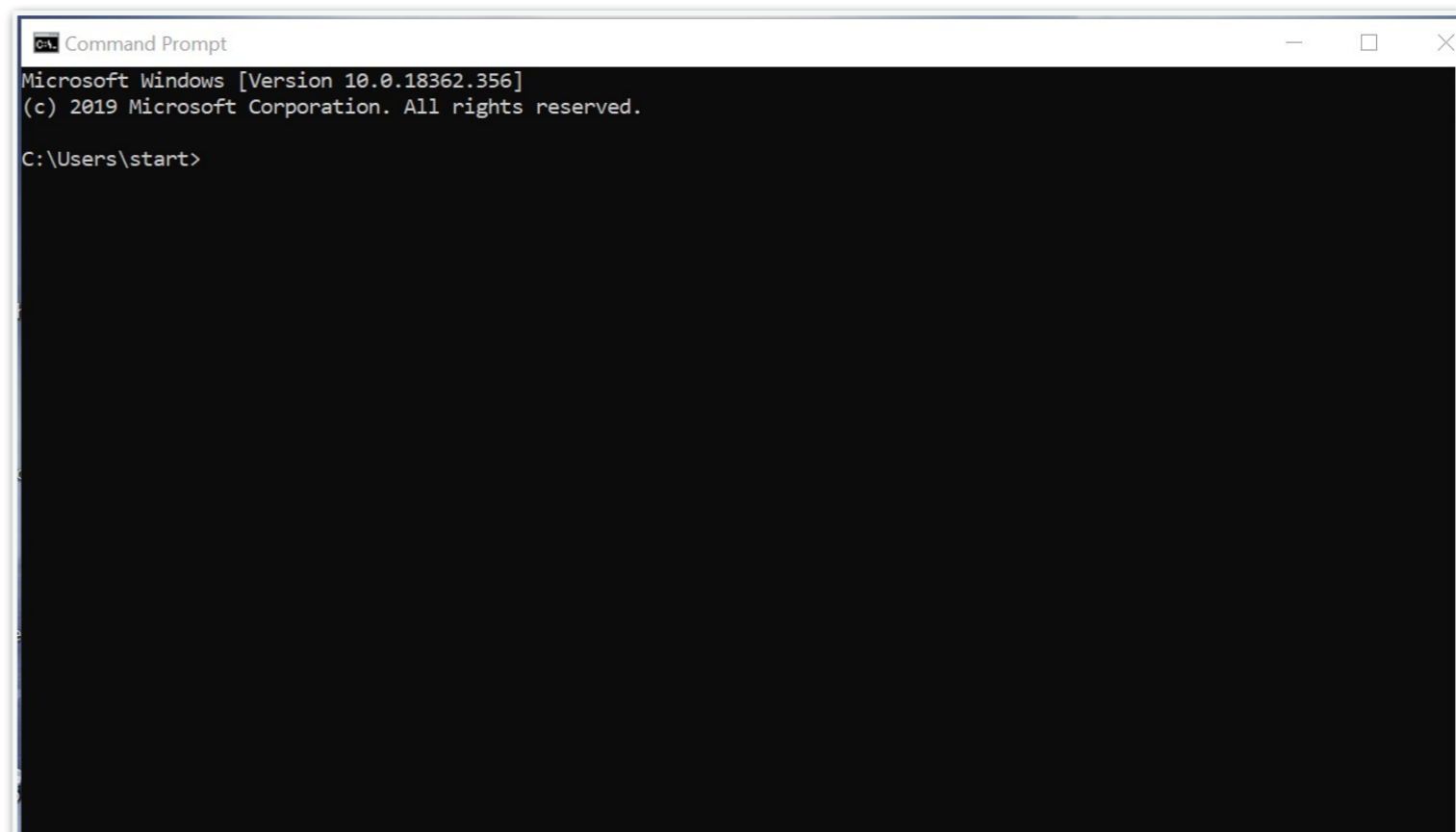
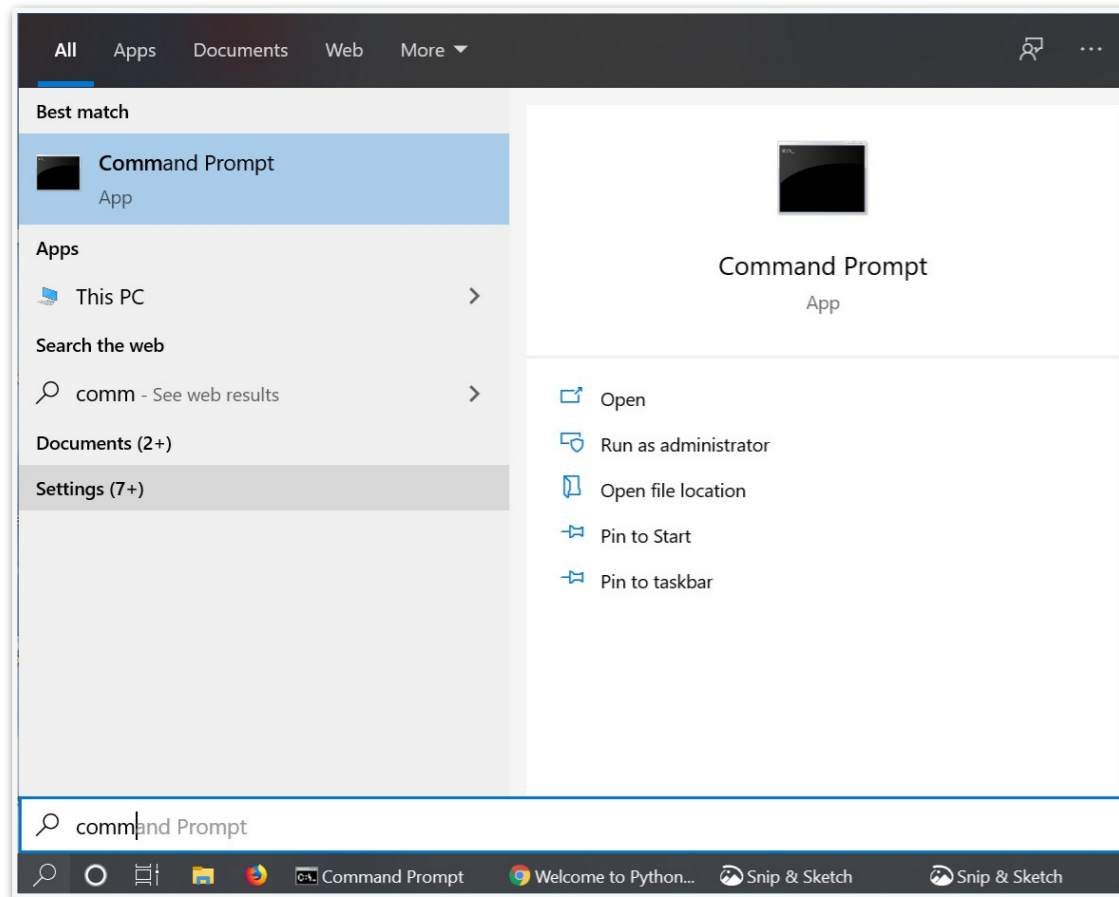
Python3 Setup: Windows

- Wait for installation to finish



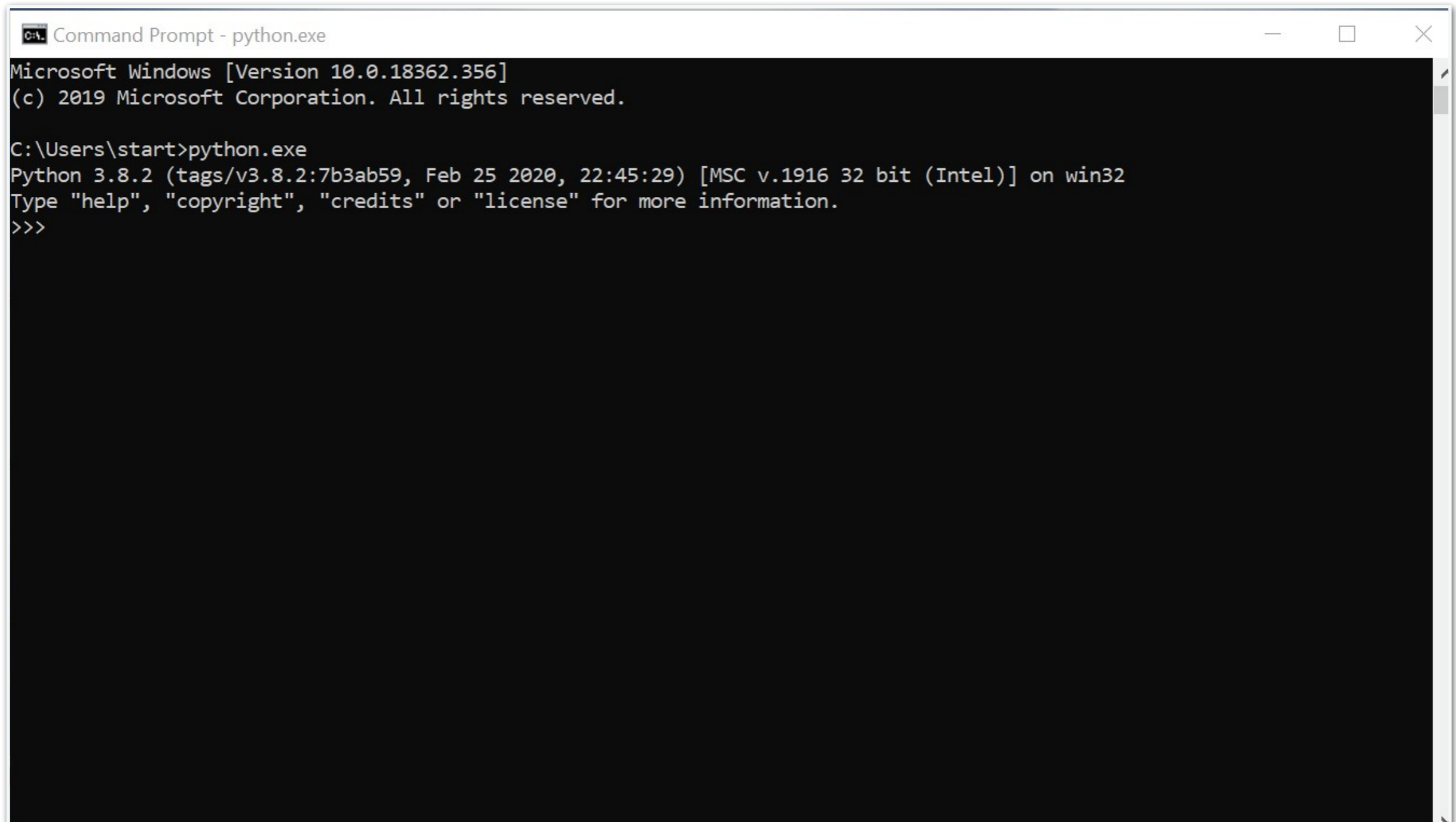
Python3 Setup: Windows

- Launch command prompt from start menu



Python3 Setup: Windows

- Type and enter `python.exe` to launch interactive python

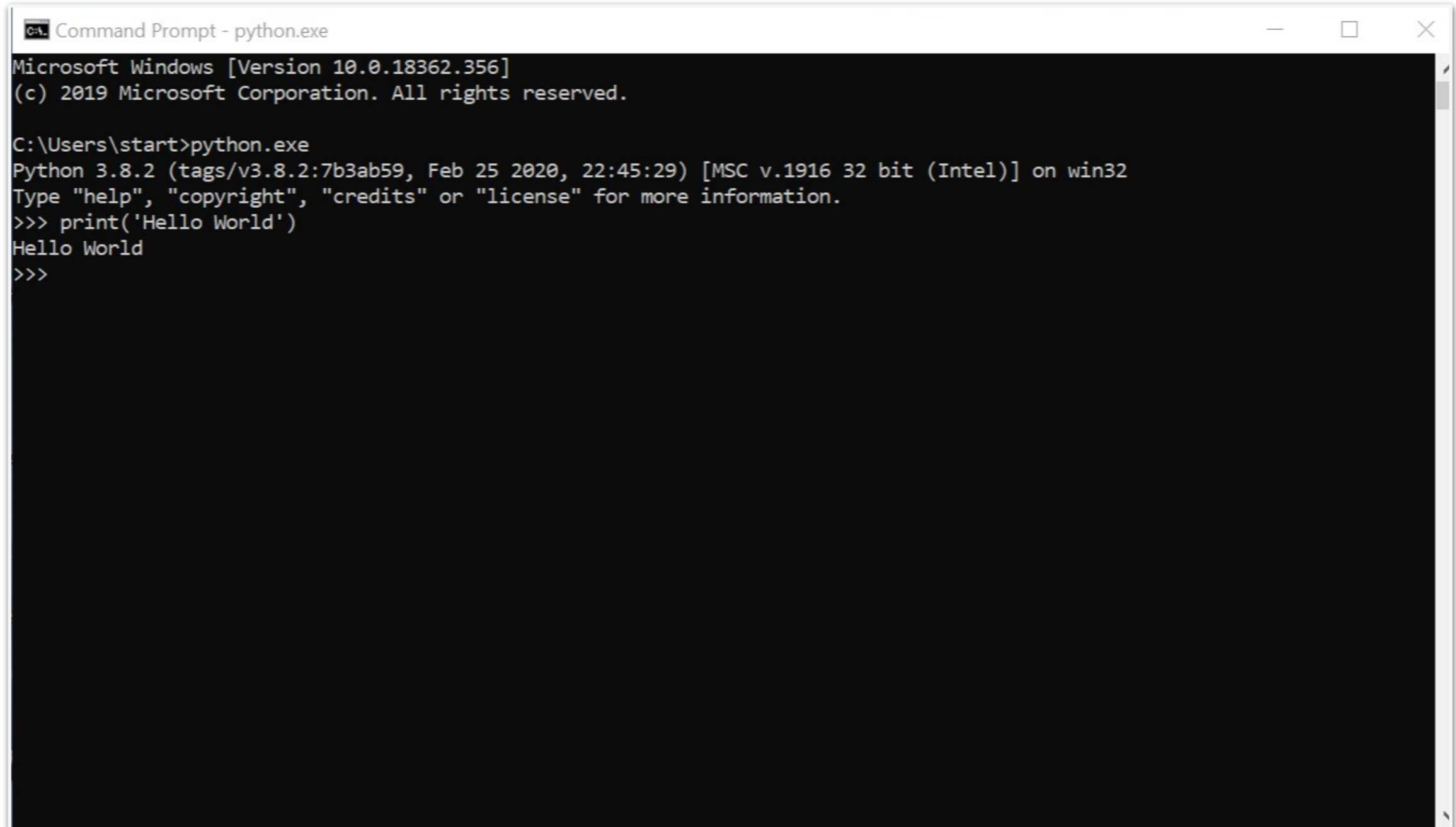


```
Command Prompt - python.exe
Microsoft Windows [Version 10.0.18362.356]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\start>python.exe
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

Python3 Setup: Windows

- Print “Hello World” to celebrate!

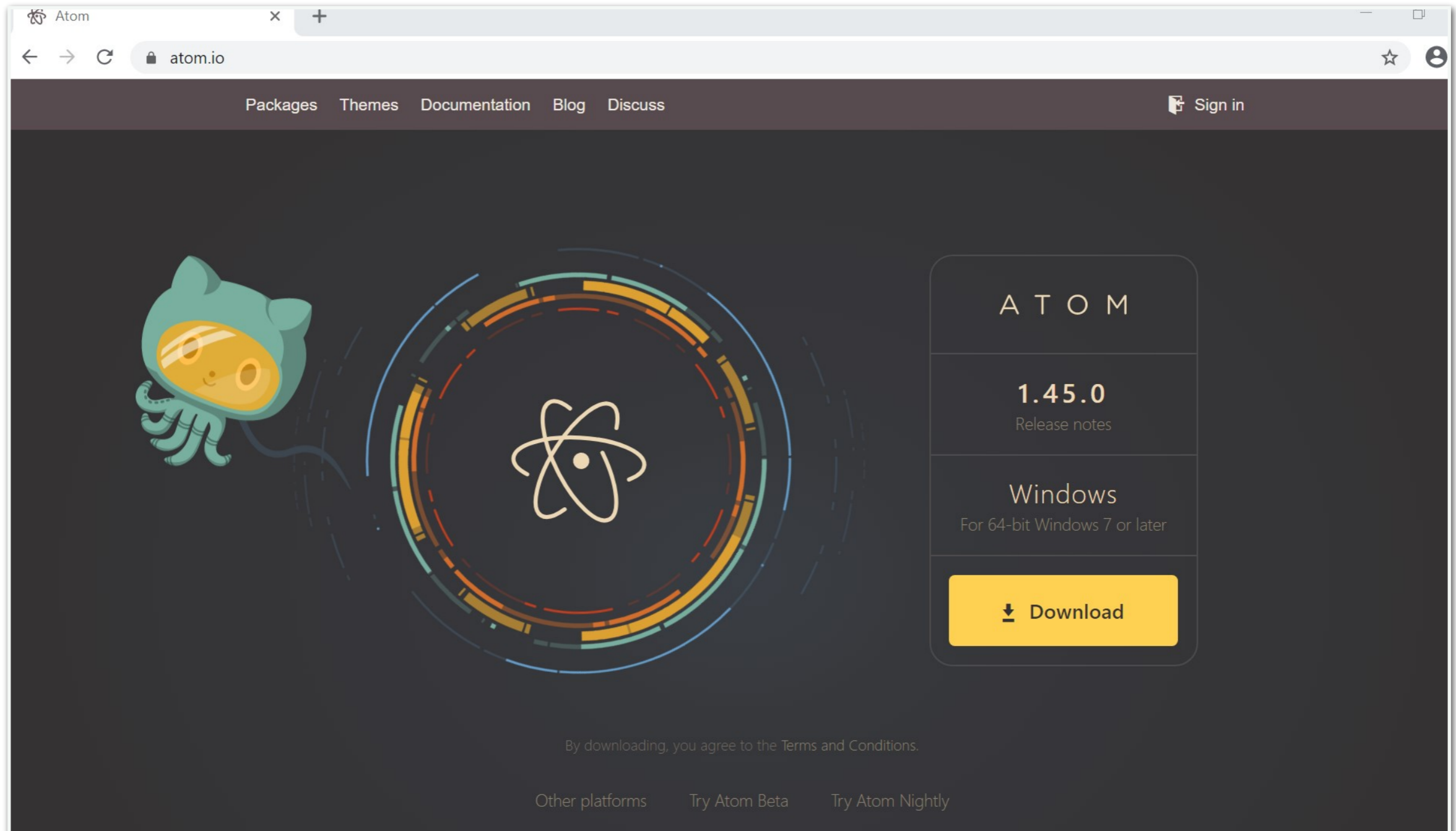


```
Command Prompt - python.exe
Microsoft Windows [Version 10.0.18362.356]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\start>python.exe
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> print('Hello World')
Hello World
>>>
```

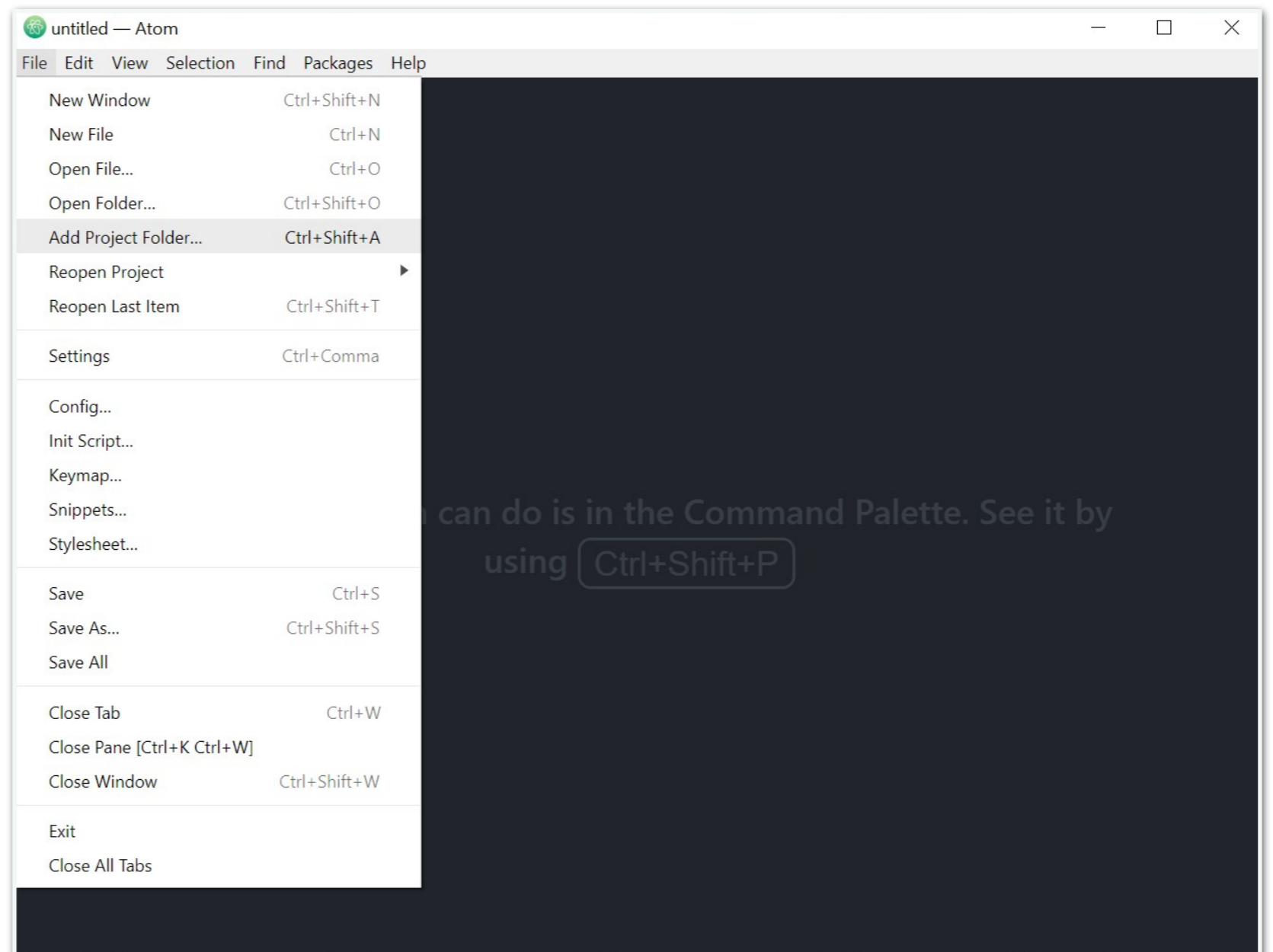
Atom Setup: Windows

- Go to <http://atom.io> and click on Download for Windows



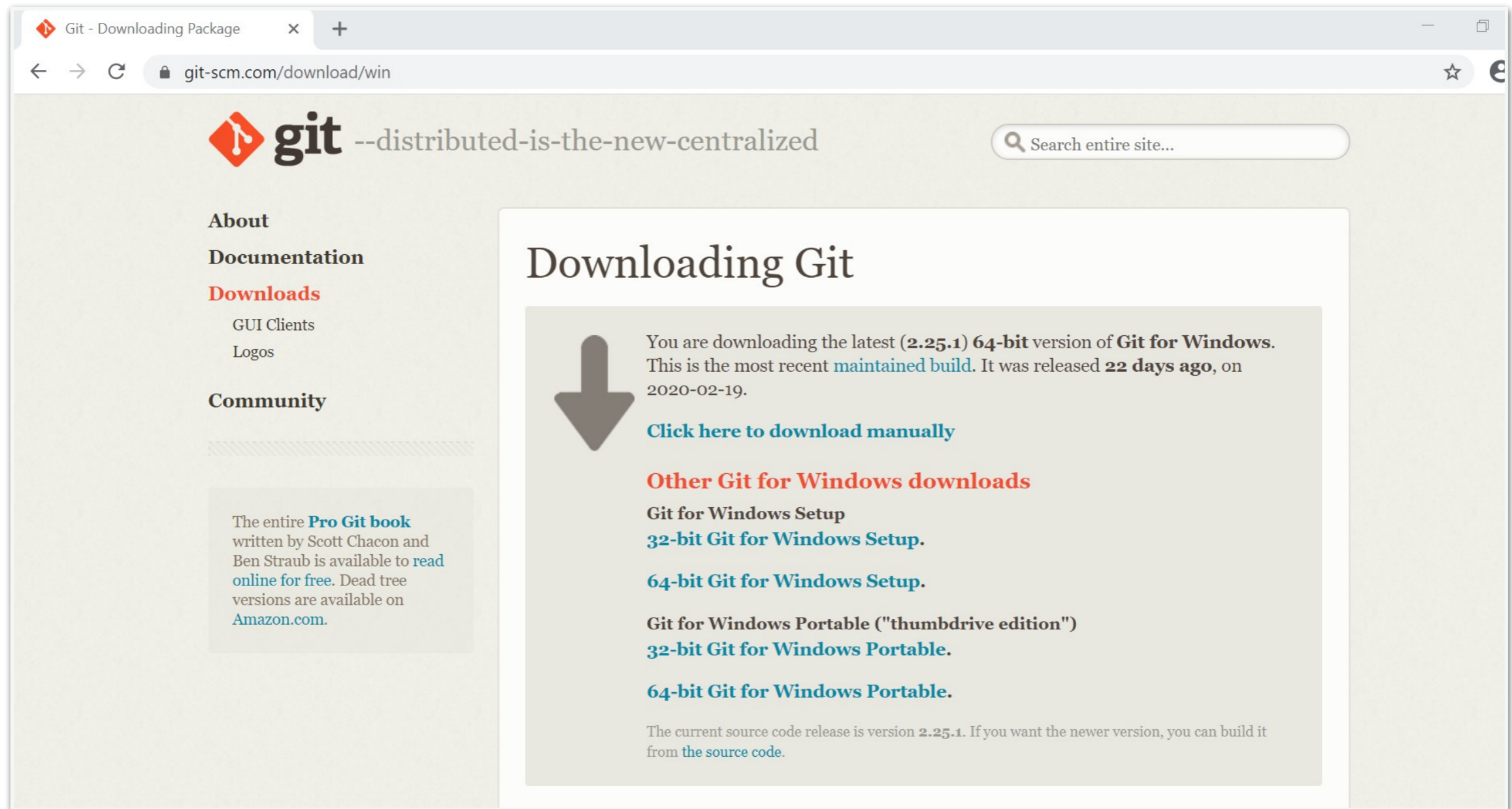
Atom Setup: Windows

- After it is done downloading, it will launch the application which can be used as as have in the lab



Git Setup: Windows

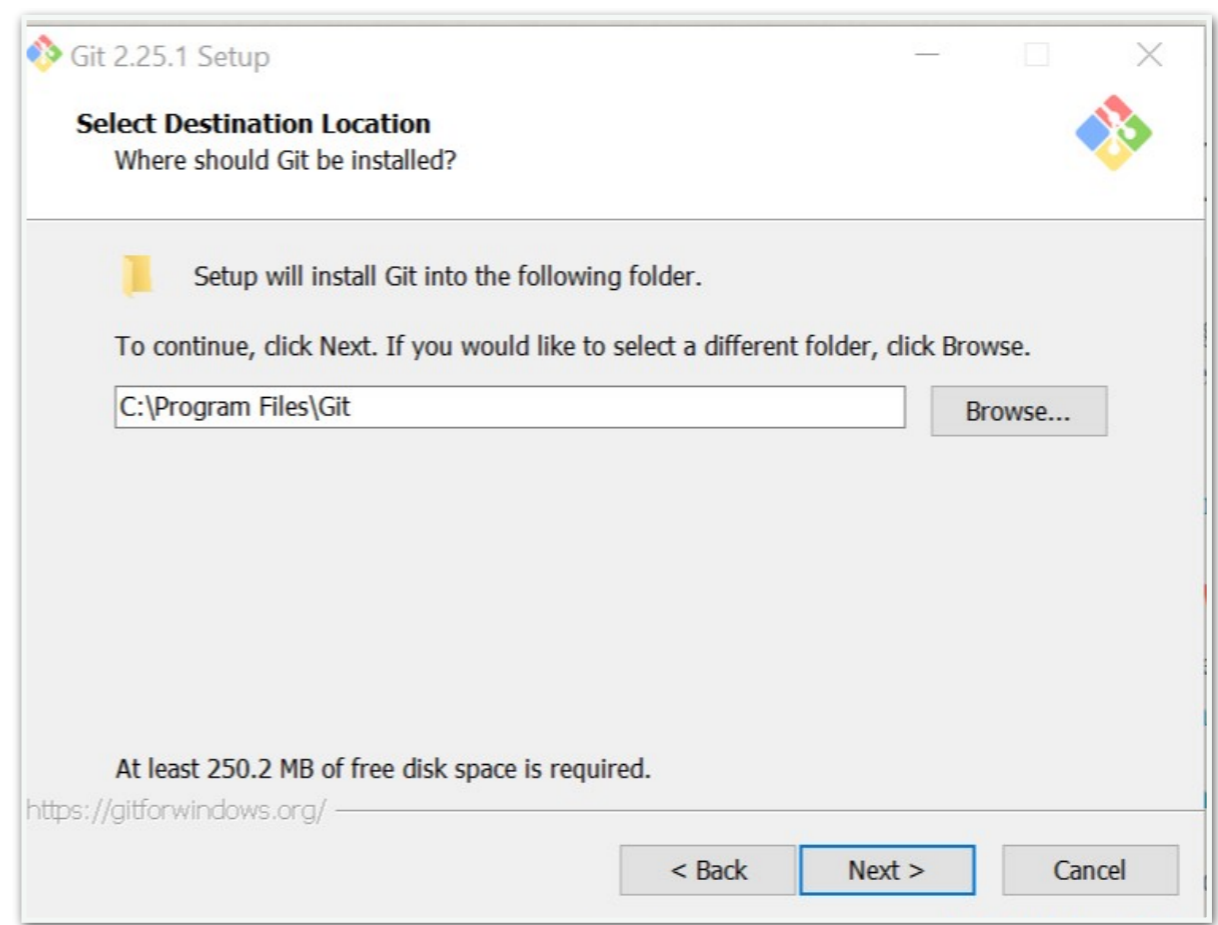
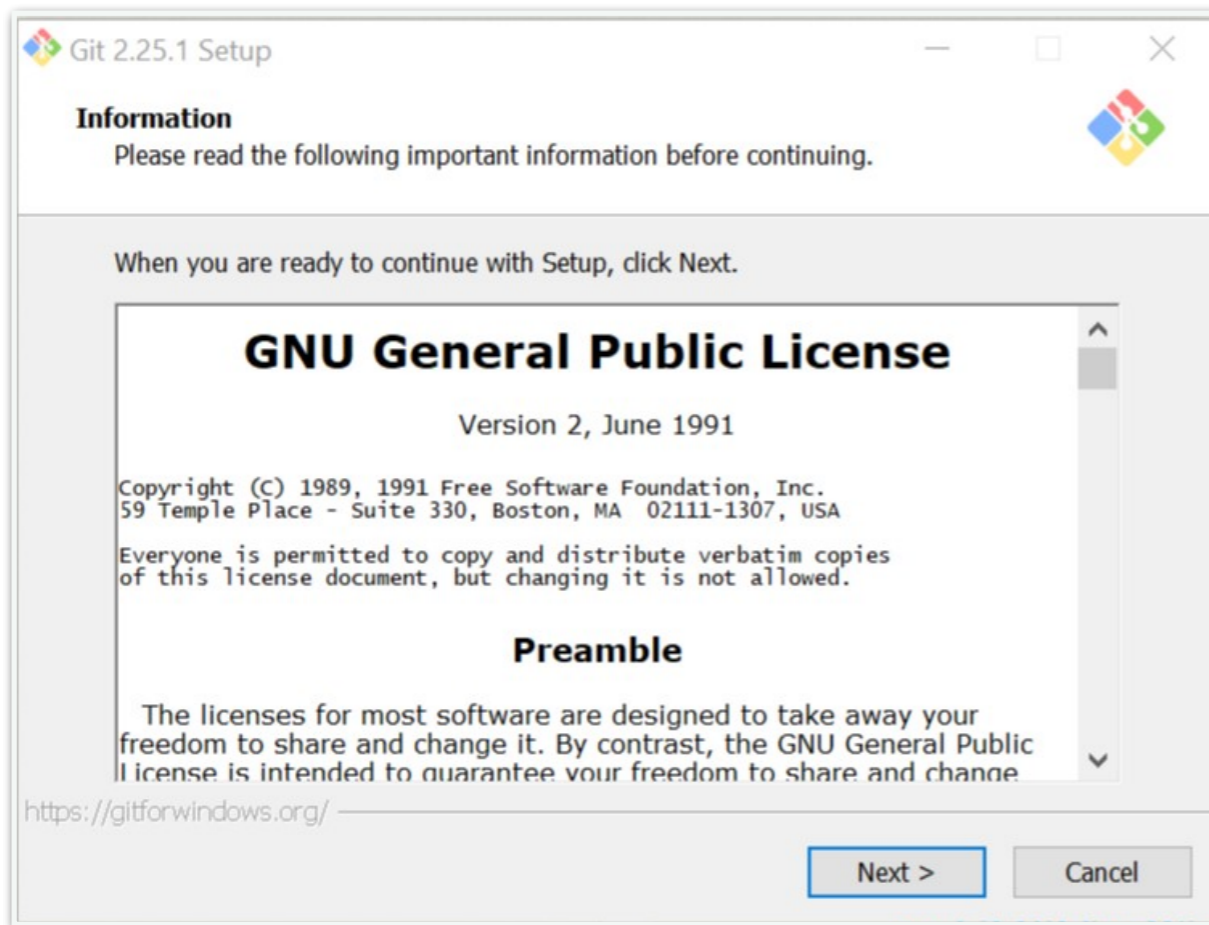
- Go to www.git-scm.com/download/win and click on the download link



The screenshot shows a web browser window with the address bar displaying git-scm.com/download/win. The page features the Git logo and the tagline "--distributed-is-the-new-centralized". A search bar is located in the top right corner. On the left side, there is a navigation menu with links for "About", "Documentation", "Downloads" (highlighted in red), "GUI Clients", "Logos", and "Community". Below the menu, a text box promotes the "Pro Git book" by Scott Chacon and Ben Straub, available for free online or on Amazon.com. The main content area is titled "Downloading Git" and includes a large downward-pointing arrow. The text below the arrow states: "You are downloading the latest (2.25.1) 64-bit version of Git for Windows. This is the most recent maintained build. It was released 22 days ago, on 2020-02-19." Below this, there is a link "Click here to download manually". A section titled "Other Git for Windows downloads" lists several options: "Git for Windows Setup", "32-bit Git for Windows Setup.", "64-bit Git for Windows Setup.", "Git for Windows Portable ('thumbdrive edition')", "32-bit Git for Windows Portable.", and "64-bit Git for Windows Portable.". At the bottom of this section, it notes that the current source code release is version 2.25.1 and provides a link to "the source code".

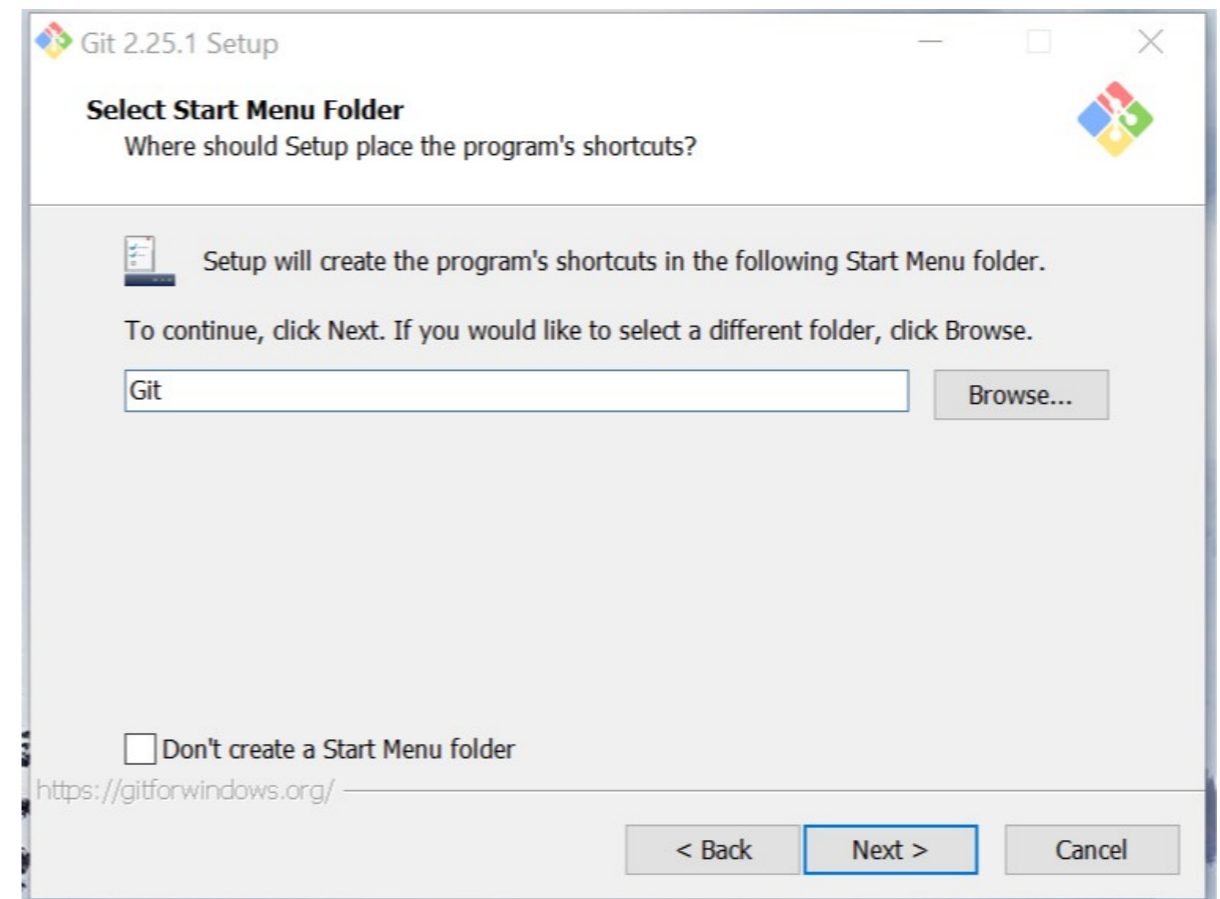
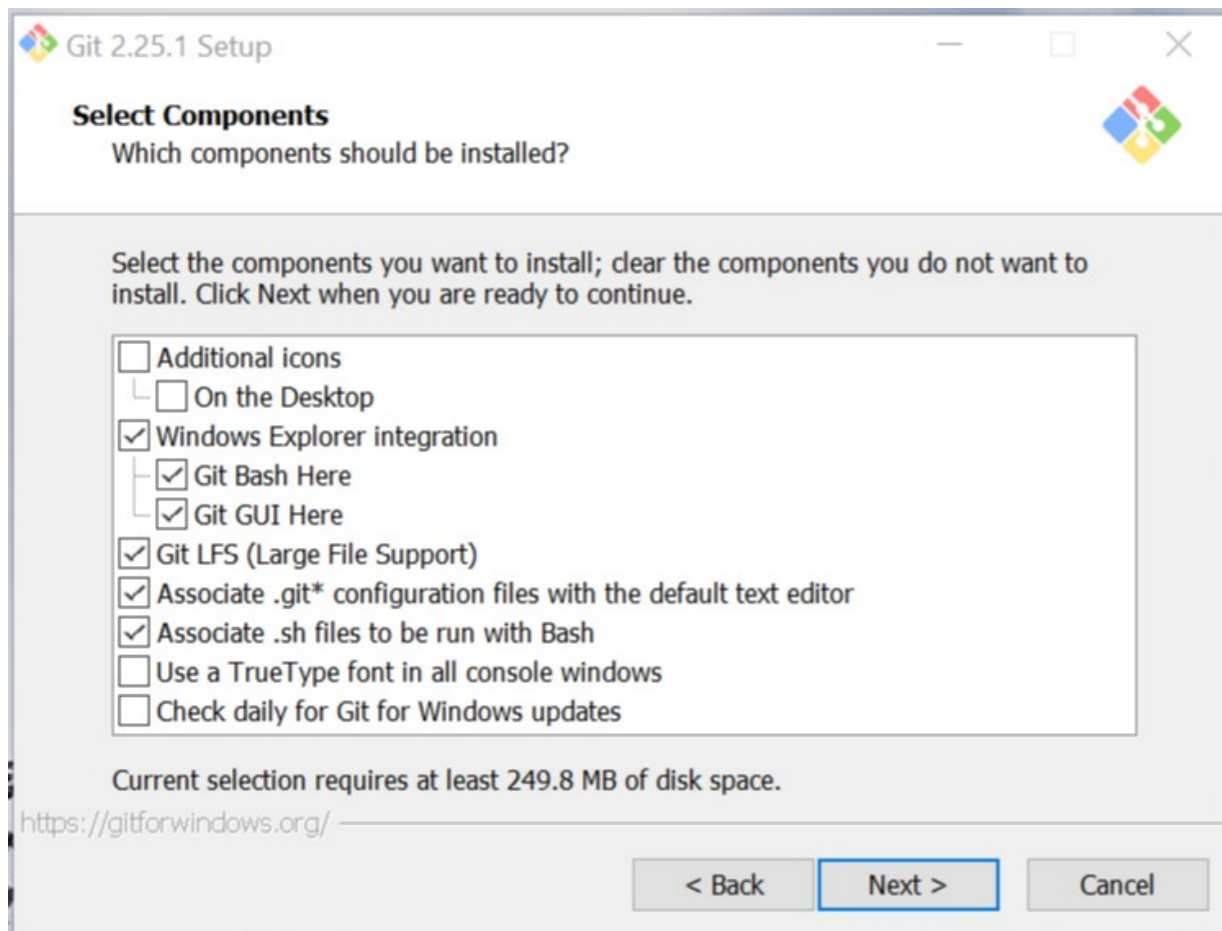
Git Setup: Windows

- The installer will launch through the installation setup



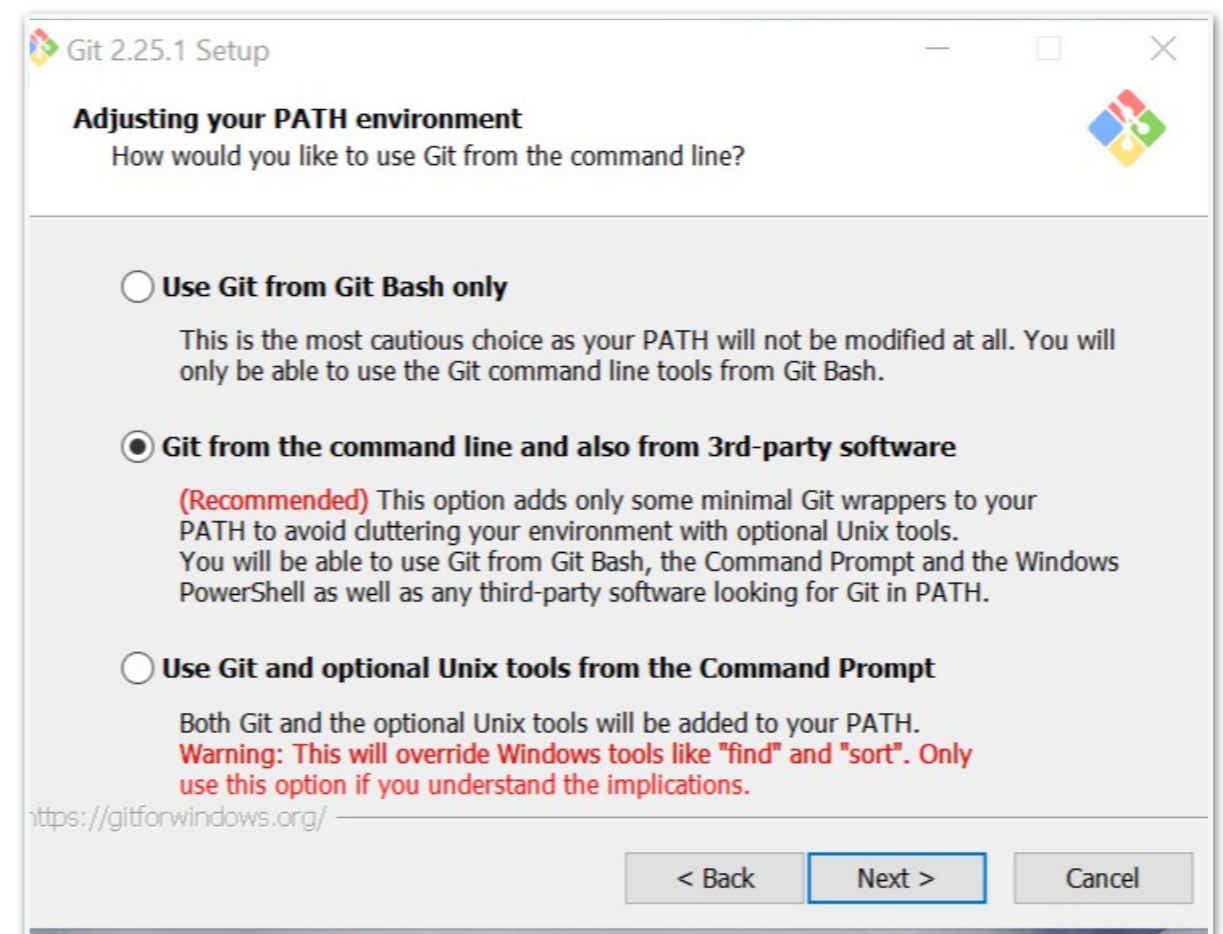
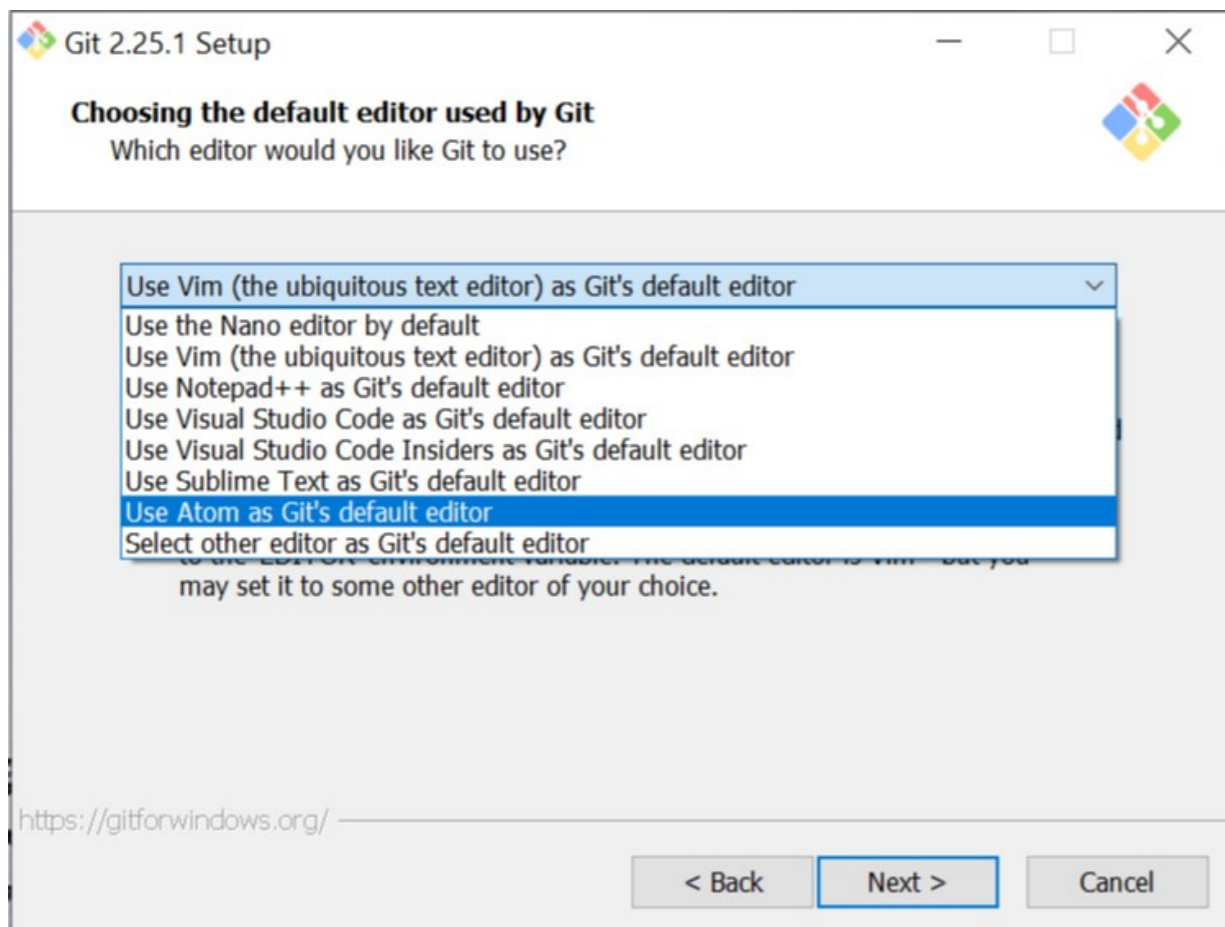
Git Setup: Windows

- The installer will launch through the installation setup



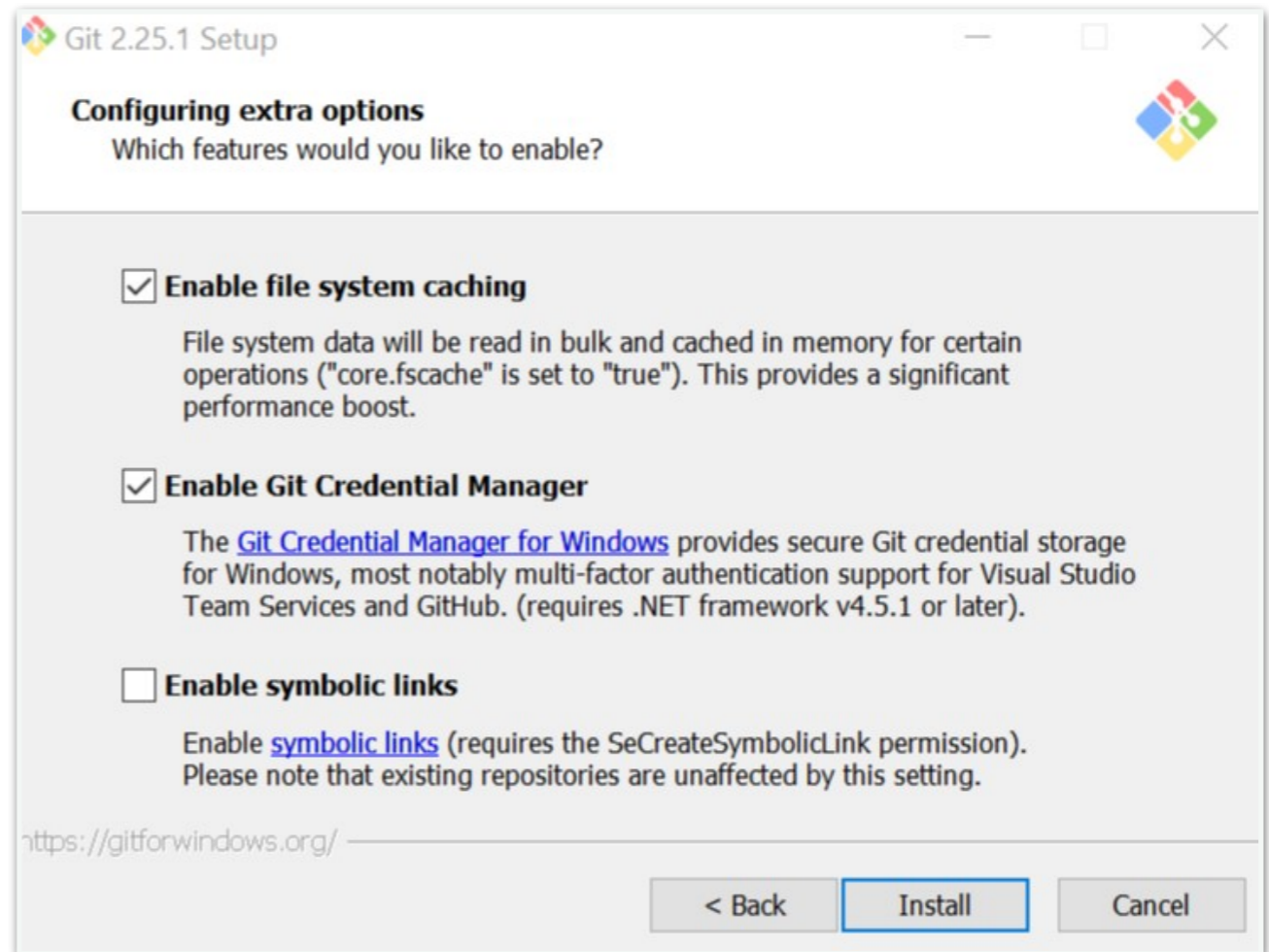
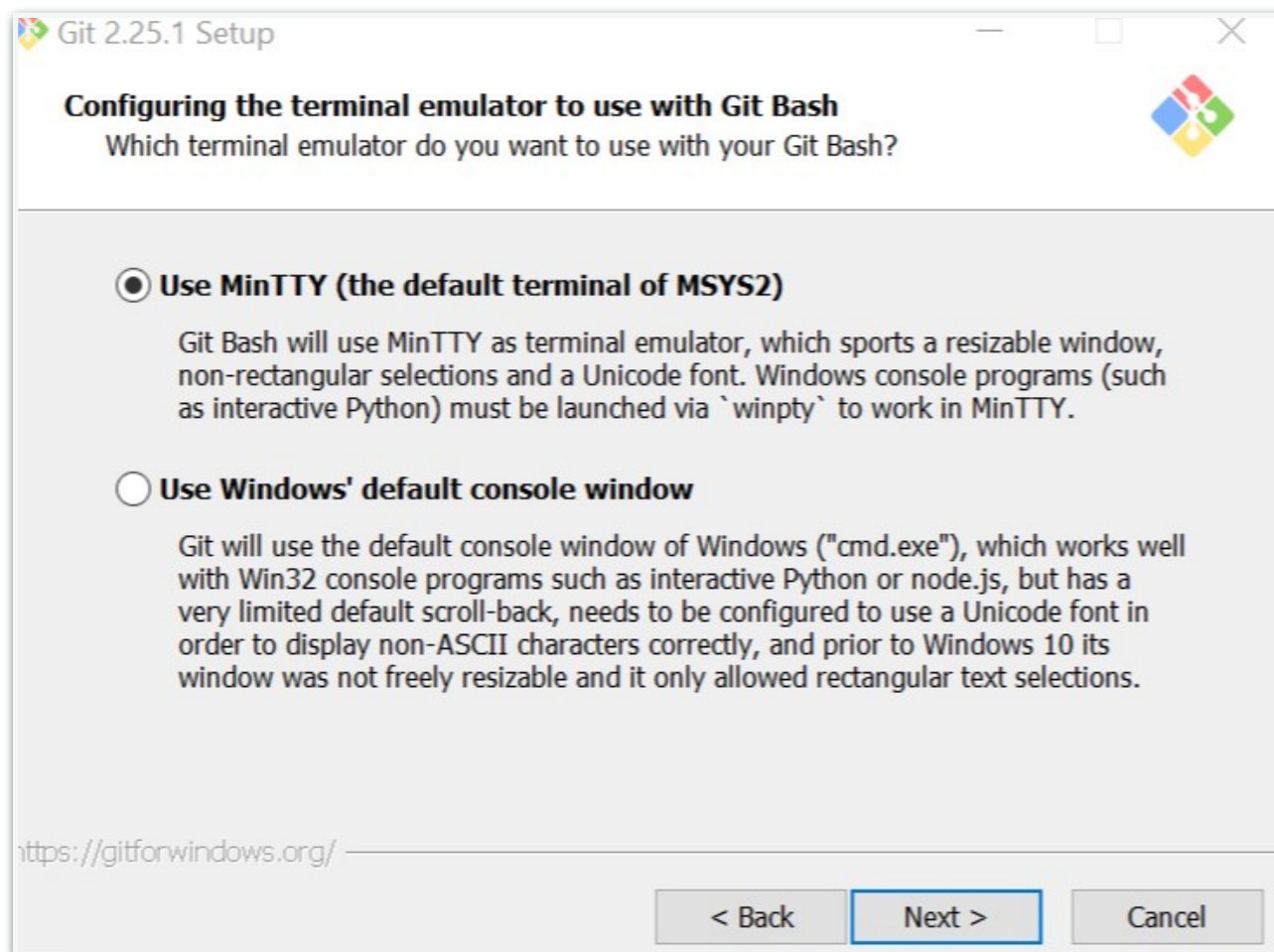
Git Setup: Windows

- Select Atom as Git's default editor
- Select git from command line and also 3rd-part software



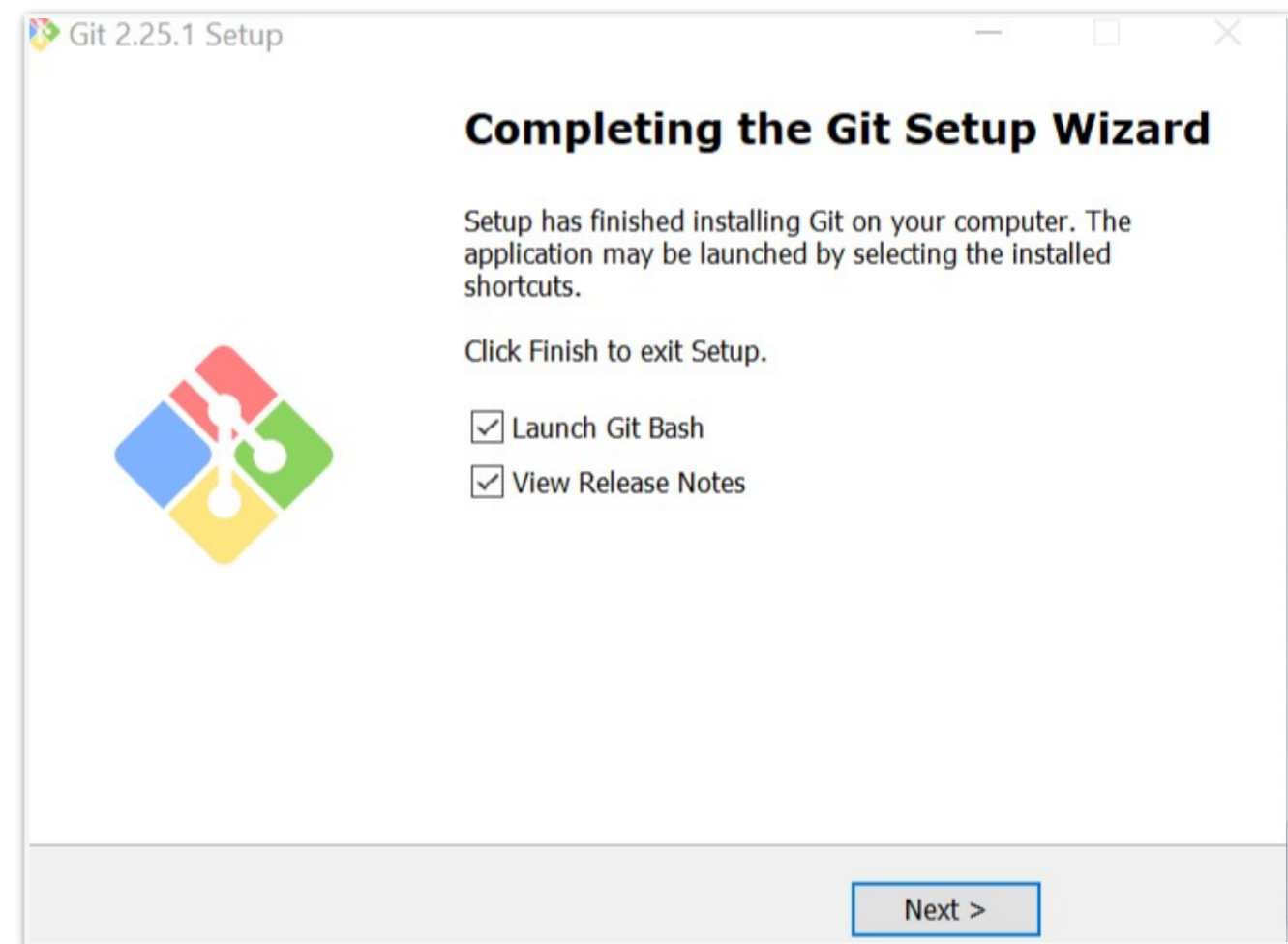
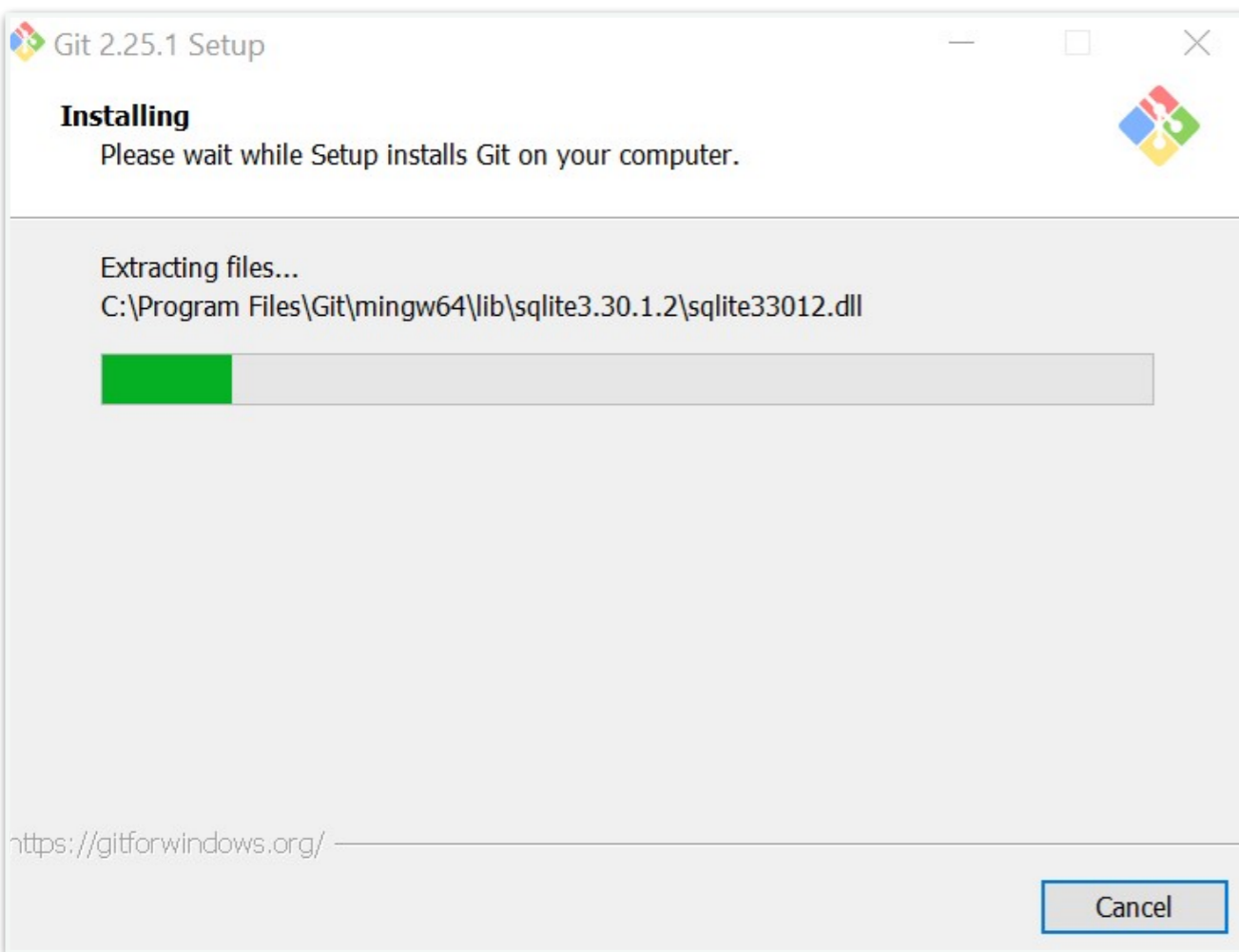
Git Setup: Windows

- Select the following options as indicated



Git Setup: Windows

- Once all setup options are complete, it will finish installation



One-Time Git Config Setup

- Before we clone our lab repository from evolene, we must run the following commands in the “**Git Bash**” to configure our setup

```
git config --global user.name "ephelia2"
```

(replace ephelia2 with your CS username)

```
git config --global user.email "ephelia2@cs.williams.edu"
```

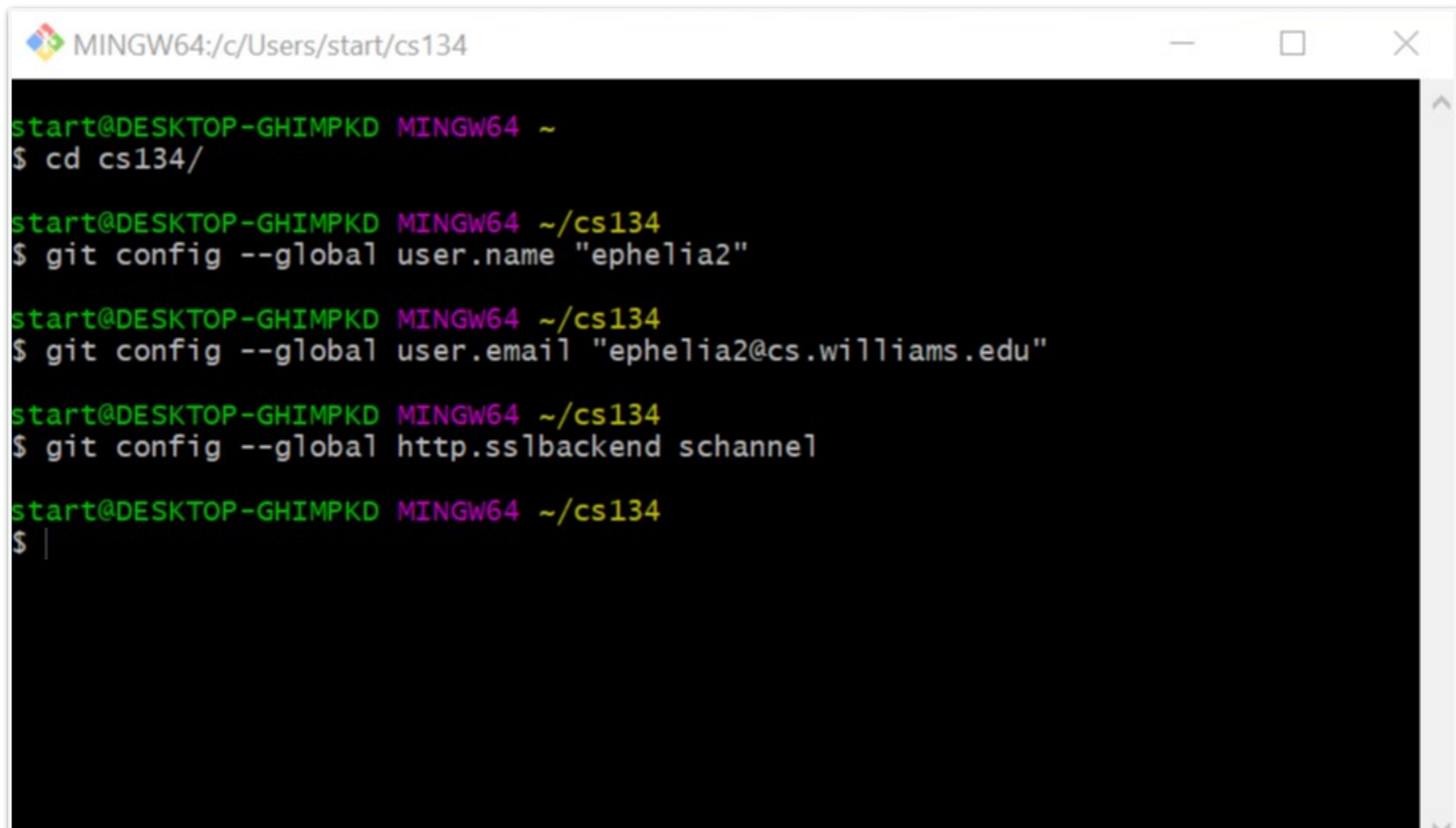
(replace with your CS email)

```
git config --global http.sslbackend schannel
```

[See example screenshot on next slide]

One-Time Git Config Setup

- Before we clone our lab repository from evolene, we must run the following commands in the “Git Bash” to configure our setup



```
MINGW64:/c/Users/start/cs134
start@DESKTOP-GHIMPKD MINGW64 ~
$ cd cs134/

start@DESKTOP-GHIMPKD MINGW64 ~/cs134
$ git config --global user.name "ephelia2"

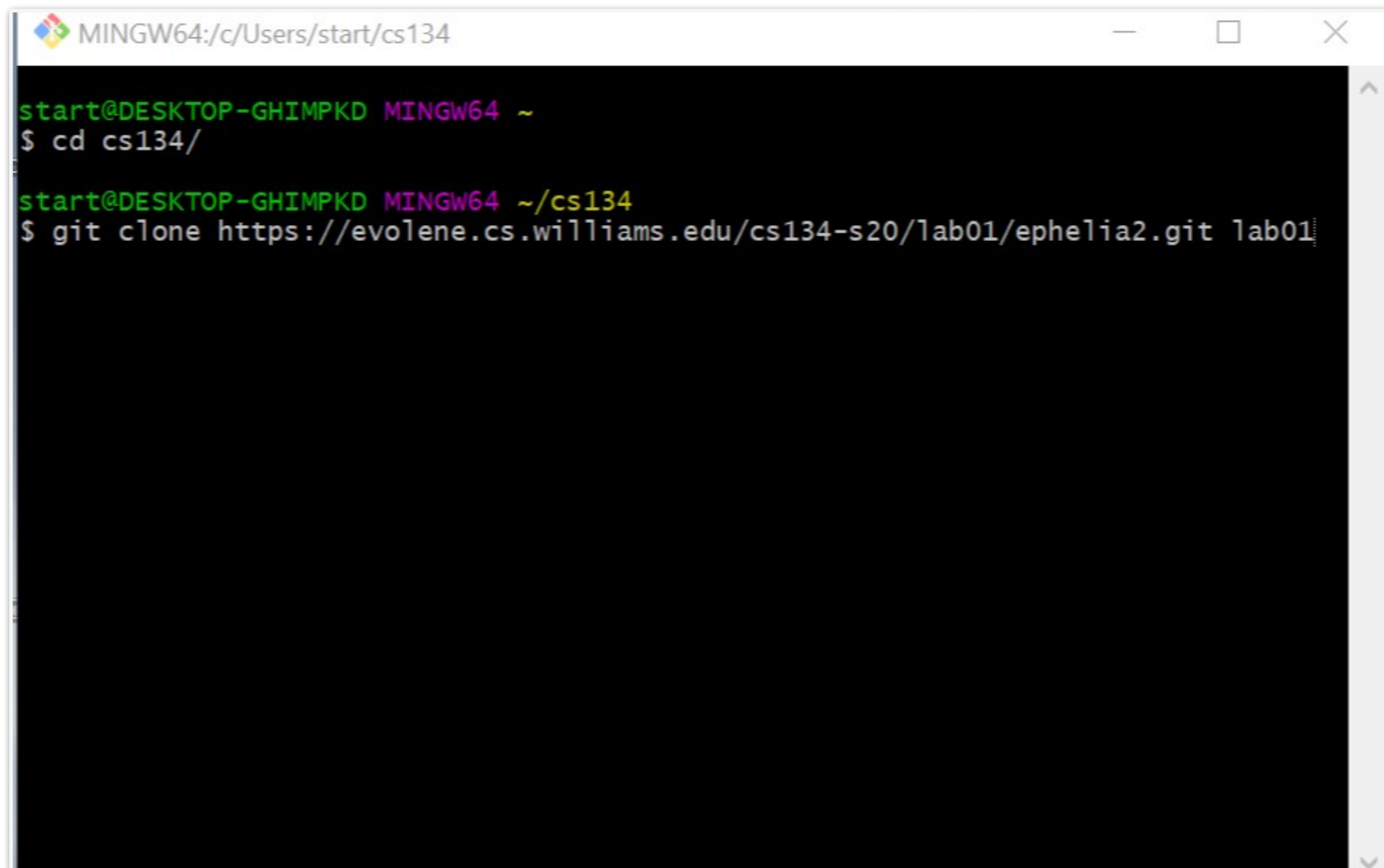
start@DESKTOP-GHIMPKD MINGW64 ~/cs134
$ git config --global user.email "ephelia2@cs.williams.edu"

start@DESKTOP-GHIMPKD MINGW64 ~/cs134
$ git config --global http.sslbackend schannel

start@DESKTOP-GHIMPKD MINGW64 ~/cs134
$ |
```

Cloning Repositories from Evolene

- Your git installation comes with a git bash that looks like this
- Here it is being used to clone the lab01 repository in cs134 folder



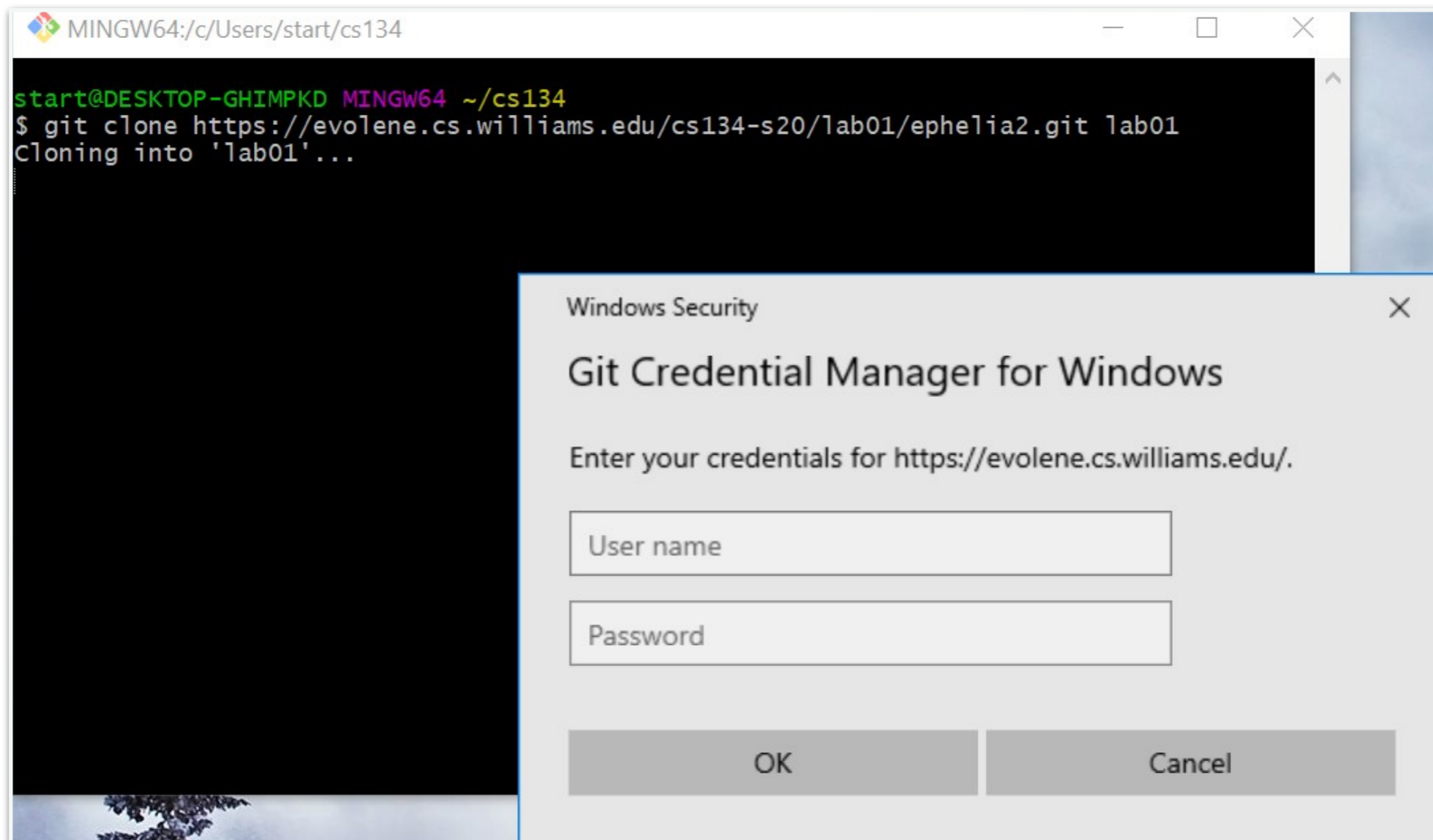
```
MINGW64:/c/Users/start/cs134

start@DESKTOP-GHIMPKD MINGW64 ~
$ cd cs134/

start@DESKTOP-GHIMPKD MINGW64 ~/cs134
$ git clone https://evolene.cs.williams.edu/cs134-s20/lab01/ephelia2.git lab01
```

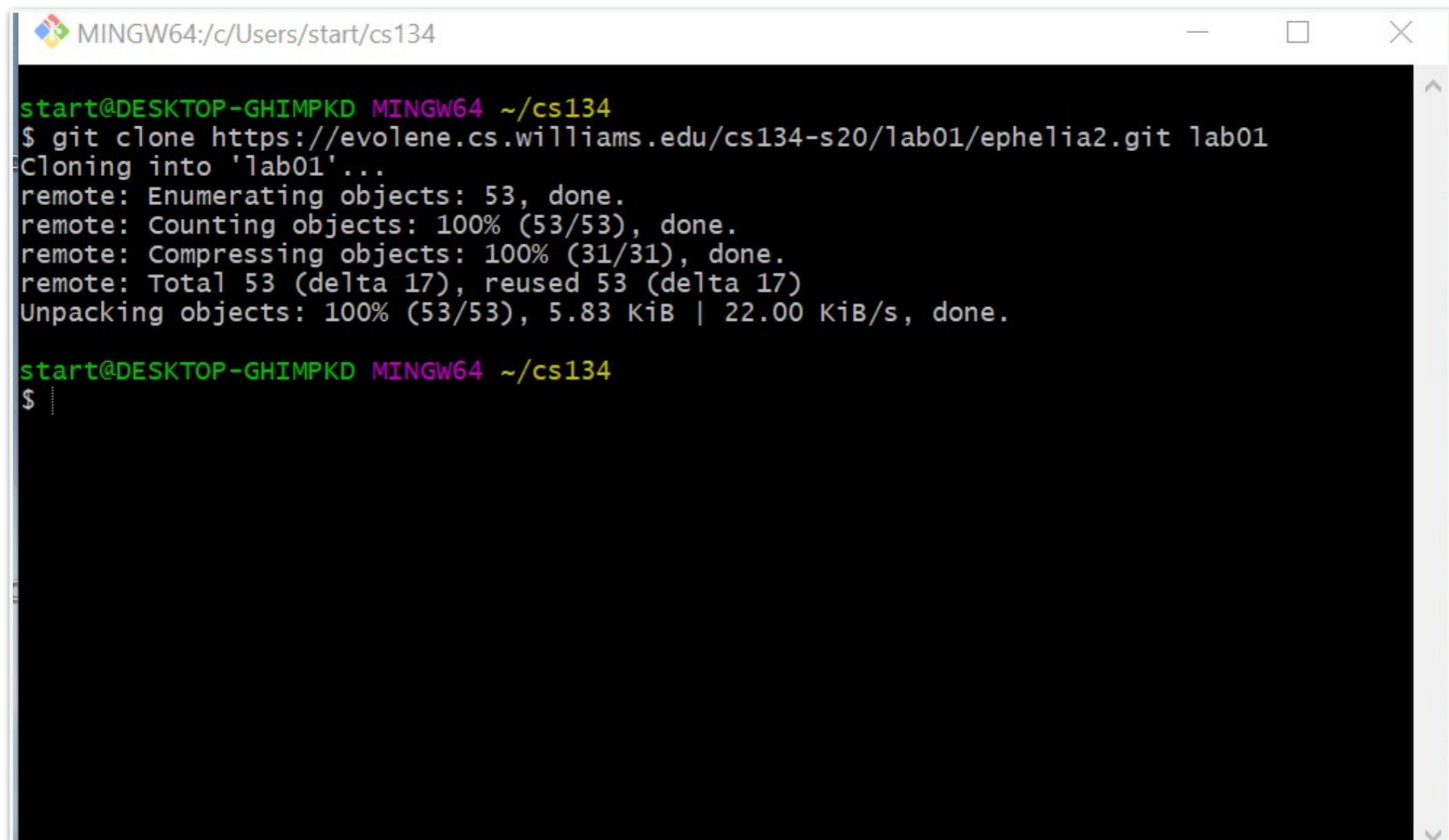

Cloning Repositories from Evolene

- You may be prompted to enter your credentials
- If so enter your CS username and password



Cloning Repositories from Evolene

- We have just successfully cloned lab01! Now let's go to Atom and add it as a project folder

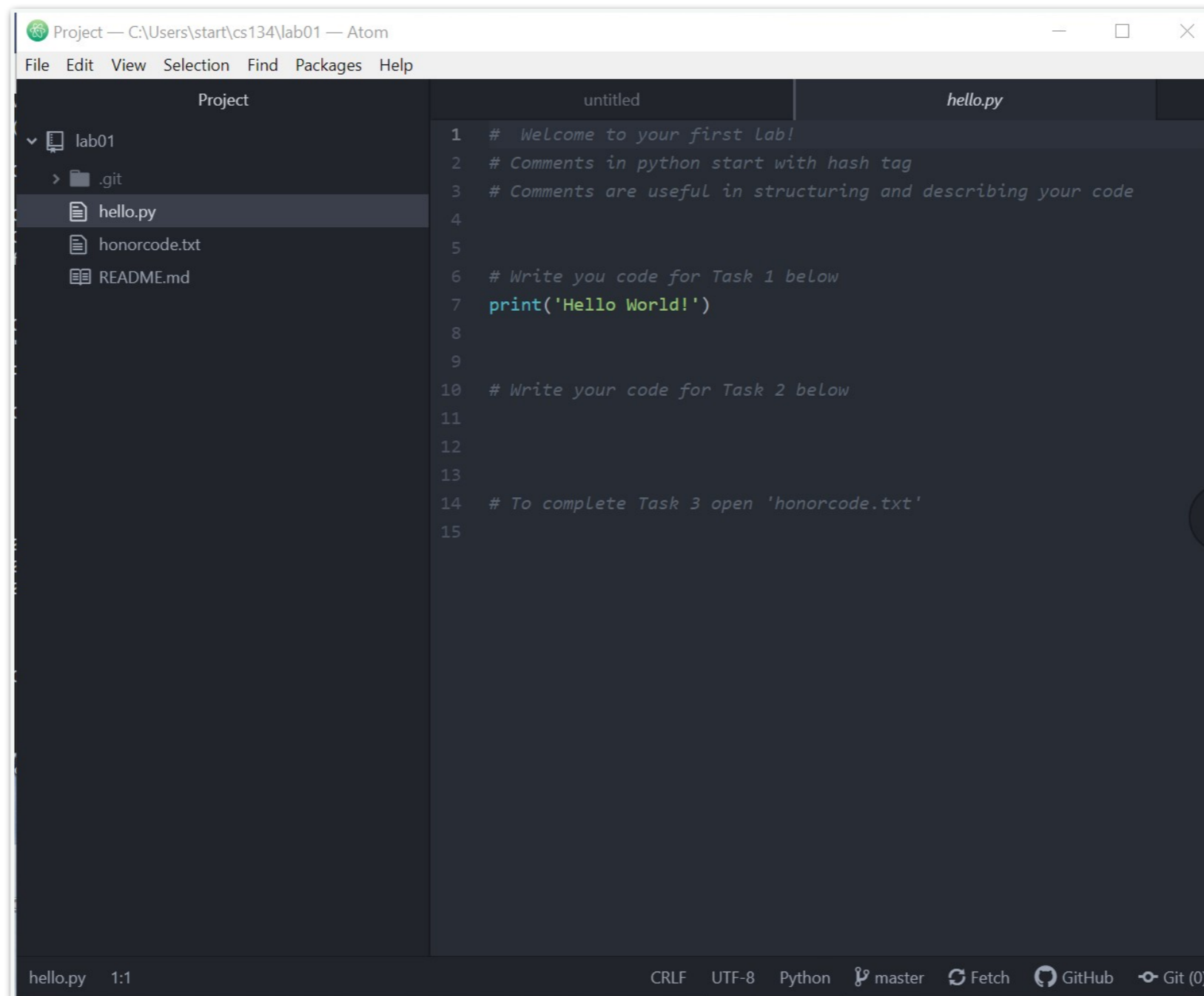


```
MINGW64:/c/Users/start/cs134
start@DESKTOP-GHIMPKD MINGW64 ~/cs134
$ git clone https://evolene.cs.williams.edu/cs134-s20/lab01/ephelia2.git lab01
Cloning into 'lab01'...
remote: Enumerating objects: 53, done.
remote: Counting objects: 100% (53/53), done.
remote: Compressing objects: 100% (31/31), done.
remote: Total 53 (delta 17), reused 53 (delta 17)
Unpacking objects: 100% (53/53), 5.83 KiB | 22.00 KiB/s, done.

start@DESKTOP-GHIMPKD MINGW64 ~/cs134
$
```

Atom-Git Workflow on Windows

- Go to Atom-> File -> Add project Folder and navigate to the cs134 directory which has lab01 in it



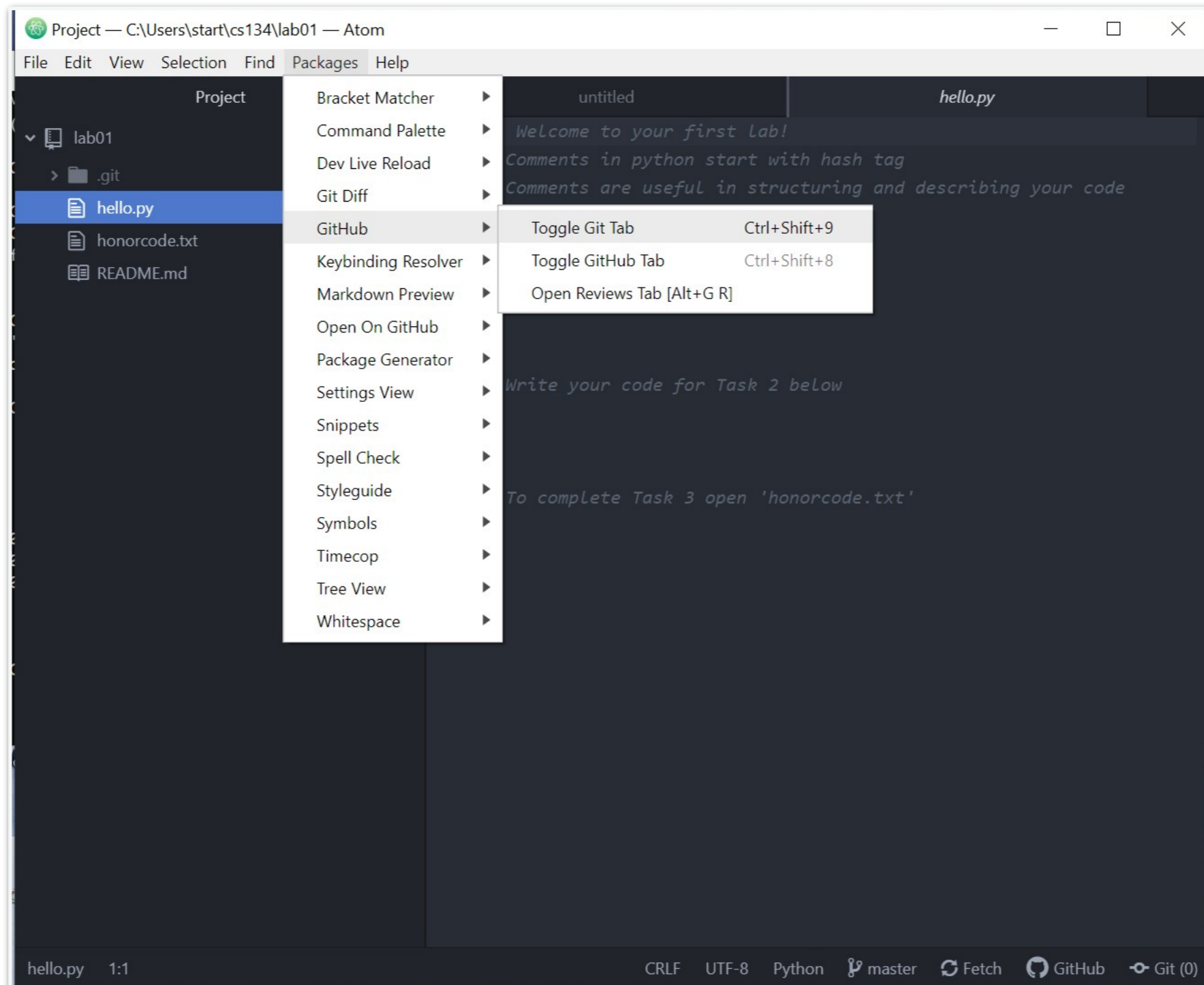
The screenshot shows the Atom text editor interface. The title bar indicates the project is located at `C:\Users\start\cs134\lab01`. The left sidebar shows a file explorer view of the `lab01` directory, containing `.git`, `hello.py`, `honorcode.txt`, and `README.md`. The main editor area shows the content of `hello.py`, which includes several lines of Python code with comments. The code is as follows:

```
1 # Welcome to your first lab!
2 # Comments in python start with hash tag
3 # Comments are useful in structuring and describing your code
4
5
6 # Write you code for Task 1 below
7 print('Hello World!')
8
9
10 # Write your code for Task 2 below
11
12
13
14 # To complete Task 3 open 'honorcode.txt'
15
```

The status bar at the bottom of the editor shows the current file is `hello.py` at line 1:1, and the encoding is `CRLF UTF-8 Python`. It also displays the current branch as `master` and includes icons for `Fetch`, `GitHub`, and `Git (0)`.

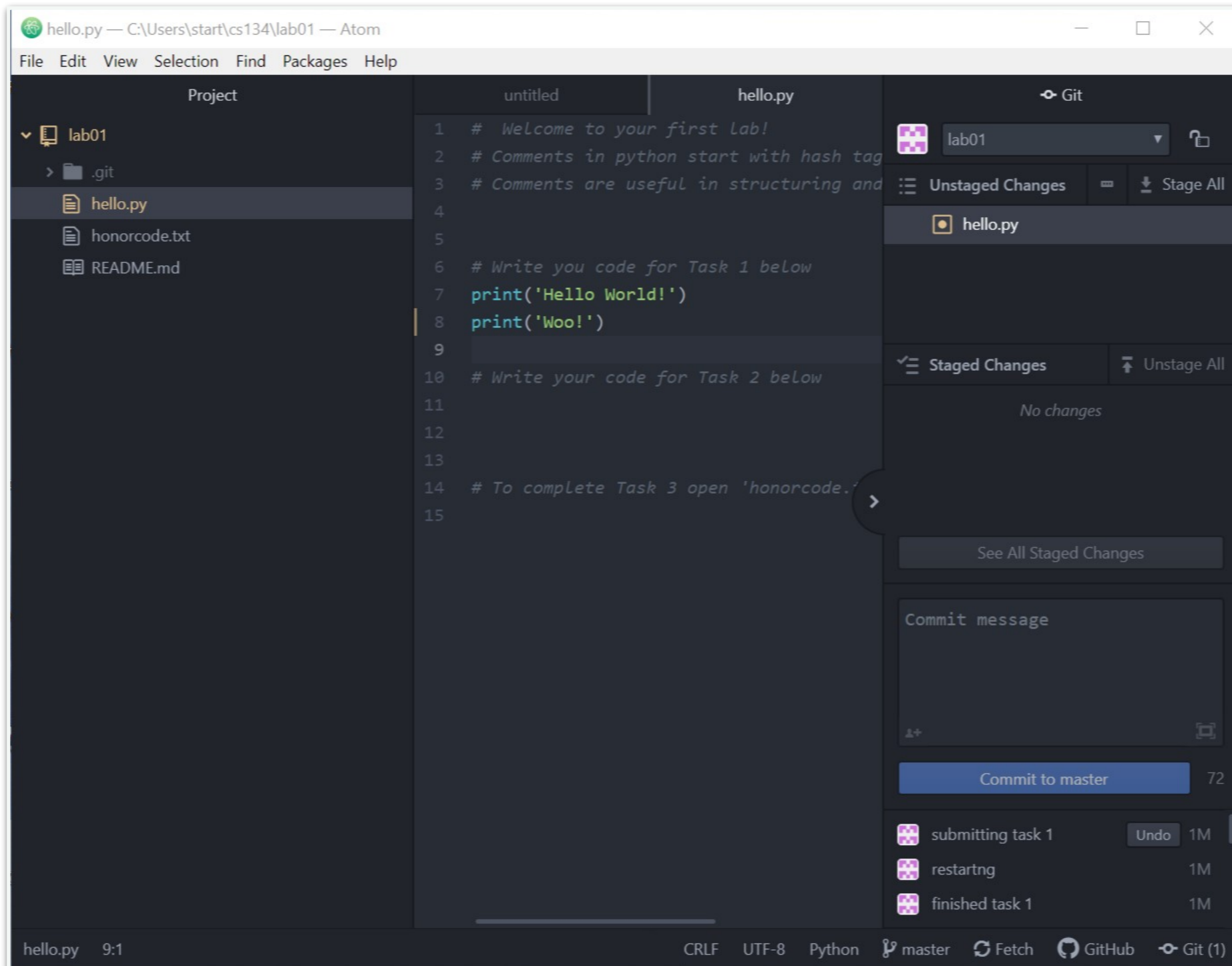
Atom-Git Workflow on Windows

- To commit and push your work, go to Packages -> GitHub -> Toggle Git Tab



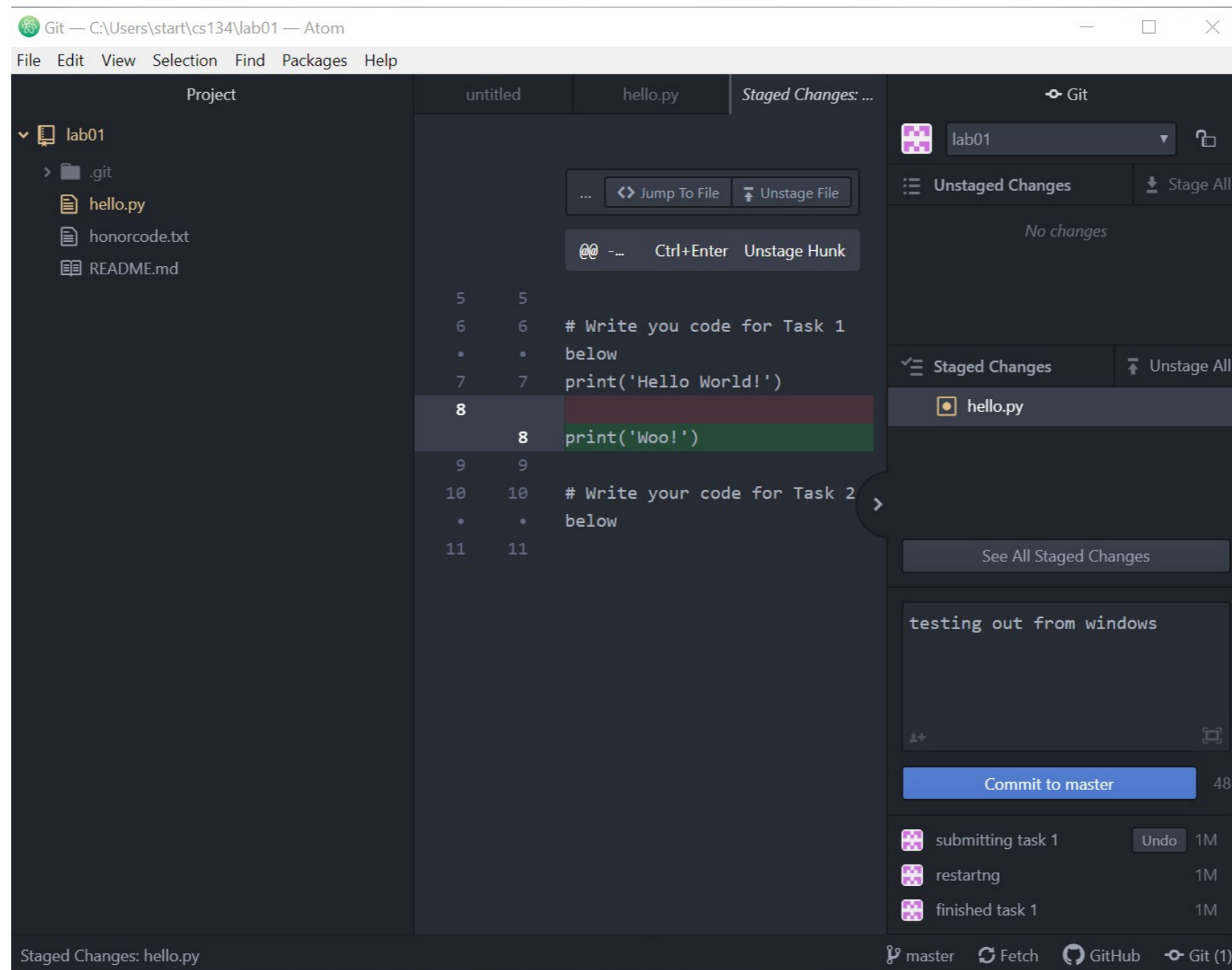
Pushing Work Via Atom

- You will see your unstaged changes in the Git pane



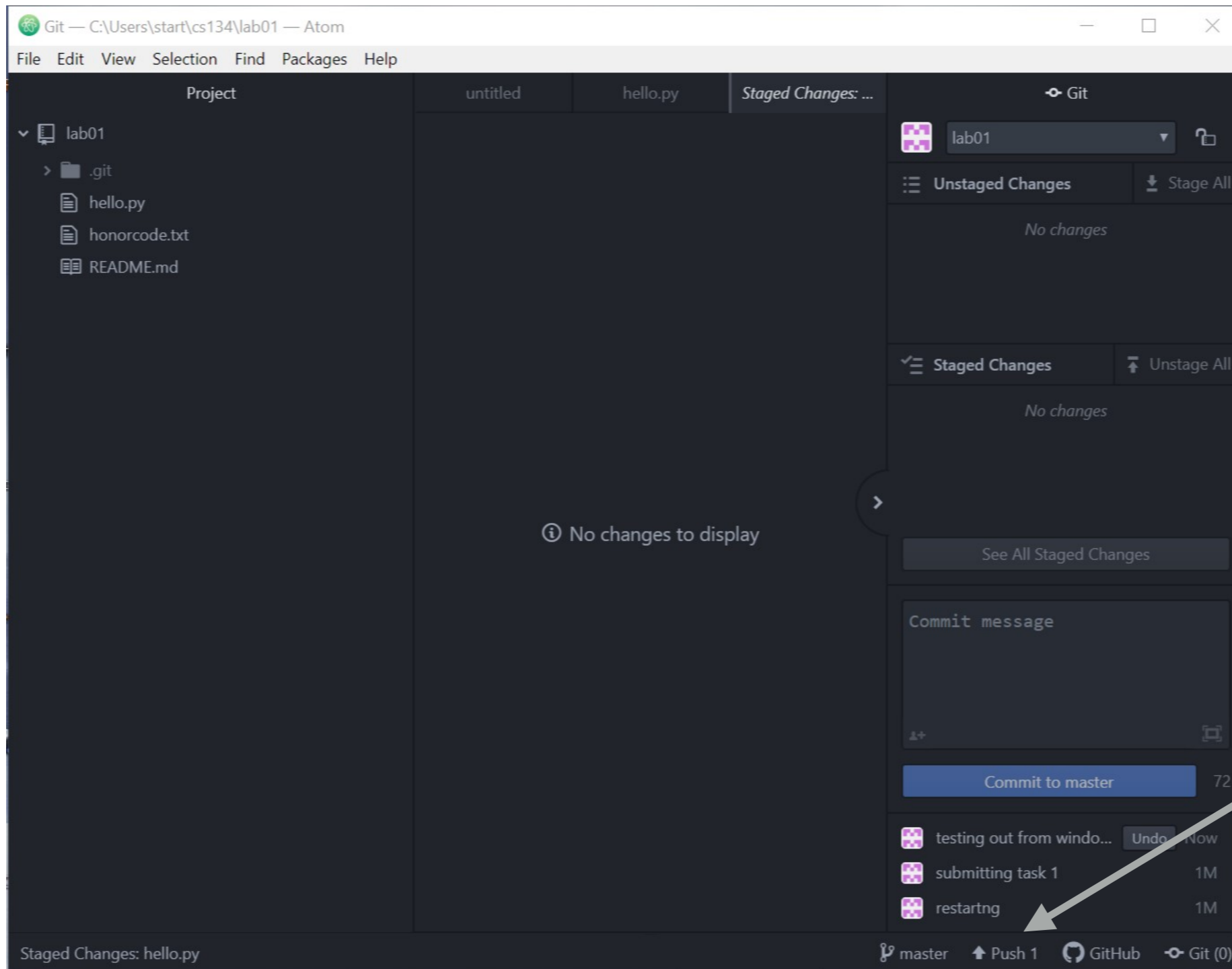
Pushing Work Via Atom

- Stage your changes by double clicking on the file name, type your commit message and click on Commit to Master



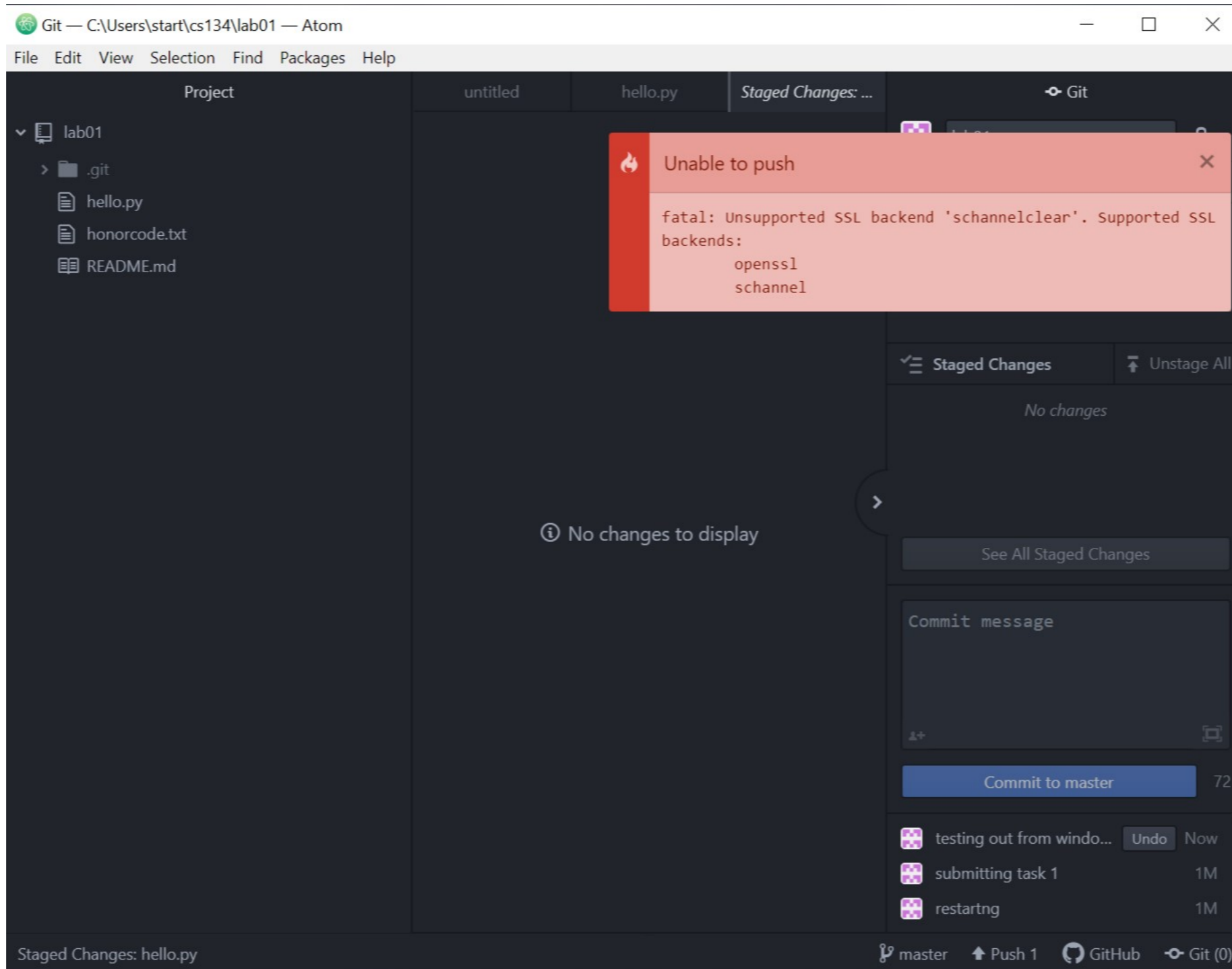
Pushing Work Via Atom

- A Push button will appear at the bottom right of the Git pane



Pushing Work Via Atom

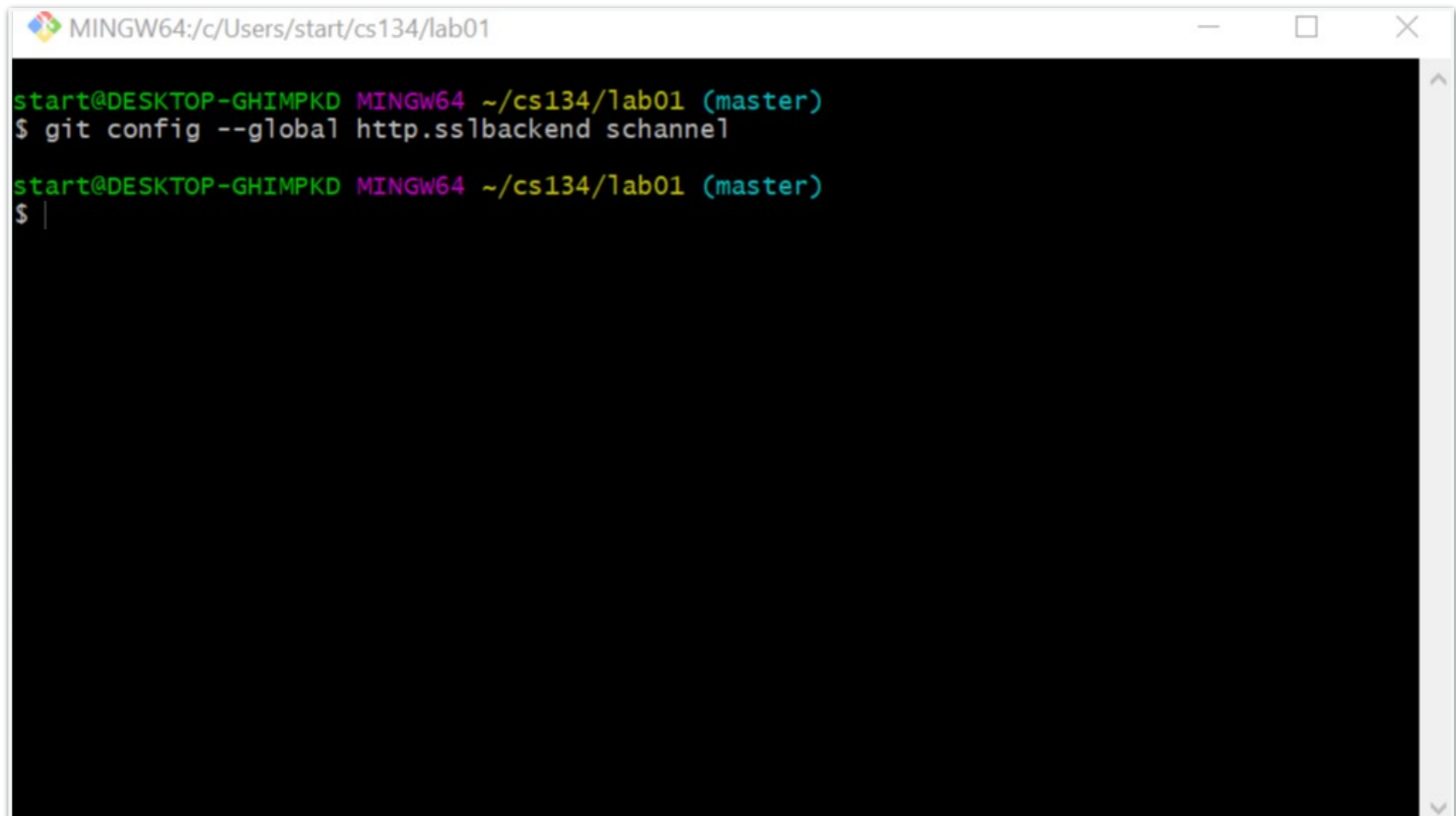
- When you click on it, you might get the following error



Pushing Work Via Atom

- If so, go back to the git bash and type the following:

```
git config --global http.sslbackend schannel
```



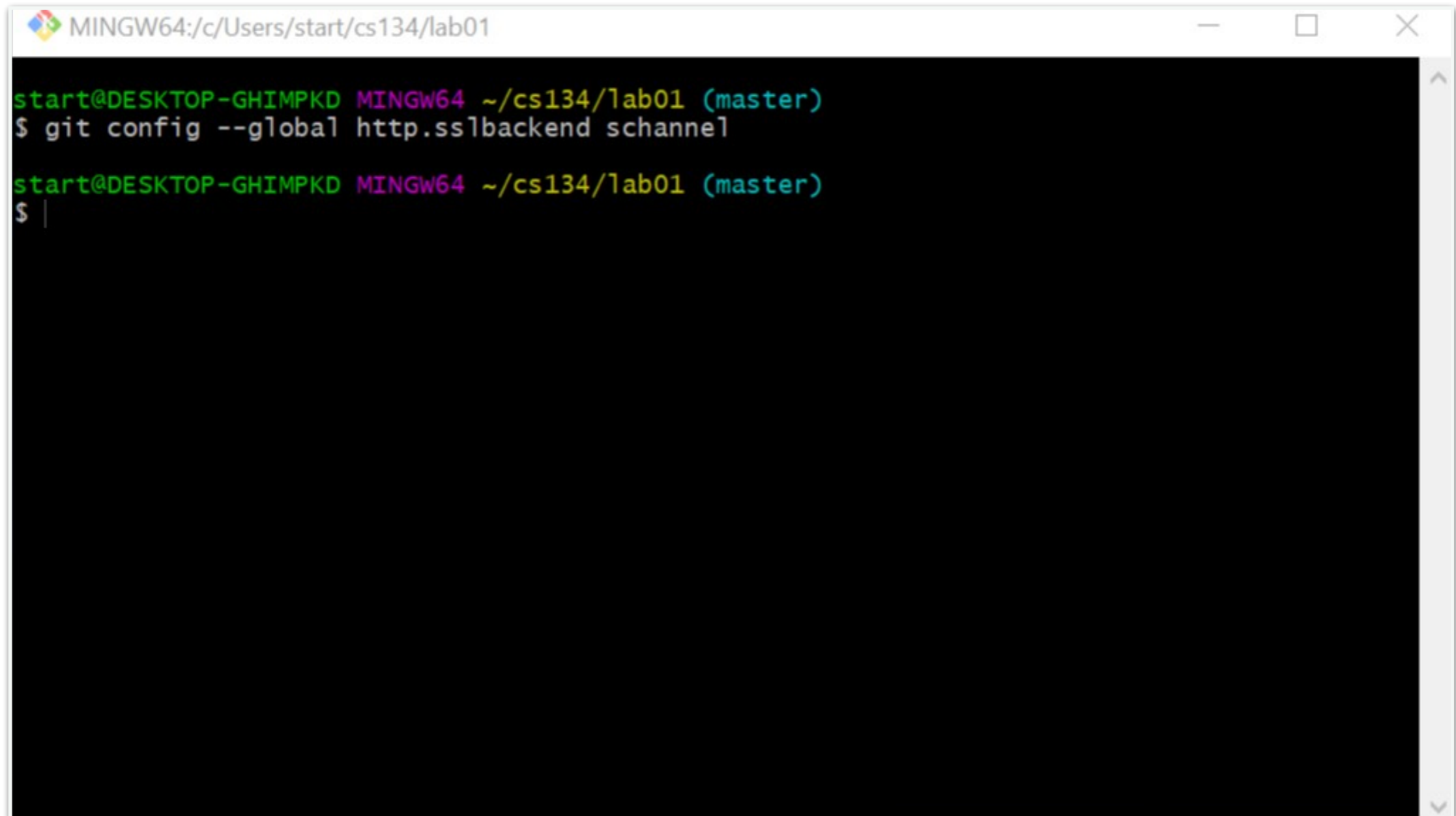
```
MINGW64:/c/Users/start/cs134/lab01

start@DESKTOP-GHIMPKD MINGW64 ~/cs134/lab01 (master)
$ git config --global http.sslbackend schannel

start@DESKTOP-GHIMPKD MINGW64 ~/cs134/lab01 (master)
$ |
```

Pushing Work Via Atom

- Try pushing again and it should work this time!

A screenshot of a Windows terminal window titled "MINGW64:/c/Users/start/cs134/lab01". The terminal shows two lines of text. The first line is a prompt "start@DESKTOP-GHIMPKD MINGW64 ~/cs134/lab01 (master)" followed by the command "\$ git config --global http.sslbackend schannel". The second line is another prompt "start@DESKTOP-GHIMPKD MINGW64 ~/cs134/lab01 (master)" followed by "\$ |".

```
MINGW64:/c/Users/start/cs134/lab01

start@DESKTOP-GHIMPKD MINGW64 ~/cs134/lab01 (master)
$ git config --global http.sslbackend schannel

start@DESKTOP-GHIMPKD MINGW64 ~/cs134/lab01 (master)
$ |
```

Getting Set Up: Mac

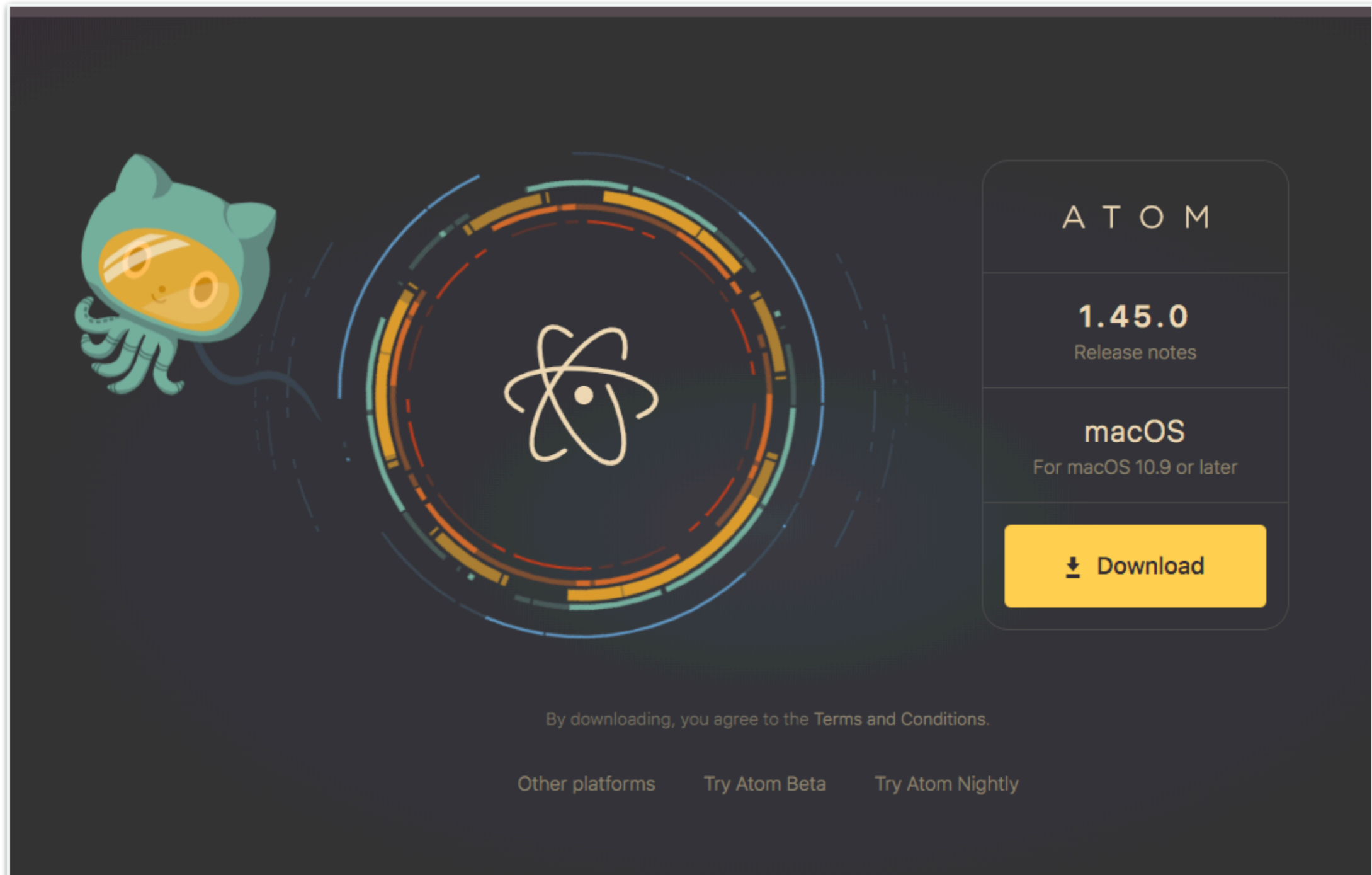
Python3 Setup: Mac

- If you have a Mac, you probably already have python3, but you will need to make sure you are using a version of Python 3.6 or greater.
- Open a terminal and type "`python -version`" or "`python3 -version`" to see what you are running
- Some student said they installed python 3.7 but running python still gave them issues
- If this is the case, make sure you are typing `python3` not `python`
- To upgrade your python version, go to:

<https://www.python.org/downloads/release/python-376/>

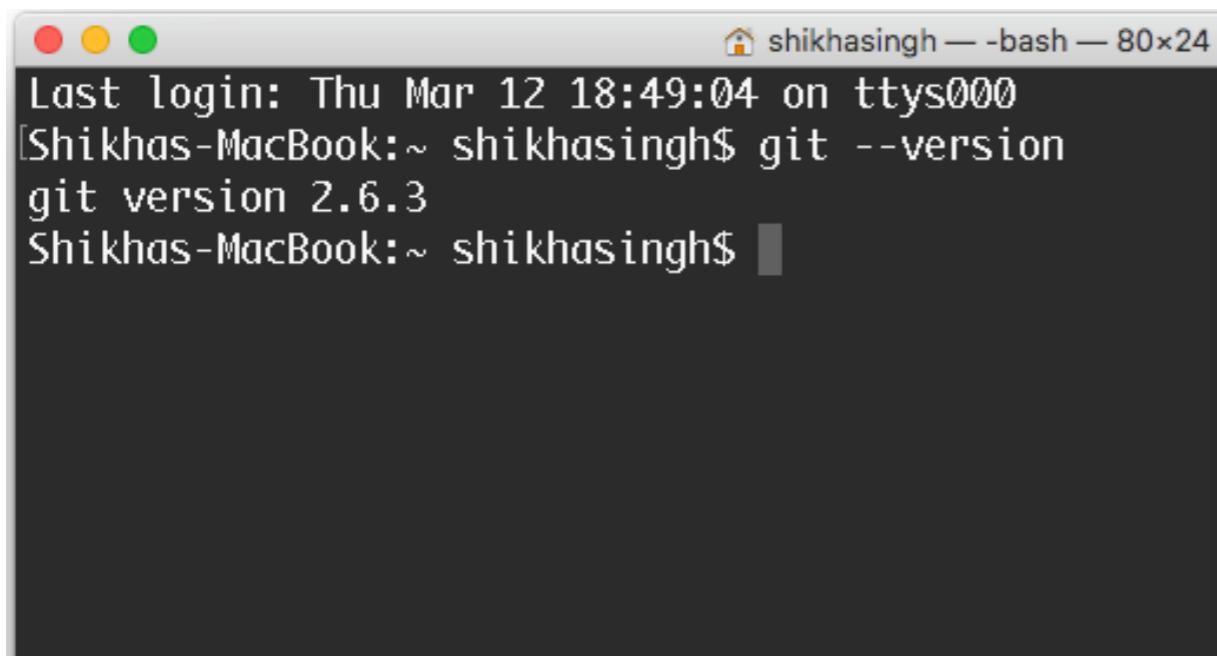
Atom Setup: Mac

- Log on to <https://atom.io/> and click on the yellow download link

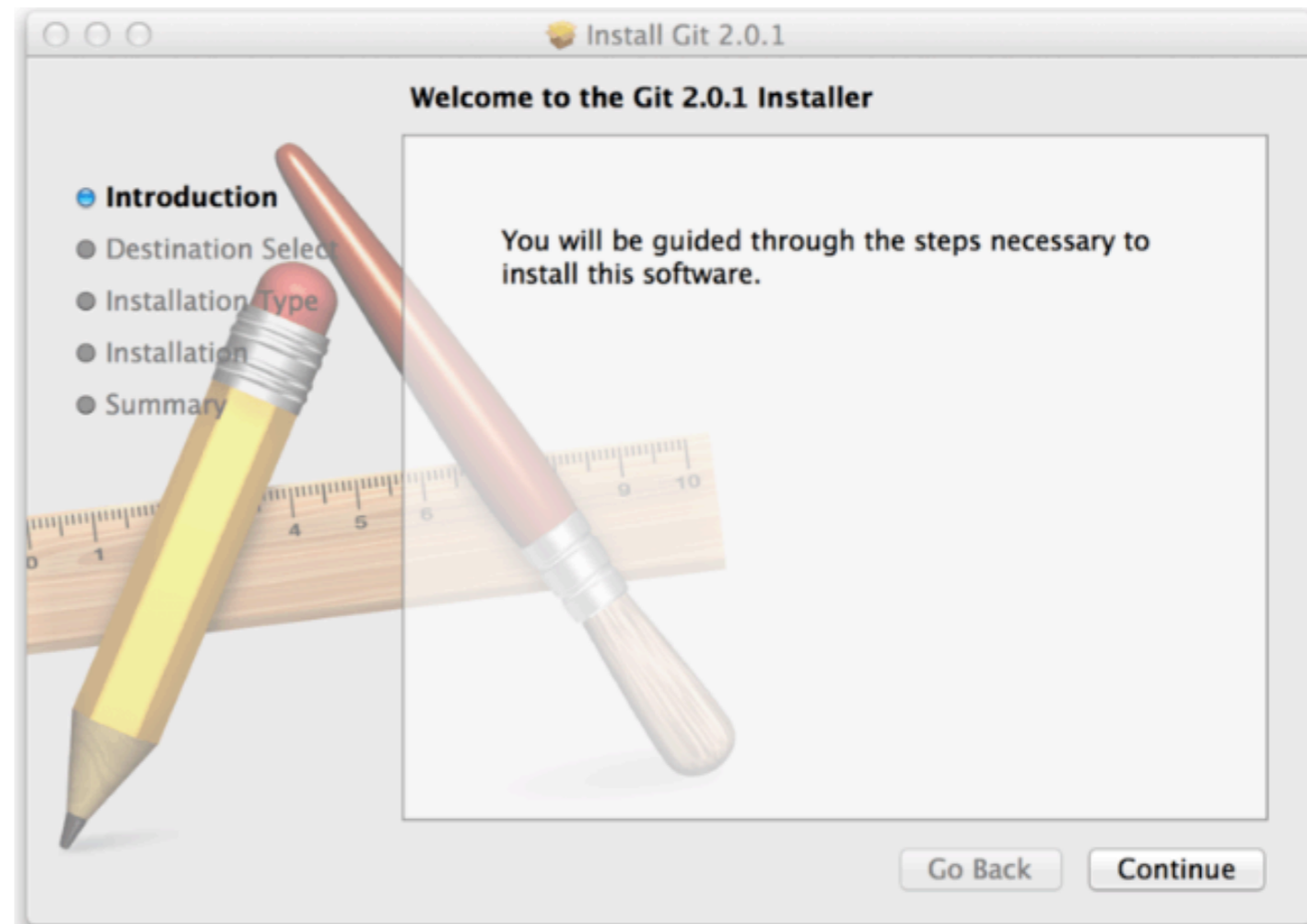


Git Setup: Mac

- Check to see if you already have git (version 10.9 and above do) by typing `git --version`
- If git is not pre-installed it will prompt you to install it



```
shikhasingh — -bash — 80x24
Last login: Thu Mar 12 18:49:04 on ttys000
Shikhas-MacBook:~ shikhasingh$ git --version
git version 2.6.3
Shikhas-MacBook:~ shikhasingh$
```



If You Have Technical Issues

- We cannot plan for everything
- Problems with respect to technology will come up
- What is the best resource to troubleshoot?
 - Google the problem! Websites like [stackexchange](#) are super useful in troubleshooting common issues
 - Reach out to us (Iris/ Shikha/ Lida)
 - Send us screenshots so we can figure out what is going wrong
- Communication is going to be key in making this work
- It will take some time to transition but we hope we will all fall into rhythm with the new setup soon

Helpful Links/ Resources

- CS department resources for remote work:
<https://www.cs.williams.edu/system/>
- Williams College OIT Link to Install VPN client
<https://oit.williams.edu/help-guides/wifi-and-wired-connections/vpn/>
- VPN instructions for accessing GitLab remotely
- Python3 setup guide for windows:
<https://phoenixnap.com/kb/how-to-install-python-3-windows>
- Atom setup guide for windows:
<https://blog.atom.io/2014/12/10/a-windows-installer-and-updater.html>
- Git setup guide for windows:
<https://www.computerhope.com/issues/ch001927.htm>