

Lab 1

Shelby Jorgensen

CS380: Software Engineering

Question 1: What are the steps involved in creating a new repository on GitHub, and how can you add files to it using the command line?

Create a repository:

Step 0: Create a GitHub account, or login to an already created account.

Step 1: From your accounts front page, go to the 'Repositories' tab, and click the 'New' button on the right side of the page.

Step 2: Give the repository a name, and declare the visibility, or who can see and contribute to your repository.

Step 3 (Optional): Give the repository a description, and add a README page, to give additional information on the project you're working on.

Step 4: Click 'Create Repository' at the bottom of the page, and your repository will be created and ready for your contributions.

Add files from command line:

Step 0: Open Git Bash

Step 1: Get the file location for the file you will be adding to the repository.

Step 2: In Git Bash, type out 'cd ' then paste in the file location, this brings you to the directory of this file.

Step 3: Move to the parent directory of your file that you will be uploading. In Git Bash this can be done with the command 'cd ..'

Step 4: Type in the command 'git init', which creates an empty git repository.

Step 5: Get the URL to the repository you will be pushing your files into.

Step 6: Use the command 'git remote add origin ', then paste in the URL to your repository at the end.

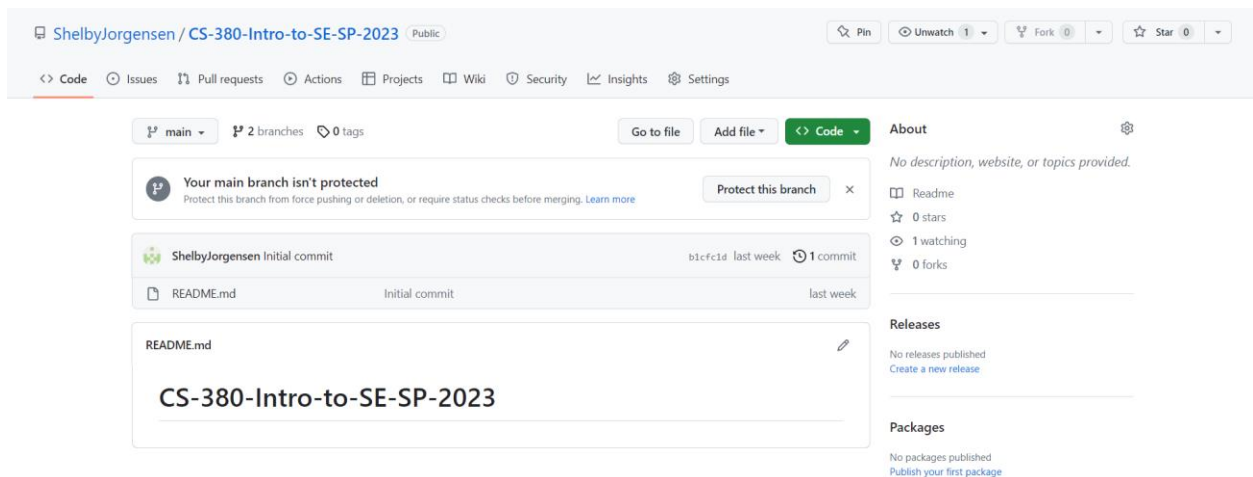
Step 7: Use to command 'git add .' which will add the file you previously directed to into the repository.

Step 8: Enter the command 'git commit -a -m "(enter commit title here)"', which will commit this file to your repository. Anything placed inside the "" in this command will be the title of your commit on the repositories page.

Step 9: Finally, enter the command 'git push -u origin -all', which will push all the files you committed to the repository.

Step 10: Now you should be able to go to your GitHub page and see your files committed to the repository.

Question 2: Create a repository and call it CS-380-Intro-to-SE-SP-2023.



Question 3: How can you reverse a given number in Java, and what would be the code implementation for it?

Ex: Given Numbers

12345

Reverse Number 54321

Outline: One way to reverse any number in Java would be to convert the integer into a StringBuilder, and use the reverse() function built into the StringBuilder class, then convert back into an integer.

Code: Screen shot from Eclipse IDE:

eclipse-workspace - CS380_Lab1/src/lab1/CS380_Lab1.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

CS380_Lab1.java ×

```
1 package lab1;
2
3 public class CS380_Lab1 {
4     public static void main(String[] args) {
5         int givenNum = 12345;
6         System.out.println("Given Number: " + givenNum);
7         StringBuilder str = new StringBuilder(Integer.toString(givenNum));
8         str.reverse();
9         int reverseNum = Integer.parseInt(str.toString());
10        System.out.println("Reversed Number: " + reverseNum);
11    }
12 }
13
14
```

CS380_Lab0
CS380_Lab1
Lab1
Lab2
Lab3
Lab4
Lab5
Lab6
Lab7
LabAssignment1
LabAssignment2
LabAssignment3
LabAssignment4
LabAssignment5
LabAssignment6
LabAssignment7
LabNumber3
MemoryGame
Project0
SingleErrorCorre
Test

Problems Javadoc Declaration Console Error Log Coverage

<terminated> CS380_Lab1 [Java Application] C:\Users\shelb\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.4.v20220

Given Number: 12345
Reversed Number: 54321