

A diagram of a code

Description automatically generatedA screenshot of a computer

Description automatically generatedA screenshot of a computer program

Description automatically generated

|  |
| --- |
| I m p o r t a n t ! |

Start Git for my first project in this course:  
1- CTRL + ` to go to terminal in my VS cdoe type: D: (powershell)

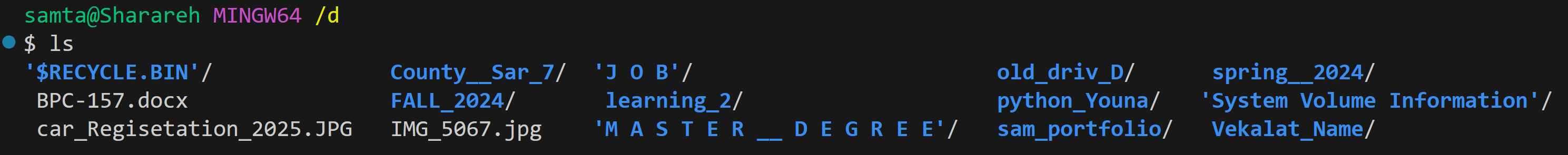
2- CTRL + ` to go to terminal then click + on terminal to choose bash (now type: cd /d/ ) to

go to D folder on computer.

A green text on a black background

Description automatically generated

3- Type ls to see the files inside of the folder we are in now.

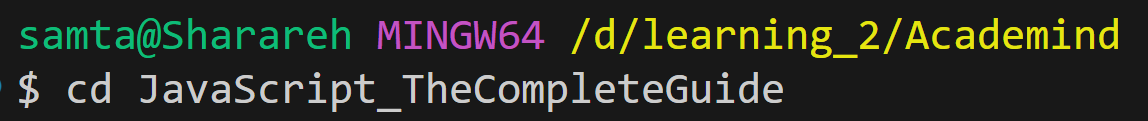


4- I choose the learning\_2 folder inside of the D folder. I type: cd learning\_2

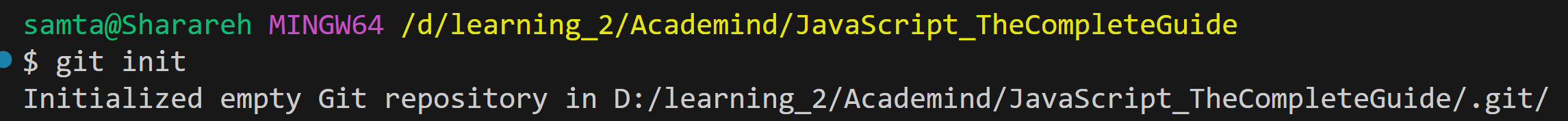
A close up of a black background

Description automatically generated

5- I type ls to see the folder inside of leaerning\_2. Then I choose Academind folder. Therefore, I typed cd Academind

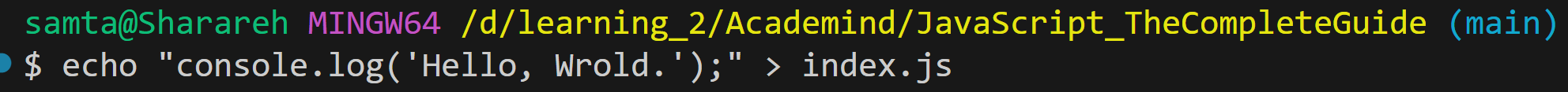
6- Again I choose another folder inside of it. I typed: 

7- Now I want to initialize a new Git repository. Inside of our project directory, we initialize a new git repository: we type: git init



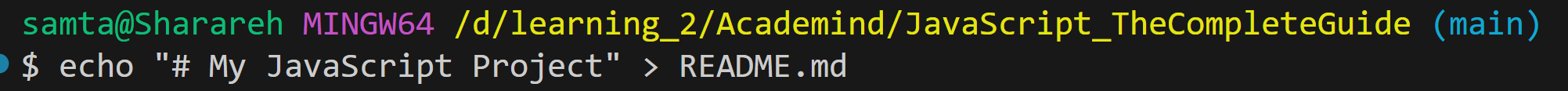
8- Now we create a simple JavaScript file to start our project: we type

echo “console.log(‘Hello, World!’);” > index.js



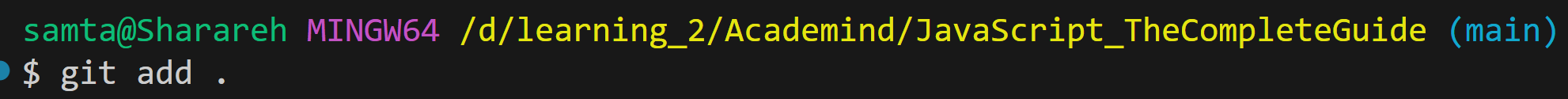
9- Now we create a README file in our project to explain what it does: we type

echo “# My JavaScript Project” > README.md



10- now as we see from the screenshot tow files has been created:  
A number of days and months

Description automatically generated with medium confidence

11- now we stage our files to commit. We need to add the files to stage area: we type git add . 

Notice: the space between git add . the point has distance from add ‘ . ‘ adds all files in the directory. We also can add specific files if needed.

12- Now commit our changes with a message: we type: git commit -m “our message” A screen shot of a computer program

Description automatically generated

13- Now create a New Repository on GitHub:

We go to GitHub account and create a new repository. We do not initialize the repository with a README, .gitignore, or license since we already have those locally. I went to my GitHub account and I created a new repository, I named it “JavaScriptAcademind”

Then GitHub Showed me this page:  
A screenshot of a computer

Description automatically generated

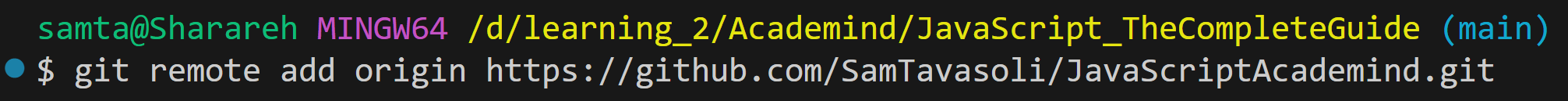
14- Now in the terminal, we can connect our local repository to the GitHub repository we just created: we see the commands in the page after creating repository in our GitHub account and we can use it.

There are two parts. One says …or create a new repository on the command line

and the other says …or push an existing repository from the command line

we use the second part since we created our repository locally: we type

git remote add origin the URL that GitHub gave it to use after we created new repository in our GitHub account.



15- Now we push our code to the GitHub repository using; git push -u origin main

A computer screen shot of a program

Description automatically generated

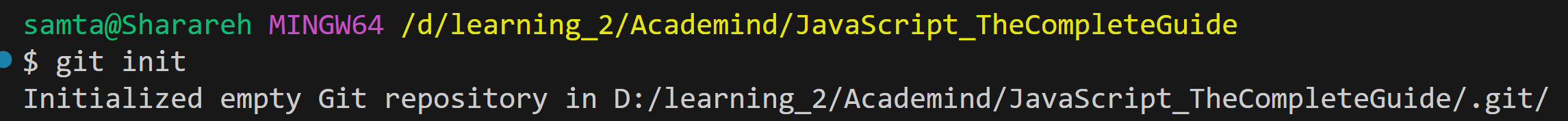
16- Now we can check our GitHub repository. We can go to our GitHub repository in our web browser, and we have to see the files there.

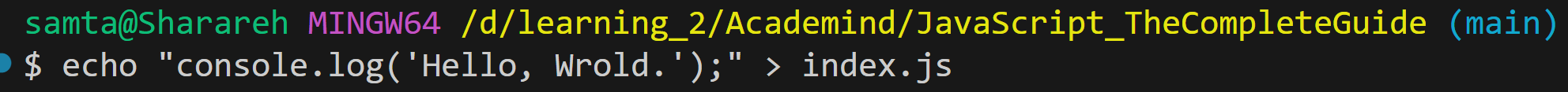
**Summary of our commands**:  
**1- Go to directory**

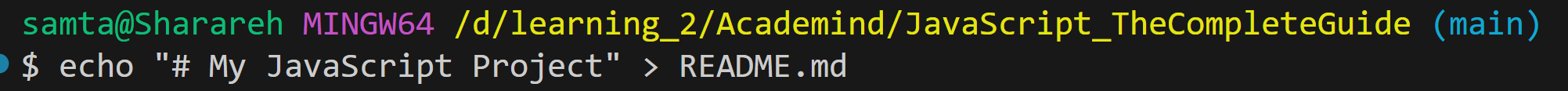
A black and green sign with green letters

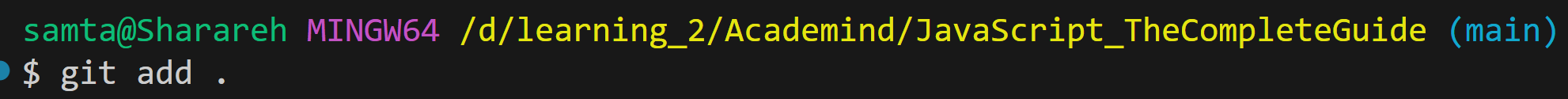
Description automatically generated

**2- Initialize the repository:**



**3- Create JavaScript file:**  


**4- Create README file:**  


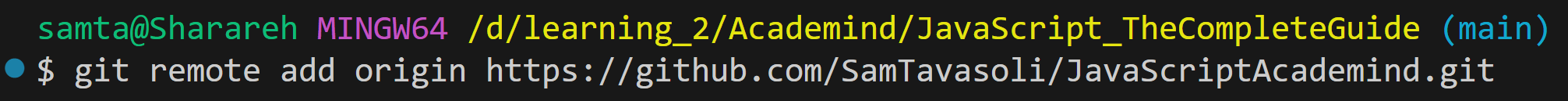
5- Stage our files:  


6-Cmmit our files:

A screen shot of a computer program

Description automatically generated

7-Link to GitHub:

8-Push our code: A computer screen shot of a program

Description automatically generated  
   
A screenshot of a computer program

Description automatically generated

Instead of master we have to type **main**

I opened the folder in VS code and I used git status and I received this:  
 A screen shot of a computer program

Description automatically generated

It tells us where I have made changes. It tells I have made changes in .docx and in a temporary files (~WRL3728.tmp) if I want to ignore temporary files or files that are not part of my project (like word temporary files) I can add them to a .gitignore like:  
A screenshot of a computer

Description automatically generated

A black screen with white text

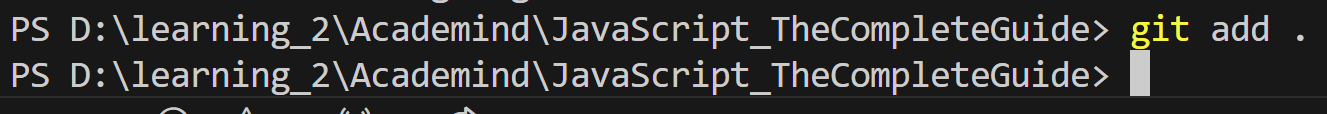
Description automatically generated

Now add the .gitignore file to our repository: we type git add .gitignore



Commit:  


Now stage files we want commit (all or specific):  
type: git add .



Commit:

A black screen with white text

Description automatically generated

Push the changes to GitHub:  
type: git push origin main

A screen shot of a computer program

Description automatically generated

I put the docx in .gitignore file, but when I use git status, it still tell me :  
A screen shot of a computer

Description automatically generated

The solution is:  
The issue you’re encountering happens because the .docx file was already being tracked by Git before you added it to your .gitignore file. Once a file is tracked by Git, it will continue to be tracked even if you add it to .gitignore.

**How to Stop Tracking the .docx File**

To stop tracking the .docx file and any other files you want to ignore, follow these steps:

1. **Remove the .docx file from Git tracking without deleting it from your working directory**:

A black and white rectangle with white text

Description automatically generated

This command removes the file from the staging area (and stops Git from tracking it), but it keeps the file in your working directory.

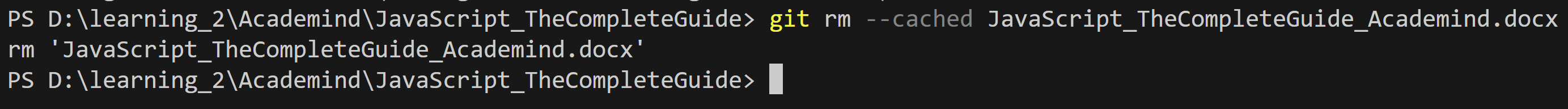
1. **Add and commit the change**:

A screen shot of a computer

Description automatically generated

1. **Verify that Git is no longer tracking the file**:



I did it :  


A screen shot of a computer

Description automatically generated

|  |
| --- |
| End of I m p o r t a n t ! |

Importing JavaScript to our index.html file:

We add it in the <head></head> section A black background with white lines

Description automatically generated



We can put on lower part, before the </body>

A black screen with orange and white text

Description automatically generated