Strings

CS 233N/ CS 162N

Mari Good

Objectives

- String and Char datatypes
- Strings are immutable
- Introduce char methods and string methods and the documentation for char and string
- Practice with several examples

String and Char

- C# has 2 datatypes for character data
  - char – is a value type (like int) that stores one (Unicode) character. A character literal must be delimited with ‘
  - string – is a reference type (like array) that stores a set of characters. A string literal must be delimited with “. String and string can be used interchangeably in C#.

Things You Can Do With a char

- The Char page at microsoft.com gives a good list of Char methods.
  - https://docs.microsoft.com/en-us/dotnet/api/system.char?view=netframework-4.8#methods
  - Most of these are static methods.
  - Lots of them can operate on both a single character and a string and an index.
  - Let’s look at some of these together and “play” a little on dotnetfiddle.net.

A String is Immutable

- Which means the state of a string can't be changed after is has been created.
  - If we try to change the value of a string by concatenation (using + operator) it actually results in creation of a new string object to hold a reference to the newly generated string.
  - It might seem that we have successfully altered the existing string. But behind the scenes, a new string reference is created, which points to the newly created string.

Things You Can Do With a String

- The Strings page at tutorialspoint.com gives a good list of string methods. BUT please remember that none of these methods CHANGES the original string. Strings are immutable!
  - https://www.tutorialspoint.com/csharp/csharp_strings.htm
  - Let's look at some of these together and "play" a little on dotnetfiddle.net.

Now that we're warmed up

- The first 4 questions in the lab ask you to improve on the first 2 versions of PigLatin.
  - Version 3 is problem 1 – multiple consonants
  - Version 4 is problem 2 – first letter is capitalized
  - Version 5 is problem 3 – last letter is punctuation
  - Version 6 is problem 4 – multiple words separated by whitespace
- The last 2 problems in the lab ask you to implement a Shift Cypher.
  - Version 1 is problem 5 – one word
  - Version 2 is problem 6 – multiple words
- For each version, I'll get you started in a screencast.
- You'll be using git and github to manage your source code.

Keeping Track of Your Work

- Professional programmers use version control software to manage source code and documents during the software development process.
- Git is currently the most popular version control system
  - Popularized by GitHub, largest cloud based source code host with 50+ million repositories
- Stores files and change information in a repository
  - Records changes to files
  - Allows specific versions to be recalled on demand

Before We Get Started

- If you don’t already have a github account, please create one now. Use the email you use for all LCC communication.
- If you haven’t already installed the git command line tools, please do that now.
  - We’re going to start with the command line tools because it’s easier for me to control the process if you type commands at the command line. AND because employers will want you to know the command line tools.
  - There are lots of GUI client tools and most IDEs have git support built in. Once you know the basics you can start to work with GUI tools.

Creating a Local Repo(sitory)

- I’m using a tool called GitHub Classroom to help manage your git repositories. Every time I assign a lab, I’ll create a “template” repository that includes the starting files.
- Accept the GitHub Classroom assignment in moodle.
  - This will create a private GitHub repository for you and will take you to the repository page in GitHub.
    - You’ll work on your code on your local repository and PUSH it to the GitHub repository regularly.
    - You’ll give me the url for your GitHub repo when you want me to look at your code for grading or answering a question.
  - Cllick clone on the GitHub page. That will copy the url of your repository to the clipboard.

Creating a Local Repo(sitory)

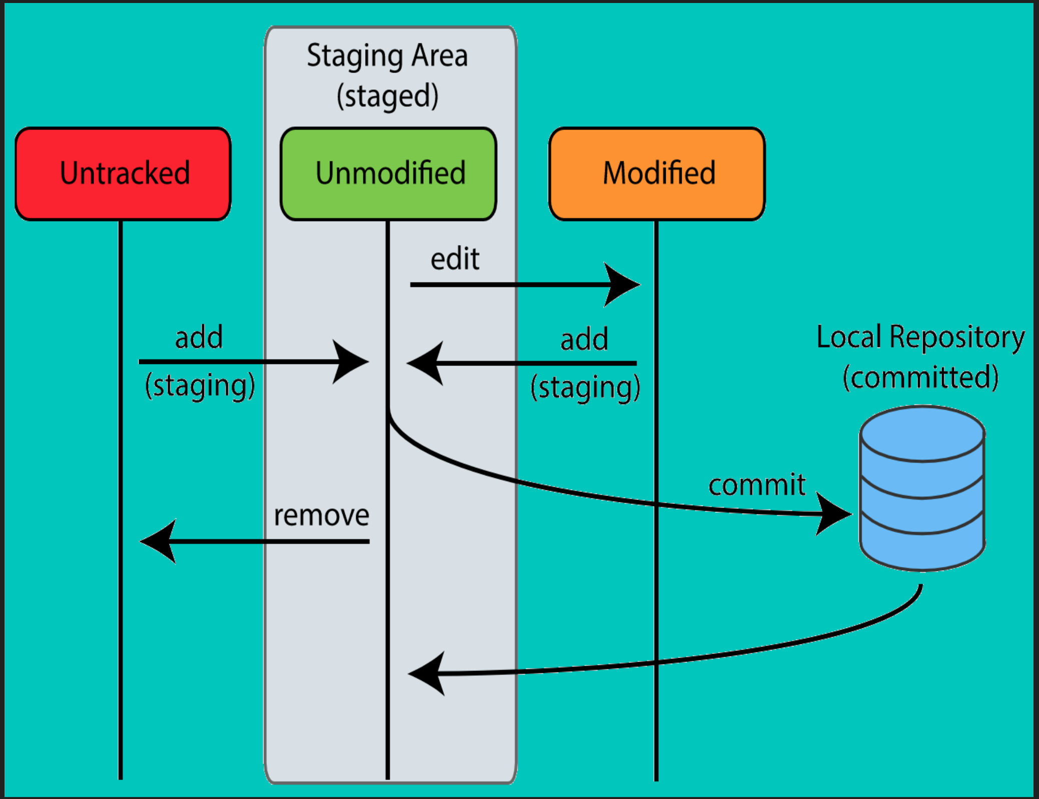
- Create a folder on your machine.
  - The folder and all of the files in it will be managed by Git. This folder will contain your local repository.
- Select gitbash here from the context sensitive menu.
  - This will open a command window that you'll use to execute Git commands.
- Type git clone and paste the url from the clipboard. Press enter.
  - A folder will be created on your machine. This folder is the local repo for lab 1.
  - Type cd followed by the name of the folder.
    - This will take you into the folder.

Working on Your Lab

- Open a solution in Visual Studio
- Notice that there are 2 projects – PigLatin and ShiftCypher
- Add some code (even if it's just a comment) in program.cs in both projects
- Save all.

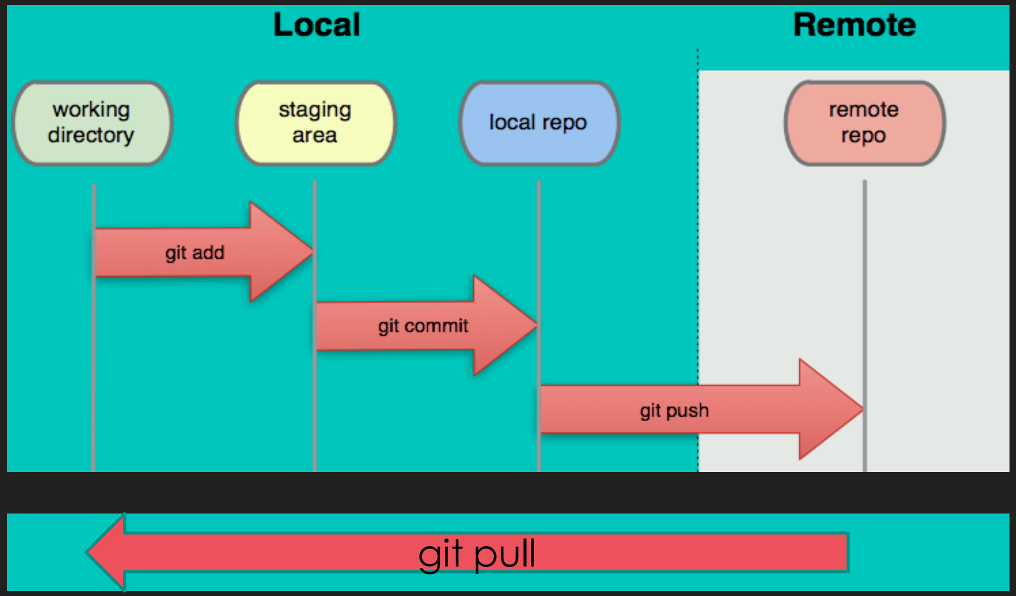
Git Managed Files

- Git's main states for the files it manages
  - Untracked - File exists but not yet added to repository
  - Modified - File changed but not committed
  - Staged - File marked as changed and ready for next commit
  - Committed - File stored safely in local repository database
- Type git status to see the status of your files



Git Managed Files

- Commands for working with remote repository
  - Push - Copy all local commits to remote repository
  - Pull - Copy remote commits to local repository and merge changes with local files. You shouldn't need this because you're the only one working on your lab.



Working With Git

- In the same command window type
  - git status
    - Git tells you that some of your files have changed
  - git add –A
    - Adds all of the files to the staging area
  - git commit –m “Added some comments”
    - Saves the changes to the repo database
  - git push
    - Saves the changes to the remote repo on GitHub. Check it out!
    - The first time you do this you’ll get an error. Type the 2 commands exactly as they appear in the error message. This will associate your local repo with the remote repo on GitHub.

Working on Your Lab

- As you finish each problem
  - git status
  - git add –A
  - git commit –m “some message”
  - git push
  - Check your files on GitHub
- When you ask a question, make sure that your most recent code is pushed to GitHub and give me the url for your GitHub repo.
- When you submit your lab you’ll give me that same url. No more copy and paste!

What’s Next

- Don’t forget
  - Reading Quiz 1
  - Programming Quiz 1
  - Lab 1
    - First Draft and Final Version. Each draft must include a Self Evaluation.
- An Introduction to Data Structures
  - List
  - Stack
  - Queue