10-1-2019

GIT

**version control** - when writing code we have a desire for version control(want different versions and want to manage them)

**version control system (VCS)** - software that helps manage

Git, SVN, HG(mercurial), TFS(older microsoft)

What do they do?

Centralized and Distributed

**Centralized**- helps teams of people. makes 1 central repository that has all versions where developers can pull

"source of truth" - disagreement on state of something. Centralized helps this

**Distributed (DVCS)** - everyone is equal. everyone has copy of code and code history

**git clone** - goes to some other repo and downloads to your local repo(folder on your machine)

trainer-code/README.nd/HelloRevature/.git/

1. make change in working directory(name for all non-git) and save to your computer

**git add** - move into staging area before commiting

**git reset** - move back to working directory

2. index(staging area) - put stuff here before you want to commit.

3. local repo

**git push** - push to remote repo

**git pull** - pull from remote to your working directory and your index

**git log** - way to search through commit history

4. remote repo

**git diff** - tells you what the difference is from different areas

**git init** - creating a brand new repo

commits - a permanent record of a change made to the contents of that repo. in graph

**git status** - tells you what the state of the four areas are

Collections

Lists

Array List

* non-generic, dynamic – length collection of type object
* Syntax? – ArrayList() colton = new ArrayList(); (colton.Add(3);)

Lists<int> - generic

Sets

* No defined order and no concept of duplicates(add something 2x is same as adding 1x)
* Good for checking memberships(faster than lists) and insert and remove is faster than lists
* Var set = new HashSet<string> {“a”,”b”,”c”};

Maps(Dictionary)

* Var dict = new Dictionary<int,int>{[1] = 2; [2] = 1;};

Collection Initializer

* Syntax? – int [] inArray = new int[] {1,2,3,4};
* What is it? – an easy way to initialize an array without using a statement to declare new array of certain size and using multiple statements to initialize each spot??
* Why useful? – the size of the array is inferred and makes it easier/faster to fill arrays

Jagged Arrays

* What is it? -
* Why useful?
* Syntax? – int[][] twoD = new int[][];

Multi-Dimensional array

* Syntax? – int[,] twoDMulti = new int[3,5];

Exceptions

Debugger

* What is it?
* Why useful?
* What is a breakpoint and why would we use it and how is it useful?
* Syntax?

**Git ignore** - The very 1st thing when you add a new repo is setup the gitignore!!

Type into gitignor.io – Csharp, Visual Studio Code and Visual Studio and copy and paste into a .gitignore file created in repo

**Git restore** - - staged – restore things you added

Git restore -- staged . – restores all just added

Git commit –m “add some notes and hello world C#”

1. All new repos should get gitignore
2. Commits should be focused
3. Every commit needs a descriptive commit message (first line is the brief description of the commit then uses added and fixed tabs)
4. Pull often
5. Push often
6. Commit frequently

**variable** – a container for a value of some type. Once declared, can’t change type