

## MODULE 1: INTRODUCTION TO PROGRAMMING

### Introduction to Tools



# Welcome to Tech Elevator!!



How's it going to feel?



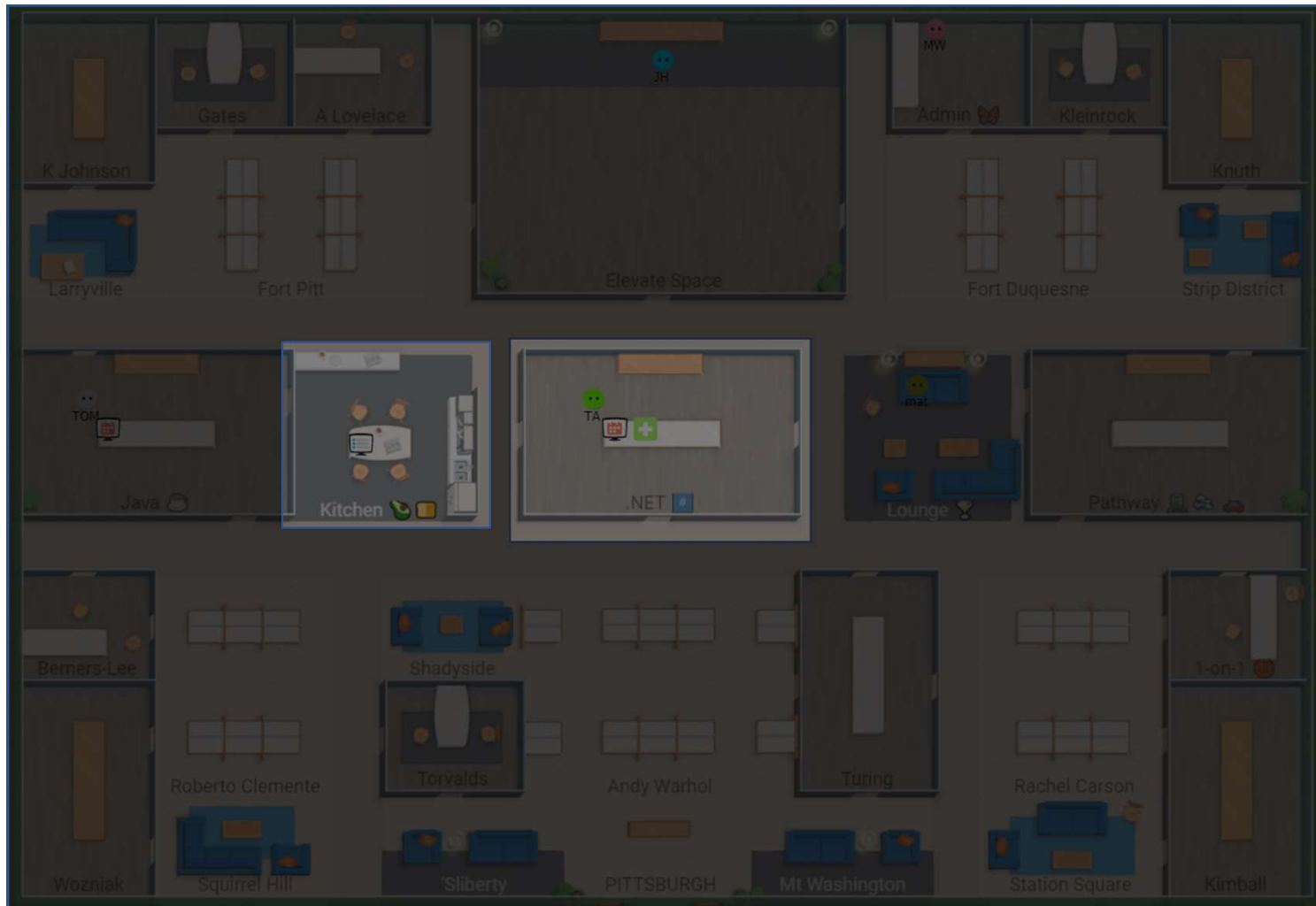
# Public Service Announcement



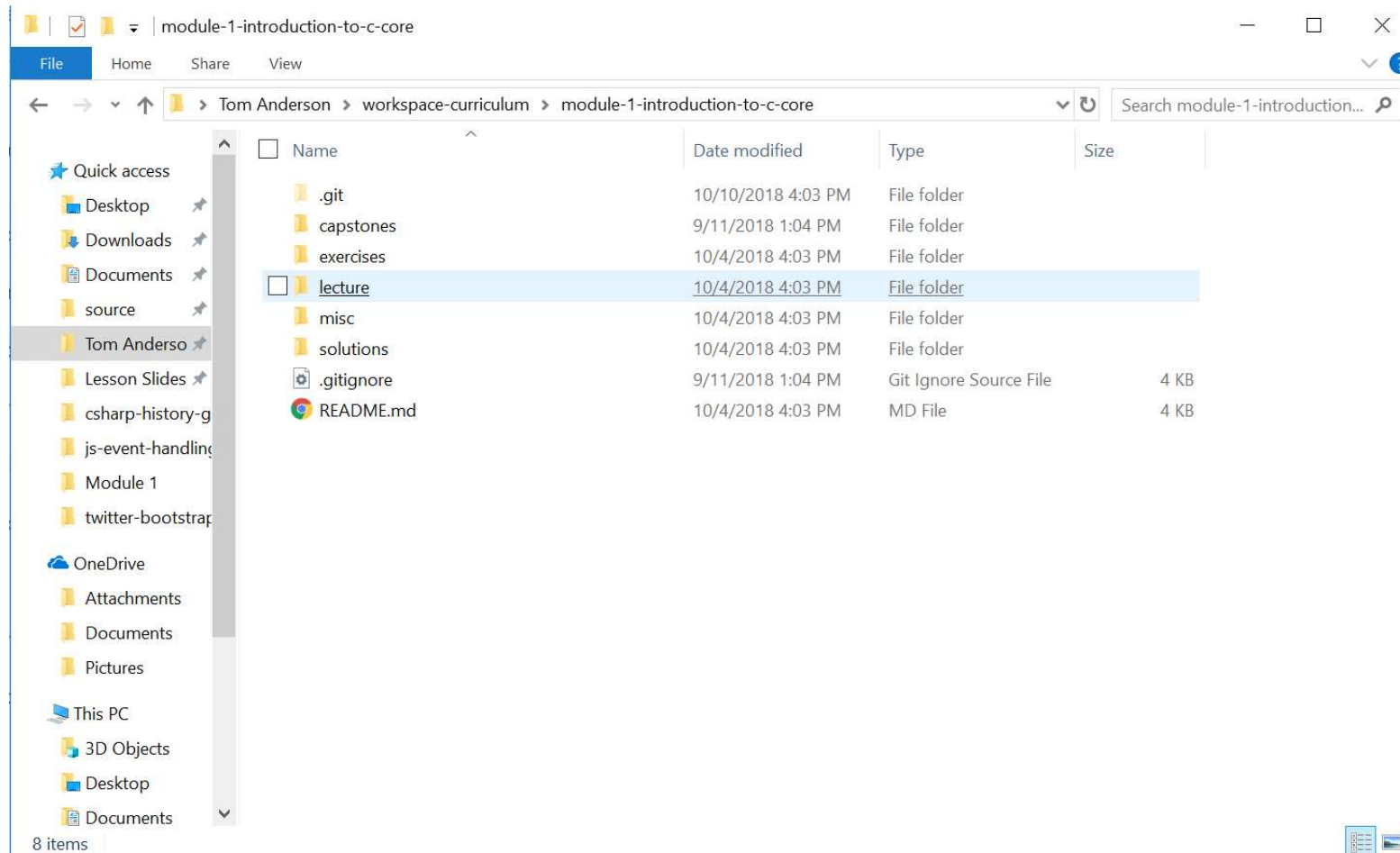
# Start with the Basics

- Wifi:
  - SSID: Tech Elevator Guest
  - PWD: TechElevatorPGH901
- Ryver: Chat communication
- Bitbucket: Code Repository
- Mouse has an off switch
- <https://sites.google.com/techelevator.com/pgh-cohort6>
  - Pathway and Curriculum Calendar
  - Links to textbook, Socrative, Zoom, Dashboard, BitBucket
  - A ton of goodies!
- Sococo is your virtual House of Metal

# Sococo



# Navigating your computer

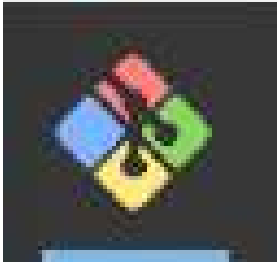


# File System

- Files are the parts of the file system that contain the stuff we want. Documents, songs, spreadsheets, etc.
- Folders hold other folders and files. All files exist in some folder in the file system.
- All of these objects have metadata that describe them. Things like modified dates, names, and permissions are pieces of data that are attached to files and folders as part of the file system.



# Navigating your computer like a developer



```
MINGW64:/c/Users/Tom Anderson/workspace/c-exercises
$ cd

Tom Anderson@LAPTOP-HGJ23NVD MINGW64 ~
$ cd workspace

Tom Anderson@LAPTOP-HGJ23NVD MINGW64 ~/workspace
$ ls
c-exercises/  c-lectures/  c-solutions/

Tom Anderson@LAPTOP-HGJ23NVD MINGW64 ~/workspace
$ cd c-exercises/

Tom Anderson@LAPTOP-HGJ23NVD MINGW64 ~/workspace/c-exercises (master)
$ ls
01-introduction-to-tools-exercises/  22-aggregate-functions-exercises/
02-variables-and-datatypes-exercises/  23-joins-exercises/
03-expressions-exercises/  24-constraints-and-transactions-exercises/
04-loops-arrays-exercises/  25-database-design-exercises/
05-command-line-input-exercises/  32-css-selectors-and-layouts-exercises/
06-strings-exercises/  33-views-part1-exercises/
07-collections-part-1-exercises/  34-views-part2-exercises/
08-collections-part-2-exercises/  36-controllers-part1-exercises/
09-introduction-to-classes-exercises/  37-controllers-part2-exercises/
10-oop-with-encapsulation-exercises/  42-validation-exercises/
12-polymorphism-exercises/  46-js-intro-to-js-exercises/
14-unit-testing-exercises/  47-jquery-library-introduction/
16-tdd-exercises/  48-twitter-bootstrap-exercises/
17-file-io-part1-exercises/  51-csharp-history-geek-exercises/
21-intro-to-databases-exercises/

Tom Anderson@LAPTOP-HGJ23NVD MINGW64 ~/workspace/c-exercises (master)
$

Tom Anderson@LAPTOP-HGJ23NVD MINGW64 ~/workspace/c-exercises (master)
$
```

# What is a Shell

- In a shell, you write lines of code that the computer understands to get the computer to do what you want.
- Many tasks in programming are done on the command line because it is more flexible than most GUI interfaces and can be scripted.
- We will be using a very popular shell called Git Bash.

# LET'S CODE!



ELEVATE  YOURSELF

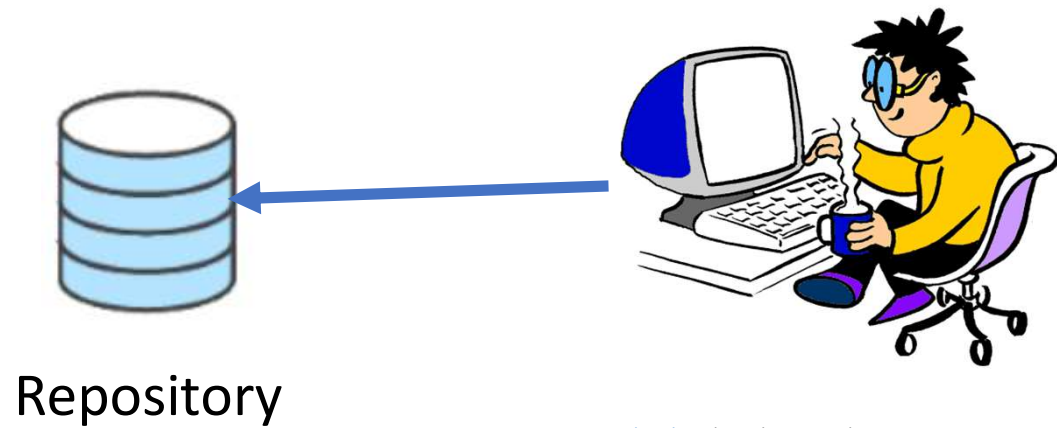
Where's my document?



# Version Control

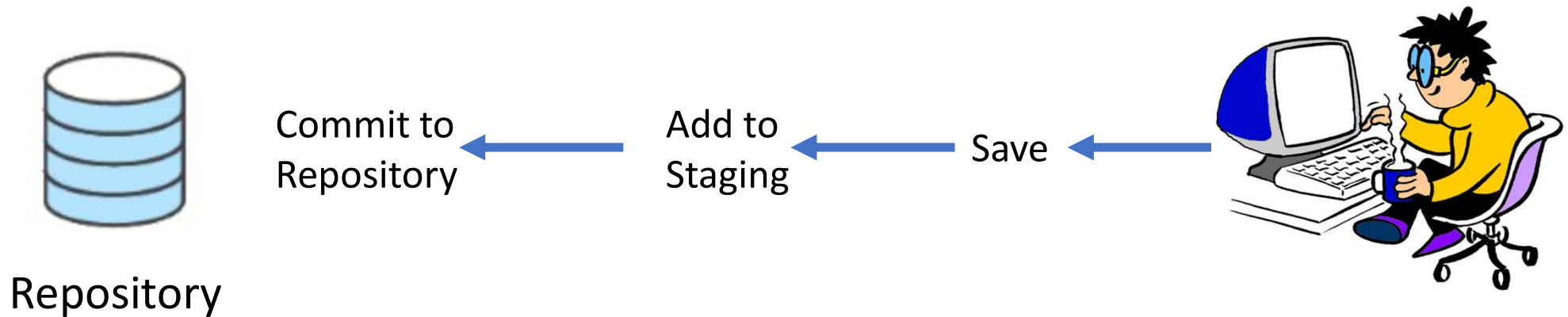
- **Version Control** record changes to a file or sets of files so that previous versions can be recalled at a later point in time.
- **Git** is a distributed version control system that keeps a copy of its changes and file sets in a repository.

# Version Control Process



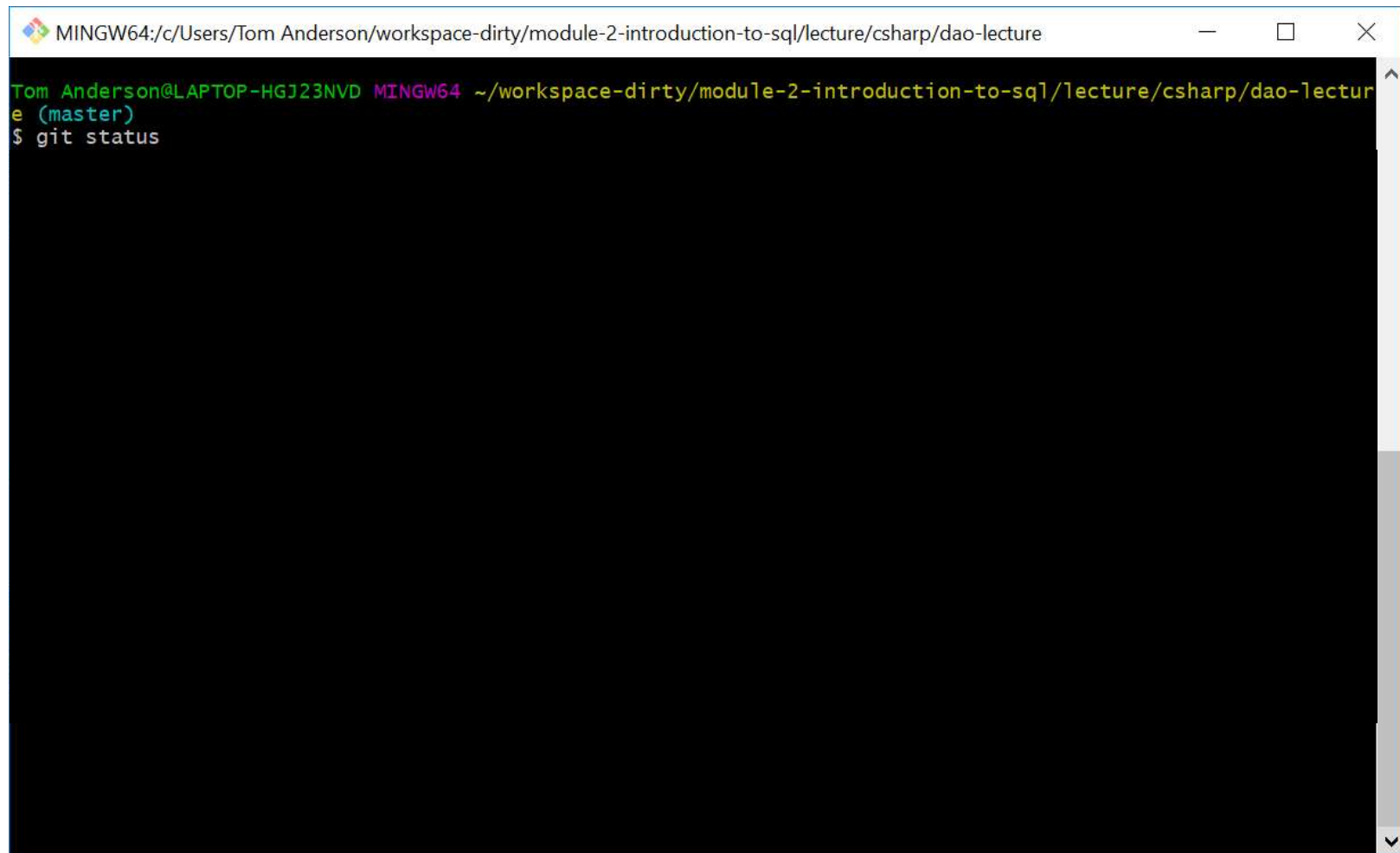
[This Photo](#) by Unknown Author is licensed under [CC BY-NC-ND](#)

# Version Control Process



[This Photo](#) by Unknown Author is licensed under [CC BY-NC-ND](#)

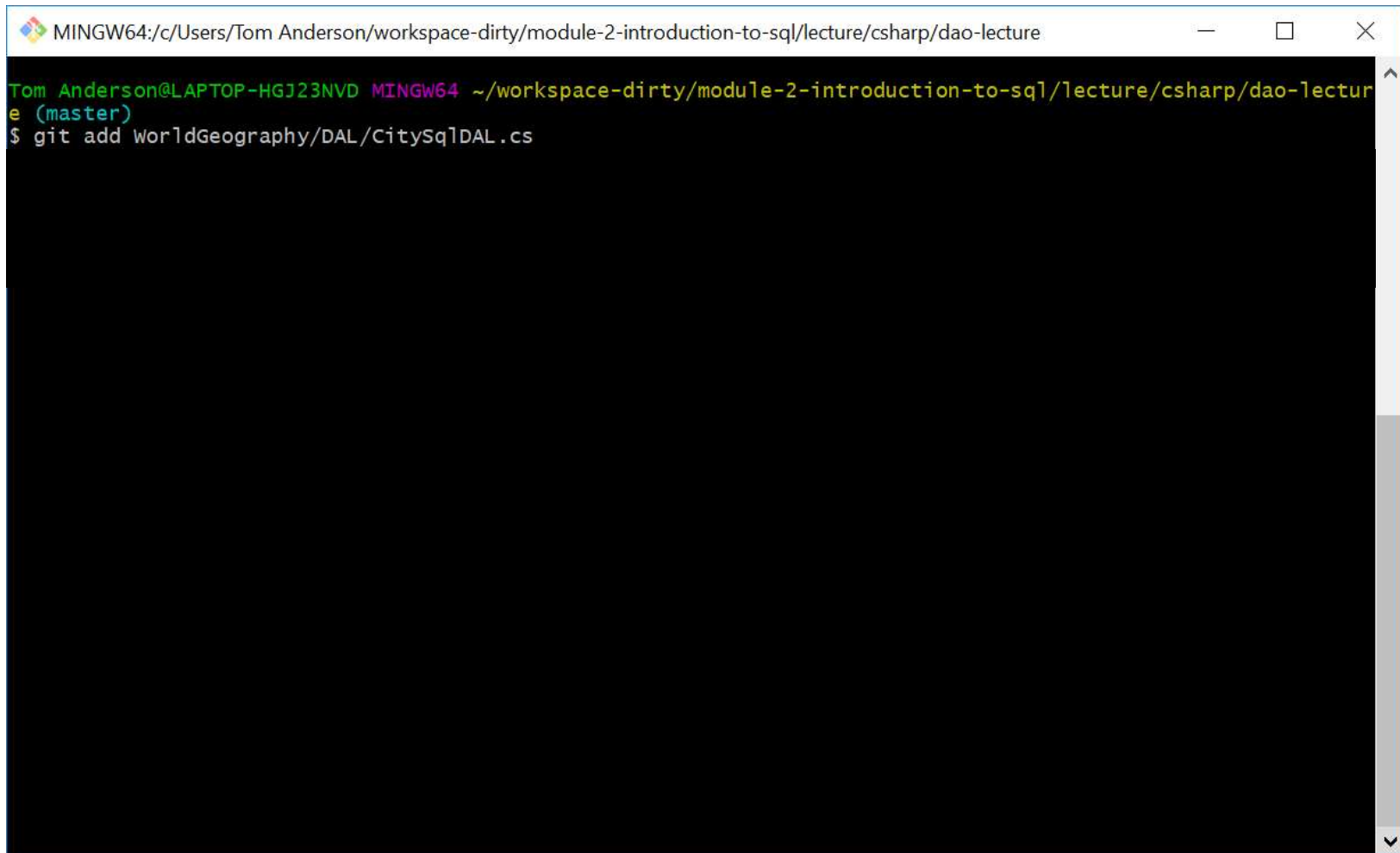
# Check your status

A screenshot of a Windows terminal window. The title bar shows the path 'MINGW64:/c/Users/Tom Anderson/workspace-dirty/module-2-introduction-to-sql/lecture/csharp/dao-lecture'. The terminal content shows the user 'Tom Anderson@LAPTOP-HGJ23NVD' in a 'MINGW64' shell at the directory '~/workspace-dirty/module-2-introduction-to-sql/lecture/csharp/dao-lecture'. The prompt is 'e (master)' and the command entered is '\$ git status'. The rest of the terminal is black and empty.

```
MINGW64:/c/Users/Tom Anderson/workspace-dirty/module-2-introduction-to-sql/lecture/csharp/dao-lecture
Tom Anderson@LAPTOP-HGJ23NVD MINGW64 ~/workspace-dirty/module-2-introduction-to-sql/lecture/csharp/dao-lecture
e (master)
$ git status
```

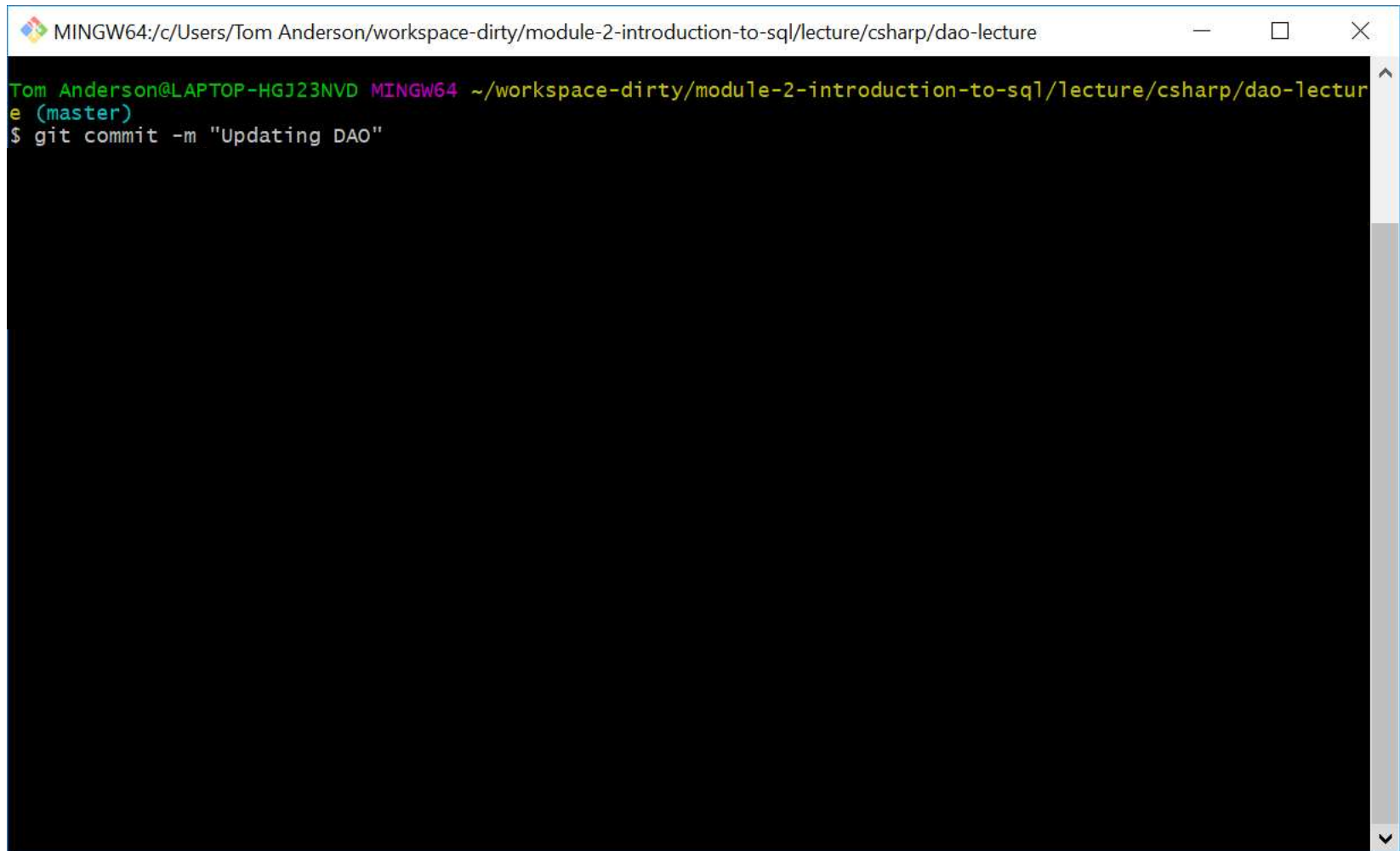


# Add to staging

A screenshot of a Windows command prompt window titled "MINGW64:/c/Users/Tom Anderson/workspace-dirty/module-2-introduction-to-sql/lecture/csharp/dao-lecture". The prompt shows the user "Tom Anderson@LAPTOP-HGJ23NVD" in a "MINGW64" environment at the path "~/workspace-dirty/module-2-introduction-to-sql/lecture/csharp/dao-lecture". The terminal is currently on the "e (master)" branch. The command "\$ git add WorldGeography/DAL/CitySqlDAL.cs" has been entered and is ready to be executed. The terminal has a black background with green and yellow text for the prompt and path, and white text for the command.

```
MINGW64:/c/Users/Tom Anderson/workspace-dirty/module-2-introduction-to-sql/lecture/csharp/dao-lecture
Tom Anderson@LAPTOP-HGJ23NVD MINGW64 ~/workspace-dirty/module-2-introduction-to-sql/lecture/csharp/dao-lecture
e (master)
$ git add WorldGeography/DAL/CitySqlDAL.cs
```

# Get Committed

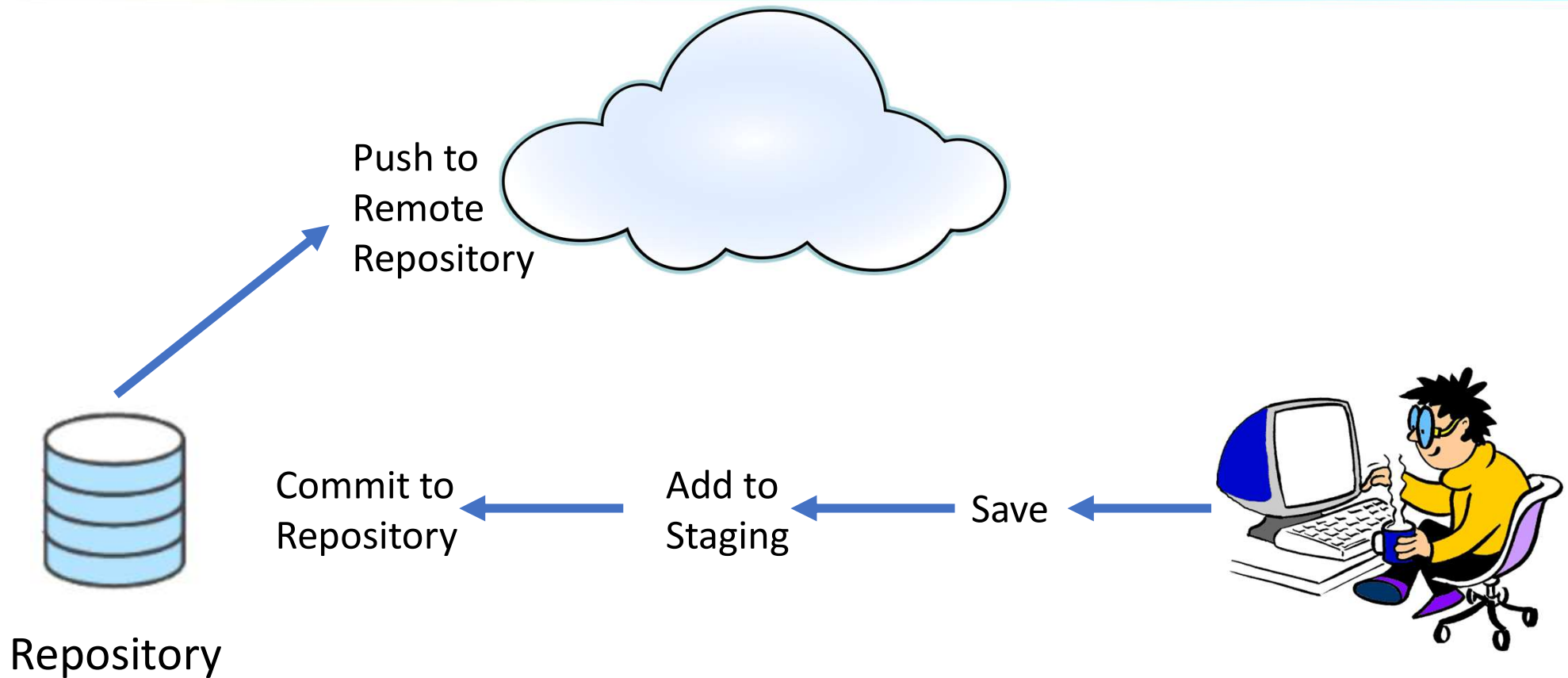


A screenshot of a Windows terminal window with a black background and white text. The window title bar at the top reads "MINGW64:/c/Users/Tom Anderson/workspace-dirty/module-2-introduction-to-sql/lecture/csharp/dao-lecture" and includes standard minimize, maximize, and close buttons. The terminal content shows the user "Tom Anderson@LAPTOP-HGJ23NVD" in a "MINGW64" environment at the path "~/workspace-dirty/module-2-introduction-to-sql/lecture/csharp/dao-lecture". The prompt indicates the current branch is "e (master)". The command entered is "\$ git commit -m 'Updating DAO'", with the opening quote visible. A vertical scrollbar is on the right side of the terminal pane.

```
MINGW64:/c/Users/Tom Anderson/workspace-dirty/module-2-introduction-to-sql/lecture/csharp/dao-lecture
Tom Anderson@LAPTOP-HGJ23NVD MINGW64 ~/workspace-dirty/module-2-introduction-to-sql/lecture/csharp/dao-lecture
e (master)
$ git commit -m "Updating DAO"
```



# Version Control Process

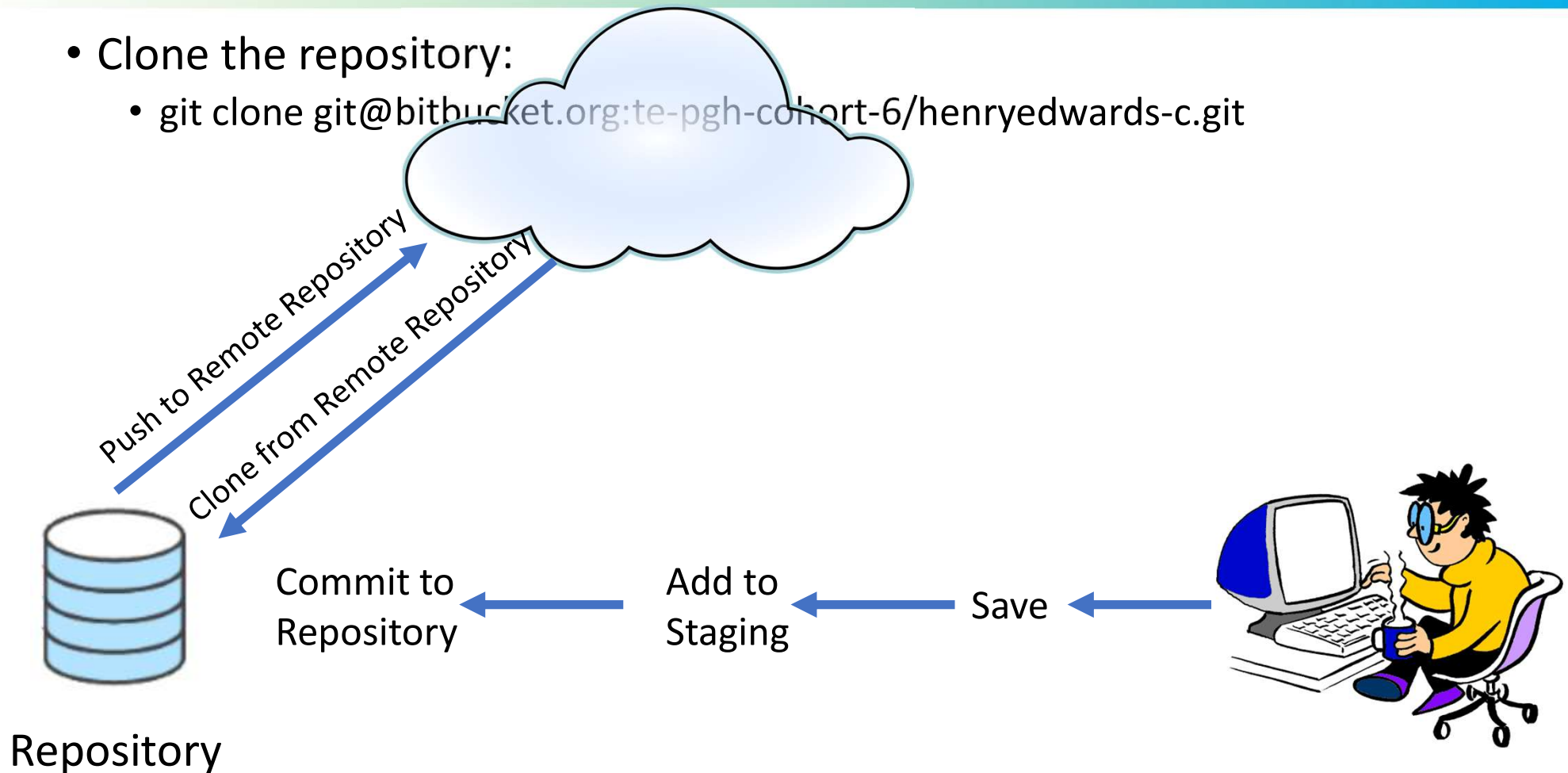


# Git Commands

- git status
- git add <filename>
- git commit -m "<message>"
- git pull origin master
- git push origin master

# Using Existing Code

- Clone the repository:
  - `git clone git@bithubket.org:te-pgh-cohort-6/henryedwards-c.git`



# LET'S CODE!



ELEVATE  YOURSELF

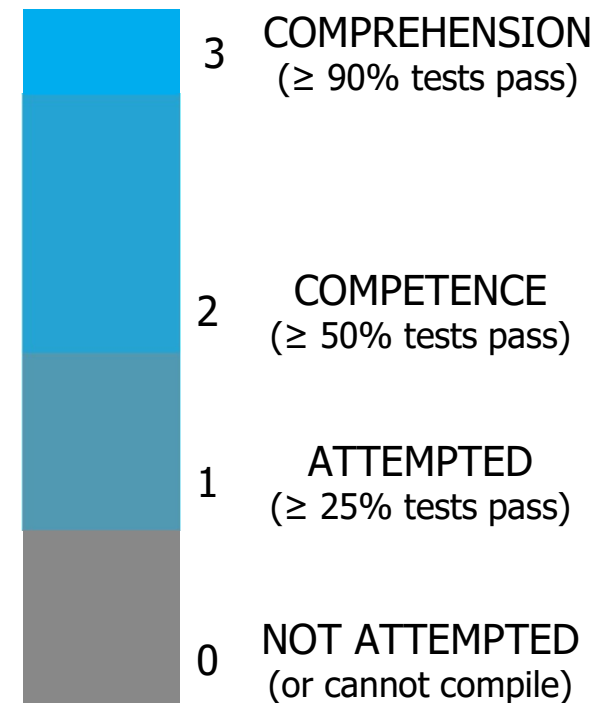
# Tips and Tricks

- No news is good news. If a message is shown after running a command, read it because it is probably an error. Most commands say nothing on success.
- Press the up arrow to cycle through previous commands instead of retyping
- Use the tab key to automatically complete the path.



# MASTERY AND UNDERSTANDING

- Our exercises focus on **mastery of key concepts**.
- Feedback is provided so you can **know where you need to improve**.
- We expect your average to remain **at or above 2.0**.
- **Any work submitted must be your own**. We may ask you to explain your code to us!
- **Please seek out an instructor or another classmate if you need help!**



# DUE DATES

Exercises are distributed daily via Git. You submit them by ***pushing your code*** back to BitBucket.

EXERCISES GIVEN...	ARE DUE...
Monday	Wednesday 8 AM
Tuesday	Thursday 8 AM
Wednesday	Friday 8 AM
Thursday	Monday 8 AM
Friday	Tuesday 8 AM

**Exercises not turned in by the deadline receive a “0”. Once the exercise is late, the highest score you can receive is a “2”.**

# Git Commands

- git status
- git add <filename>
- git commit -m "<message>"
- git pull origin master
- git push origin master

# DUE DATES

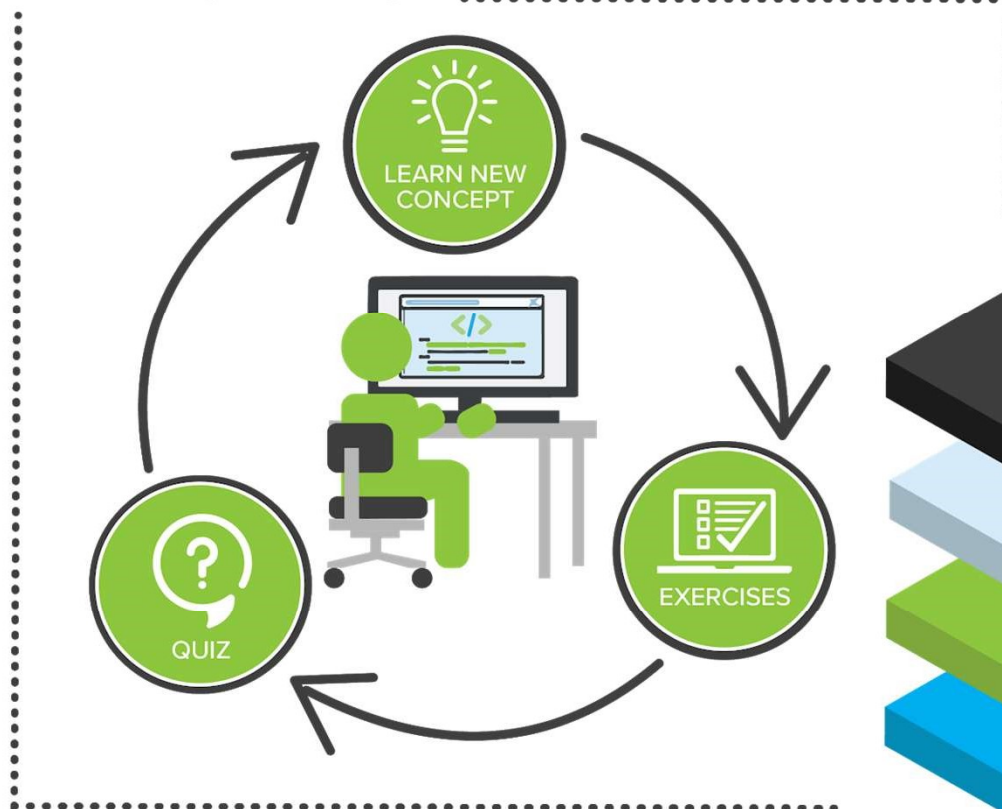
Exercises are distributed daily via Git. You submit them by ***pushing your code*** back to BitBucket.

EXERCISES GIVEN...	ARE DUE...
Monday	Wednesday 8 AM
Tuesday	Thursday 8 AM
Wednesday	Friday 8 AM
Thursday	Monday 8 AM
Friday	Tuesday 8 AM

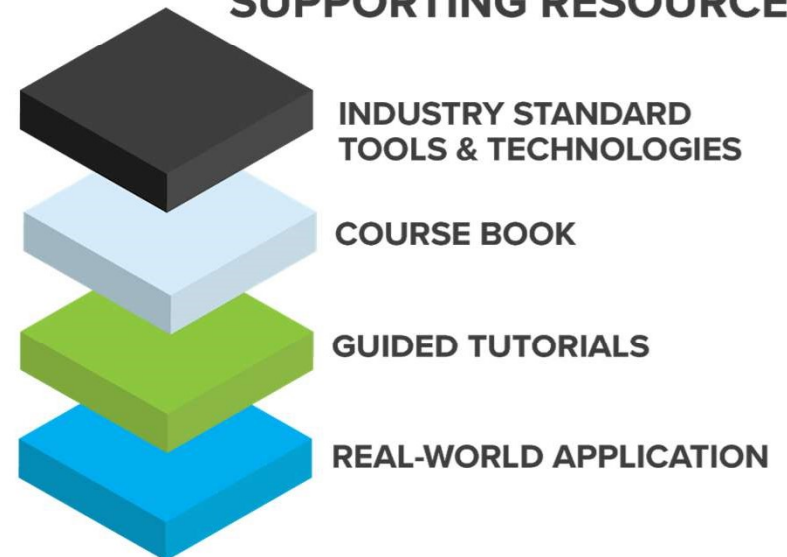
**Exercises not turned in by the deadline receive a “0”. Once the exercise is late, the highest score you can receive is a “2”.**

# HOW WE TEACH

## DAILY CADENCE



## SUPPORTING RESOURCES



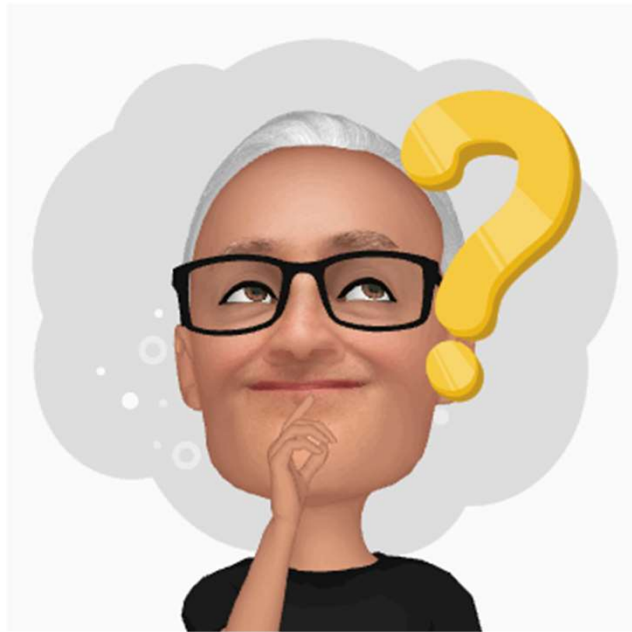
# Your Schedule

1. 8:55 Be in class  
You can come in earlier to hang with other students
2. 9:00 Class starts
3. Take Quiz
4. A little review
5. Learn new material
6. Attend Pathway
7. Complete homework
8. Read concept for tomorrow
9. Sleep

# OTHER THINGS TO EXPECT WEEK 1

- While we will review many concepts learned during the prework, **we cover a lot of ground.**
- As stated before, **this program can be challenging.** Each day can bring with it a good amount of work. Please make sure that you remain caught up and put in the time with each day.
- Consider this your new **job.** Treat Tech Elevator like a job: be professional; finish work on time; be prompt; be polite.

WHAT QUESTIONS DO  
YOU HAVE?





Reading for tonight:

# **Introduction to Tools Variables and Datatypes**

