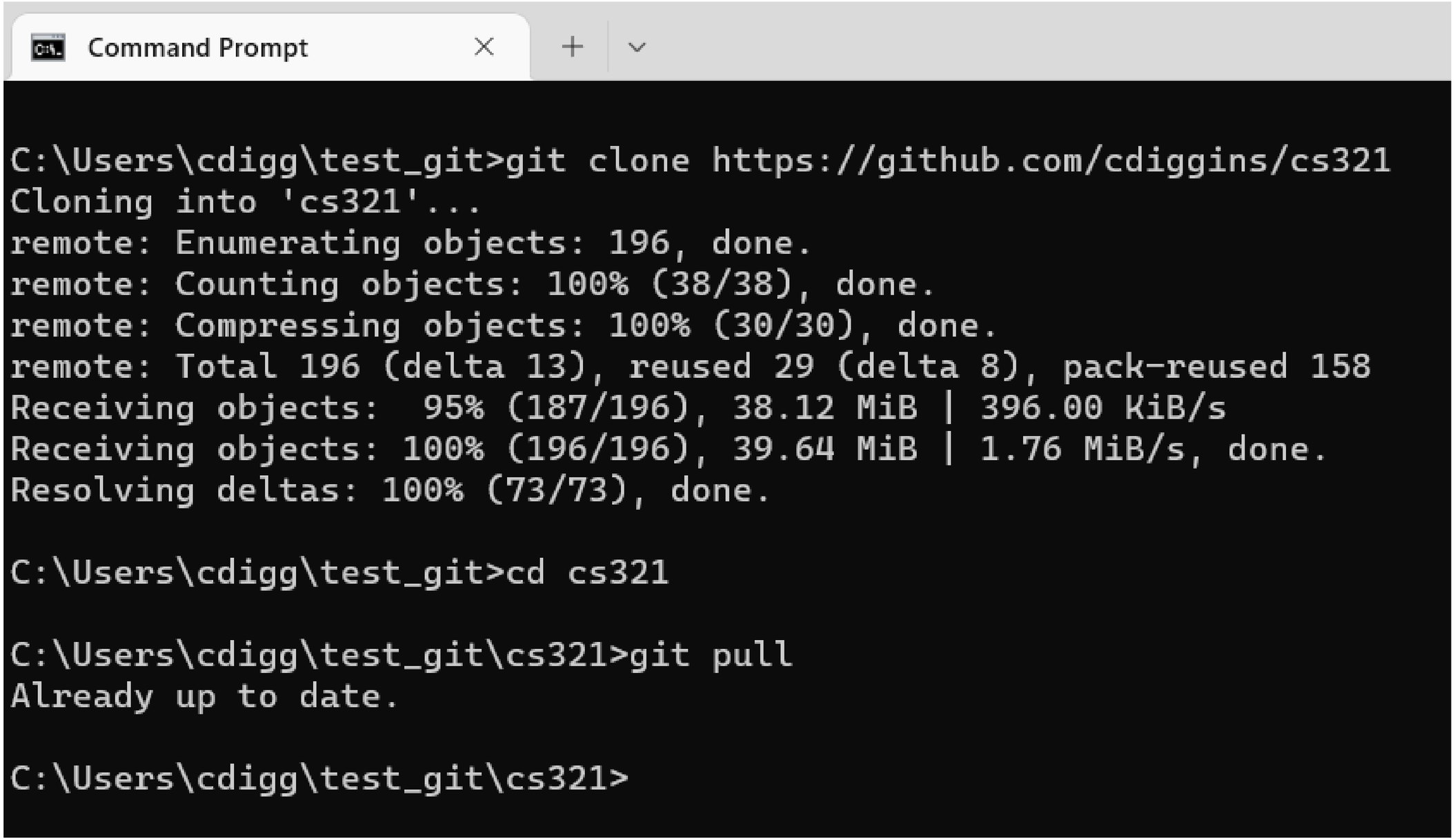
CODING TIPS


Command Prompt
C:\Users\cdigg\test_git>git clone https://github.com/cdiggins/cs321
Cloning into 'cs321'...
remote: Enumerating objects: 196, done.
remote: Counting objects: 100% (38/38), done.
remote: Compressing objects: 100% (30/30), done.
remote: Total 196 (delta 13), reused 29 (delta 8), pack-reused 158
Receiving objects: 95% (187/196), 38.12 MiB | 396.00 KiB/s
Receiving objects: 100% (196/196), 39.64 MiB | 1.76 MiB/s, done.
Resolving deltas: 100% (73/73), done.
C:\Users\cdigg\test_git>cd cs321
C:\Users\cdigg\test_git\cs321>git pull
Already up to date.
C:\Users\cdigg\test_git\cs321>

First Pieces of Advice

- Practice, practice, practice
- You will get much faster
- If you are stuck, ask for help

Learning to Program

- Recognizing mistakes
- Reading code
- Problem solving
- Writing code

Programming Note

- If you are confused
- Set a breakpoint see what it does
- Step through the code
- Ask yourself, is it doing what I expected?
- Why or why not?

If you are not sure what you expect

- Valid!!
- Write out what you want to happen in code
- Find the inconsistency in your code and understanding
- Ask yourself in a comment
- Make the best guess
- Mention you are guessing

Tracing and Asserts

- Use Debug.WriteLine() in regular code
- Use Console.WriteLine() in tests
- Use Debug.Assert() in regular code
- Use Assert.IsTrue or Assert.AreEqual in tests projects

Reversing the Direction

- A problem solving trick is to reverse the direction that we loop over stuff
- Look for a “counter example” as opposed to proving something is true

Consider the Test Cases

- Make them explicit
- Don’t solve every problem right away
- Sometimes code is “good enough”
- That is where documentation is important

Use Braces

- If statements and loops don’t require curly braces
- However, it can make code easier to read and avoid errors

Use Variables

- While learning, break complex expressions down into smaller ones
- Assign sub-expressions to variables
- Yes your code is longer, but easier to find mistakes
- You can also debug easier

Comments

- Don’t put too many
- Don’t state the obvious
- They are for you and the reader
- Tell us something useful
- Code should speak for itself

Don’t Leave Unused Code in a File

- It can be confusing to yourself and others

For Note:

- Don’t monkey with the loop index

What was

- oxFFul - hexadecimal number 255, typed as a 64 bit unsigned number (long)
- 123.45f – single precision floating point number
- TIP: When using the “var” declaration hover the mouse over the var