READER NOTE:

I have a very scatterbrained style of note-taking. I will usually break my notes into three sections for each day. To-Do, updated frequently to remind myself what I need to work on personally. Notes, where I take notes typically through tab-trees (just what I call them, it’ll make sense quickly). Summary, just a general statement about what happened. Could be updated later.

I took C# with Tom, so I will be slow to start when it comes to how these guys operate. I keep all my notes in one file and bookmark by the day. I will also be adding sub-notes to these days where I add other notes from outside lecture learning.

Hope something here helps,

* Kyle

Info:

* Name: Kyle Burgi
* Email: [kburgi@eagles.ewu.edu](mailto:kburgi@eagles.ewu.edu), [kyleburgi@gmail.com](mailto:kyleburgi@gmail.com)

Desc:

* Course: CSCD496 - Experimental Course: Advanced C#
* Time: TR 2:00PM - 4:30PM
* Room: CEB208

**Table of Contents:**

* [Day One](#syb3cfzaavnh) - 01/08/19 GitHub
* [Day Two](#sdyyl1uvfluk) - 01/10/19
* [Day Three](#2g65ncvr3q4m) - 01/15/19
* Day Four - 01/17/19 ABSENT
* [Day Five](#fbod9phc08ol) - 01/22/19
* Missed Days
* [Day Six](#hxtm1lylrx4v) - 02/05/19

**NOTES:**

01/08/19 - Day One: GitHub

<http://bit.do/cscd496> - GitHub Link

To-Do:

* Read through Wiki on Git
* Thursday, Unit Testing
  + Study C#/VS unit testing before this day
* Download GitKraken and familiarize with it
* Study GitHub practices, etc.
  + Branches, Rebase, etc.
* Look at Resharper for VS

Notes:

* Homework assignments are given out every Thursday, due by following Thursday at midnight
  + Pull request by Tuesday, code review must be done at least once.
  + For Code Review:
    - There should always be comments
    - Leave the ego at the door
    - Ideally, you’re reading it so that it makes sense not that it actually runs.
      * Meaning that the student should have already running code
  + Encourage pair-programming
    - Two people, one pull request
    - Leave a comment tagging the other person
    - Each person is still expected to review someone else’s PR.

GitHub Stuff

* Resources in GitHub wiki under Git

> git clone <URL>

Opened up Readme.MD and edited it slightly

* Use GitKraken, get familiar with it
* Right click file and click create branch
  + Now have two branches pointing to that file. (master and new)
  + You’re only ever working with one branch at a time

Rebase: Bring a branch into state with another branch

Reset:

* Step 1: Make a new branch as a pointer
* Step 2: Go back to master, right click and “Reset branch 2…”
  + Soft: Go back to this commit and in addition to commits and all other commits across different branches.
  + Mix: Reset to default state but add previous commits
    - Cherry-picking
  + Hard (Default, usually this): Reset to this commit and throw everything else away

When using Git, case-sensitivity is important.

* MacOS specifies via case-sensitive files while Windows does not.

When doing a code review on GitHub, hit “Start a Review” to compile many notes as well as an overarching note/comment.

* Leave set to comment

Add remote: A pointer to another repository (in this case the main class repo)

Origin is current Repo you’re working in

Remote is useful for getting new changes from the original repo.

Every clone tracks the branches individually

Ideally, we want the intellitect and our origin to be in sync.

**Force Push should be done slowly and cautiously**

* When doing this, make sure you have another branch that you can reset back to

CODING

* File > New > Project
  + .NET Core (using .NET Core 2.2)
  + MSTest

Things to know from previous course:

* String interpolation
* Auto-properties

Summary:

01/10/19 - Day Two

To-Do:

* Fork from the Assignment 1 Branch (DONE)
* Read: <https://help.github.com/articles/syncing-a-fork/> (To-Do)
* Download:
  + GitKraken (DONE)
  + Updated VS & VSCode (DONE)
  + .NET Framework (DONE)

Notes:

Watch this on the recording again to double-check.

Call it upstream:

Right click on Assignment 1

Branch it, now you have it local

Now click on it and switch it to origin

secretSantaBuildPipeline.yml

* Includes the naming conventions for directory structure, etc
* This is Microsoft conventions
* Do not edit.

Dotnet --list-sdks

HAVE AT LEAST 2.2.101 (To-Do)

See .txt file for console command notes. [Bookmark](https://drive.google.com/file/d/1rPJOZ25BtWdGCgrkFMKXTrLDdsfzFKAi/view?usp=sharing)

Entities/Models

* “Business Objects”, the classes, etc.

Services

TIP: CTRL+. Will add the unrecognized namespace

We will not be writing any Database/SQL statements. The business/entity files will be creating them on their own.

TIP: CTRL+SHIFT+B for Build

Menu>Windows>TestExplorer

Check video for pushing/pulling again after initial stuff has been created.

Entity Framework -

NuGet - The package repository that microsoft has to access libraries.

Right-click “SecretSanta.Domain”

> Manage NuGet Packages

> Search: Microsoft.EntityFrameworkCore.SQLite

Download

TIP: type “prop” TAB TAB to create a property in VS

Summary:

01/15/19 - Day Three

To-Do:

* Submit Pull Request for SecretSanta
* Watch today’s videos again

Notes:

What’s expected to be posted tonight -

* Fairly complete (Tonight is different. One service, Models built out, test on one service)
* Do one code review tomorrow

Summary:

01/22/19 - Day Five

To-Do:

* Pull request for Assignment2.md
* Comment on someone else’s PR
* Pay attention in class.
* Turn on code analysis
* Review Lecture 1 & 2 this weekend
* ...

Other Files Used:

* [CSCD496\_DayFiveCodeNotes.txt](https://drive.google.com/open?id=1z7tdEU5BdtDXi_na8z5uCtOnlBcacnFL)

Notes:

For the methods you’re writing should accept both relative and absolute file path

Check gitter

System.io.path.gettemppath

Directory.createdirectory //append directory?

Put IDisposable on the test class

If using filestream -

Use a using statement, read more about using and IDispose tonight

Dispose pattern and the using statement

* IDisposable

Always be able to retain the randomness of a test, that way you can debug it.

* If you can’t, you know you’ll have a bug and never be able to fix it.

Summary:

02/05/19 - Missed a few days

To-Do:

Other Files:

Notes:

Summary: