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

Lists<int> - generic

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 add .

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