A close up of a logo

Description automatically generated

**Lesson 3 Demo 1**

**Git Commands**

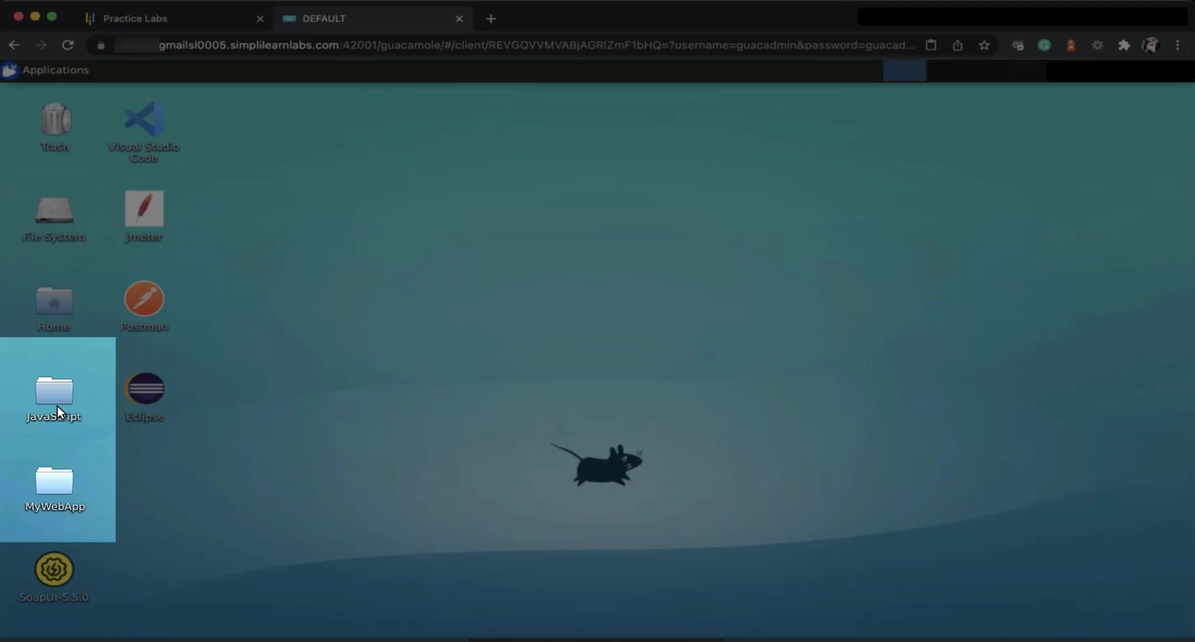


Steps to be followed:

1. Execute Git commands

**Step 1: Execute Git commands**

* 1. You have two different folders in which you will be initializing the Git. Execute the few commands to understand how to sink the code on the GitHub and the bitbucket.



* 1. Go to **github.com.** Create a repository, you can use the same name as MyWebApp, or it can be any name of your choice.  
       
     Graphical user interface, website

     Description automatically generated
  2. Write the description *This is my web app with HTML and CSS tutorials*. You can also make the repository public or private. You can also add a README file or .gitignore or choose a license.  
       
     Graphical user interface, application

     Description automatically generated
  3. Open your terminal window and move to your desktop, then move to your MyWebApp. Type ls, you will find files and directories you have created in your project. The first command is going to be **git init.**  
       
     Text

     Description automatically generated
  4. Open Mywebapp from the desktop. Open the terminal window again. Type ls -a to enable the hidden files. Once you do Git initialization, you will see this hidden directory Git coming up

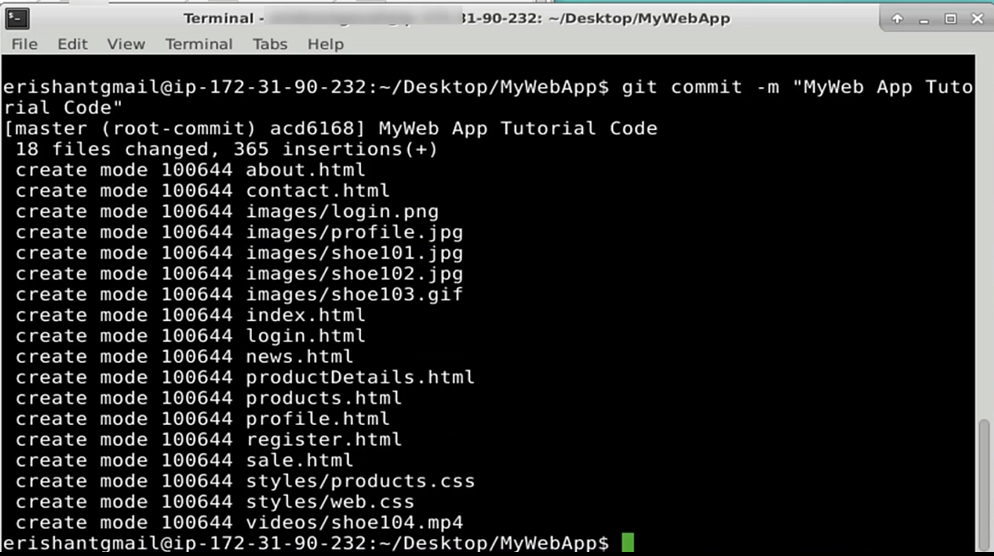
Text

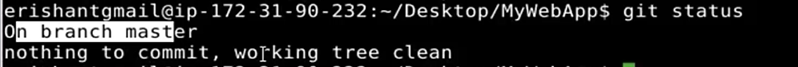
Description automatically generated

* 1. Execute the command git status. With git status, you will comprehend what exactly lies on your branch master. It will show you some untracked files which will be added to an empty repository.

Text

Description automatically generated

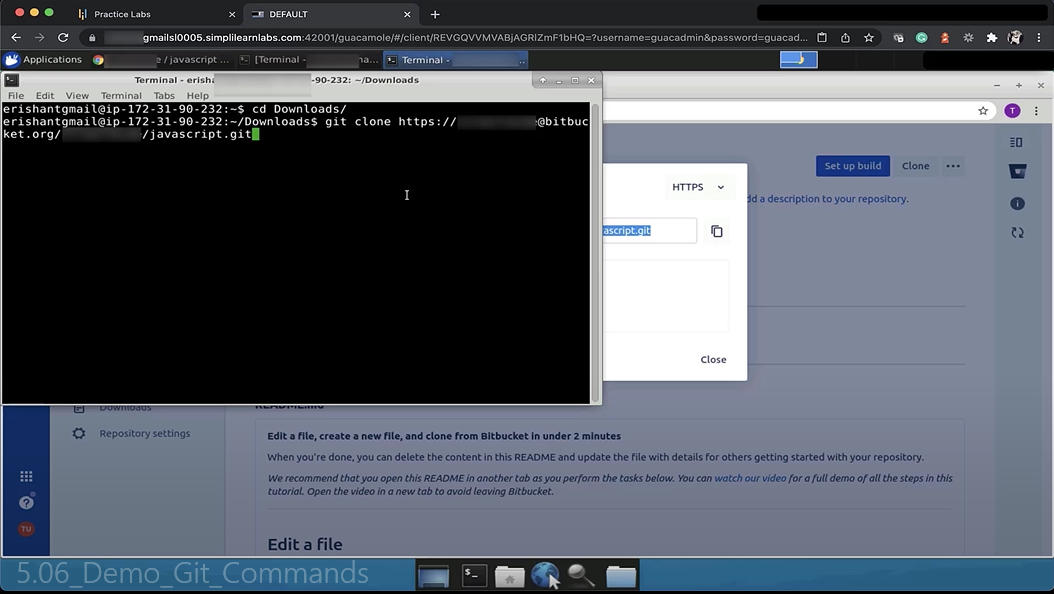
* 1. To add the files in the repository give “git add.”. With the ., you will be able to add all the files. Recheck the status of git with *git status.*  
       
     
  2. The next step is to give the command git commit -m MyWeb App Tutorial Code. The files are recommitted, and you can push them into the repository. After checking the status of the Git, you will see that there is no file on the branch master. It means it is a working tree.  
       
     



* 1. Open the BitBucket from *bitbucket.org*. Login to the bitbucket. Create a repository, and give it the name *JavaScript*. Make it private or public. Select the project in which you are going to work. Let's select a *demo*. Set a default branch name as *main.*

Graphical user interface, application

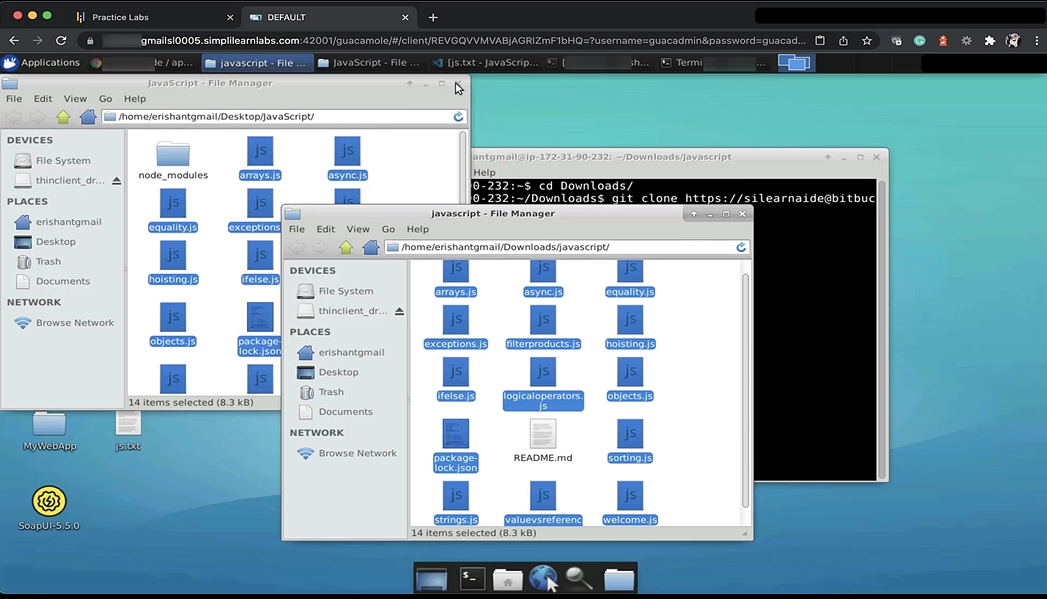
Description automatically generated

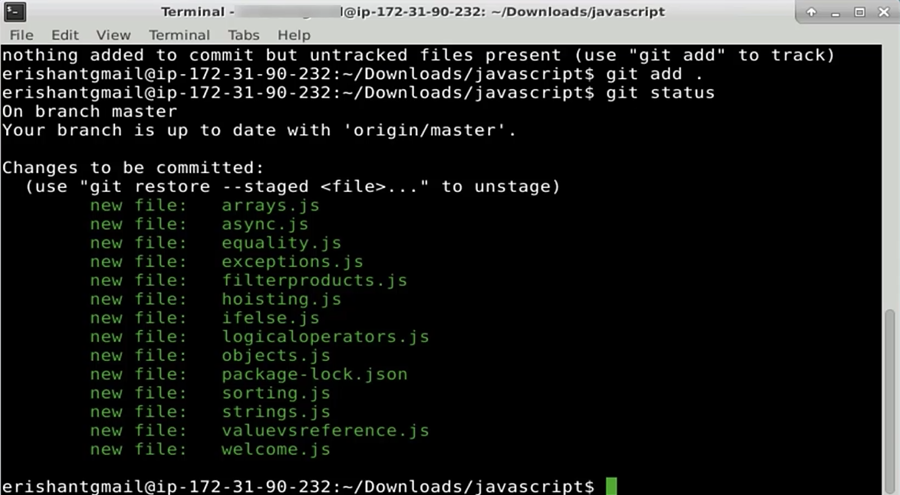
* 1. Clone the repository. Copy the link on the screen. Go back to the terminal. Go to the **downloads directory**. Paste the link copied. Press **enter** to execute the git clone.  
       
     
  2. To fetch the password, go to the repository in your settings and get a password. Go to the app password and generate a new password. Check some of the rules. Label it as jskey or js and click on **create.**

Graphical user interface, application

Description automatically generated

* 1. Open the Visual Studio Code and save the password. Name it as js.txt and save it on the desktop. Paste that password in a terminal window. You will get your repository cloned, which is the Javascript web app.  
       
       
       
     Text

     Description automatically generated
  2. Now type **cd javascript**. You will observe in your downloads directory that you have a javascript folder created. Select all folders from Javascript except node\_modules, copy all folders and paste it into a directory.  
       
     
  3. In the terminal window, type **git status**, and you will get all the files which are supposed to be added.  
       
     Text

     Description automatically generated
  4. Use "git add." to add the untracked files, and all the files will be added. Check the status again with “git status”  
     
  5. To commit the files, type git commit -m *JS Tutorials*. Press enter. A list of committed files is shown on the screen.  
     