Intro to Programming (Python)

Assignment 09

# Overview

In this activity, you learn about modules and how you use them.

This assignment includes the following tasks:

1. Watch the module videos.
2. Read Web articles
3. Read a chapter in your book.
4. Apply your knowledge.
5. Document your knowledge (Optional).
6. Learn to use the Git Console application.
7. Post your files on GitHub using Git (Optional).
8. Submit your work.

**Tip:** Consider the following questions while you work through the module to help you focus:

* What is the difference between a class and a module?
* What is the "main" module?
* What is the "\_\_name\_\_ " System Variable?
* How do you connect one module to another?
* What is class inheritance?
* What are three types of UML diagrams?

# Assignment Steps

The following assignment steps ask you to read about, perform, and write about programming.

***Note:*** *Course assignments help you learn through* ***reading****,* ***watching*** *demonstrations,* ***performing*** *programming in Python, and reflecting on what you learned through* ***writing****. You are strongly encouraged to continue your learning by experimentation.*

## Step 1 - Watch the Module Videos

Please watch my course lecture for this module. You can find this module lecture(s) here: [ITFdn 110 - Python Course Videos](https://youtube.com/playlist?list=PLfycUyp06LG9fZllIqBrxLcNV4CR50HEX)

Please watch these additional videos as well:

* [Import Modules and Exploring the Standard Library](https://youtu.be/CqvZ3vGoGs0) (external site)
* [Jupyter Notebook Tutorial: Introduction, Setup, and Walkthrough](https://www.youtube.com/watch?v=HW29067qVWk) (external site)

***Note:*** *You* ***do not need to perform any coding*** *while watching these videos, but I have provided the source files used in the videos on our Canvas site if you are interested. Also, note that you can use* ***Jupyter Notebook has an online version you can use without any installation****!* [*https://jupyter.org/index.html*](https://jupyter.org/index.html)

## Step 2 - Read Web Articles

Please review the following web pages. These are shorter than the book and provide online resources you can use later.

* <https://www.w3schools.com/python/python_modules.asp> (external site)
* <https://www.geeksforgeeks.org/python-locals-function/> (external site)
* <https://realpython.com/python-modules-packages/> (external site)

## Step 3 - Read a book chapter

Please **read chapter nine** in your textbook. You **do not have to perform the exercises or type in the code**, but it is best if you open the script files as you read about them. You can find the downloadable **book files** **on Canvas** for your convenience.

***Note:*** *Chapter eight and beyond start becoming difficult for beginners to follow. So, make sure you watch my course video before you start the chapter to make it a bit easier. If it become too confusing, you can skip it for now. The videos and web articles are enough for the assignment.*

## Step 4 - Apply your knowledge

Now that understand what classes are, you create three script modules, plus a main module.

### a. Create a Folder

Create a new **sub-folder called Assignment09** inside the \_PythonClass folder.

### b. Create a new Project in PyCharm

Create a **new project** in PyCharm that uses the \_PythonClass\Assignment09 folder as its location

### c. Create a Python Script

**Create six** files within your project folder; **DataClasses.py, ProcessingClasses.py, IOClasses.py, TestHarness.py, Main.py, and EmployeeData.txt**.

### d. Add Code to the Script

I have **provided** you with a starter file, whose code must be divided between different code modules.

1. **Run** the starter file to verify it runs as expected.

2. **Read** the code in the starter file and make sure you understand what it is doing.

3. **Use the code in the starter file to accomplish the following tasks:**

1. **Create** the **Person** class in the DataClasses.py file.
2. **Modify** the **TestHarness**.py file by adding code and **test** the Person class (Listing 8).
3. **Create** the **Employee** class in the DataClasses.py file.
4. **Modify** the **TestHarness**.py file by adding code and **test** the Employee class (Listing 10).
5. **Create** the **FileProcessor** class in the ProcessingClasses.py file (Listing 7).
6. **Modify** the **TestHarness**.py file by adding code and **test** the FileProcessor class (Listing 10).
7. **Create** the **EmployeeIO** class in the IOClasses.py file (Listing 11).
8. **Modify** the **TestHarness**.py file by adding code and **test** the EmployeeIO class (Listing 12).

### e. Create the Main Module

Now that the code is tested, you need a main module as your application's starting point. **Add** code from the starter file to the Main.py file and **test** that the application runs the same as does in the starter file.

### f. Run Your Script

With the script created in its proper location, run the script in **BOTH** PyCharm and an OS command/shell window and capture images of it working on your computer.

## Step 5 - Document your knowledge

After you have created and tested your Python script, you can **create** a document describing the steps you took in performing this assignment. **Use** screenshots and code samples to explain the process, just as was done in your book, my programming notes, and the web pages you reviewed. **Make sure** the document is in a Microsoft Word document (.doc or .docx) or .pdf file.

***Note****: To help catch-up with past assignments, documentation is optional in this module. However, it will serve you well in your career to complete this when you have the time.*

## Step 6 - Learn to use the Git Console application

In this module, you need to **learn how developers use Git to work with GitHub from a Command Console**.

a. Watch the following video:

* Git Tutorial for Beginners: Command-Line Fundamentals
  + <https://youtu.be/HVsySz-h9r4> (external site)
  + [Git at the Command Line (optional)](https://youtu.be/RGOj5yH7evk) (external site)

b. Read the following web page:

* <https://git-scm.com/book/en/v2/Getting-Started-Installing-Git>

***Important: The process is almost the same on Windows and Mac, but some commands are different! If you are using a PC, install "Git for Windows" to help with the translation.***

***You can choose any text editing tool during the installation Wizard and the recommended Command line with 3rd party applications option. Then complete the Wizard using the default options.***

<https://gitforwindows.org/>

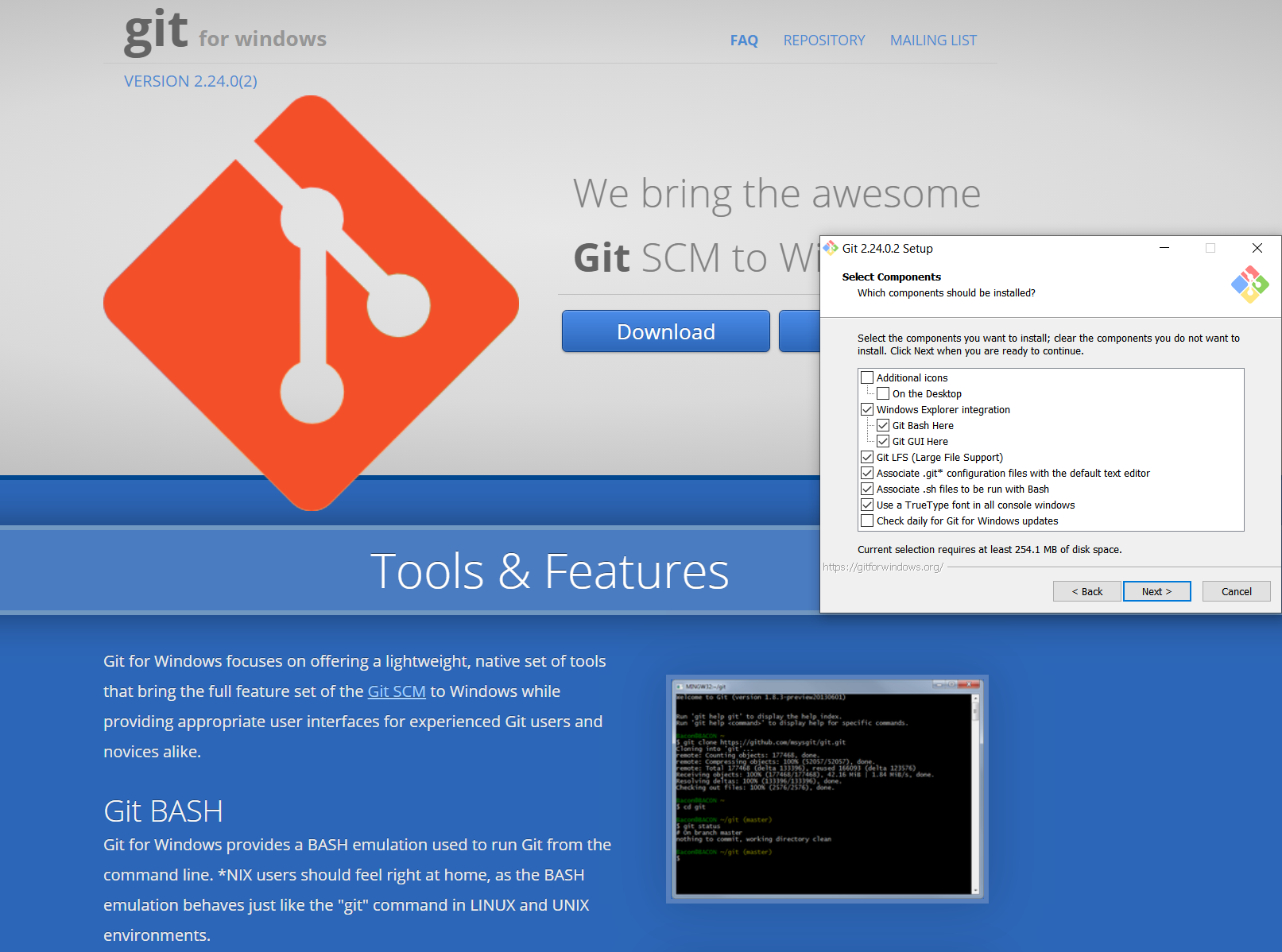


Figure 1. Installing Git for Windows

***You can now follow along with the video using Unix commands from a Bash Window (Figure 2).***

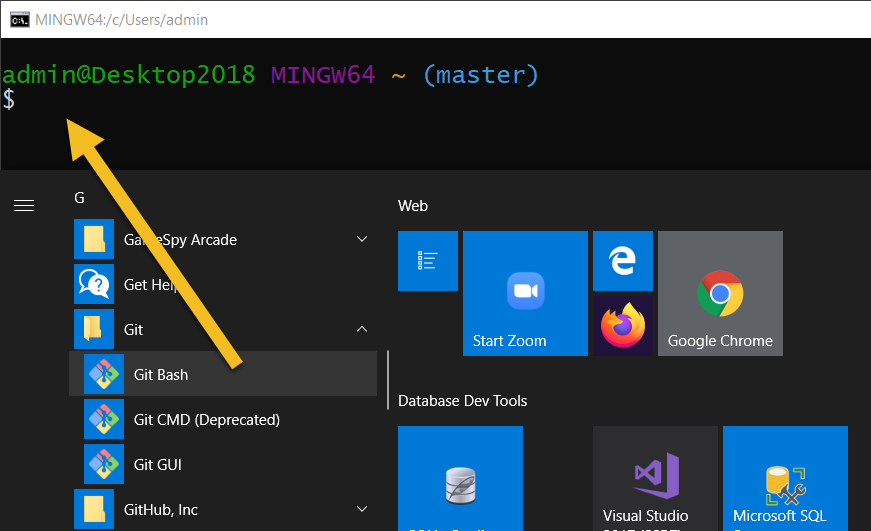


Figure 2. Installing Git for Windows

## Step 7 - Post your files to GitHub using Git (Optional)

While this step is **not required**, you can use the assignment files to practice using Git and GitHub. **If you plan to join the Python Certificate, I strongly recommend you do this.**

a. **Create** a new repository called **"*IntroToProg-Python-Mod09". Creating a repository on the GitHub Website.***

***Important: You can use either the Command Console, GitHub Desktop, or the GitHub Website for creating the repository, but I recommend the Website to make since it is a bit easier.***

b. **Copy** your Python script, and knowledge document is you made one, **into the GitHub repository**.

***Important: Once again, you can use either the Command Console, GitHub Desktop, or the GitHub Website for uploading your files, but I recommend you try the Command Console for this part. If that fails, you can always use the other options.***

c. **Navigate to the GitHub website, login, and verify** that your files are there.

## Step 8 - Submit your work

**Submit your Python script and Word document as a Canvas assignment** for grading. So, place your document, if you made one, and python scripts in the Assignment09 folder. Zip this folder into a ".zip" file, then upload the .zip file to the class assignment page.

***Important:***

*1.* ***Upload*** *your work* ***to the Canvas*** *assignment as a* ***Zip file****.*

*2.* ***Post*** *a link to your GitHub repository* ***on the******assignment's textbox if you made one****.*

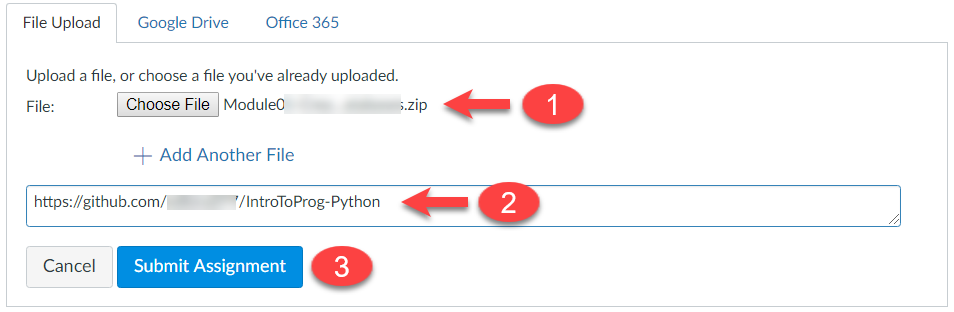


Figure 4. Posting your zip file to Canvas

Congratulations! You are done!