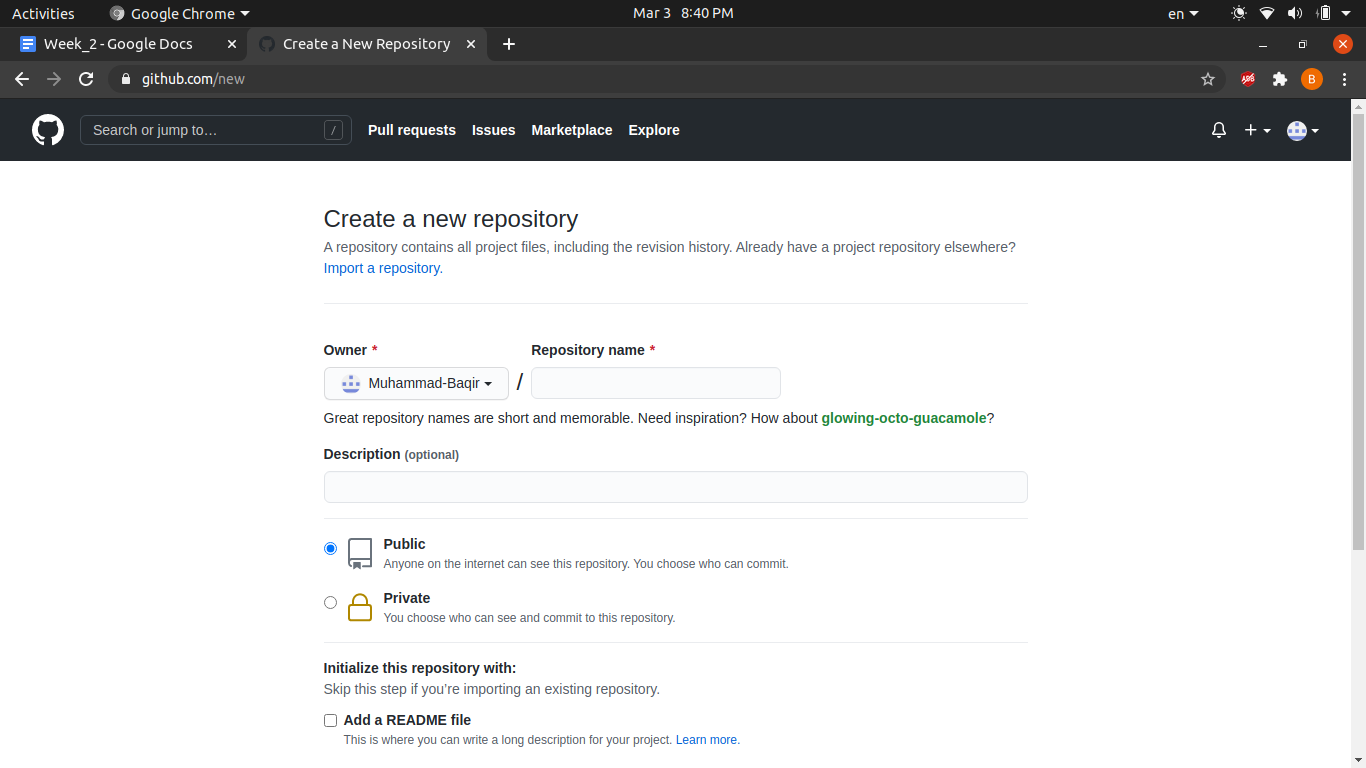
# **Documentation of Week 2**

## **Lecture 3 Practice**

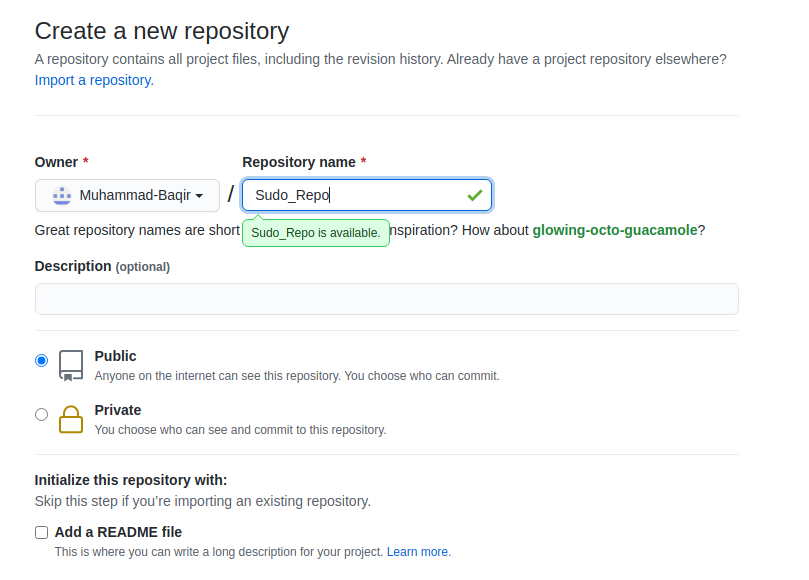
* + I am trying to create a new git repository and will run some commands of git for learning purposes. Details of each step is given with a screen shot.

### **Step 1:** Create a repository on github website.

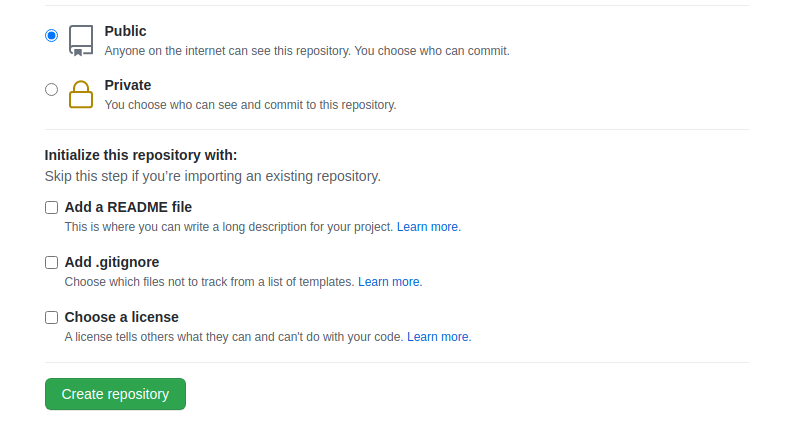
#### **Step 1.1:** Go to [github.com/new](https://github.com/new)

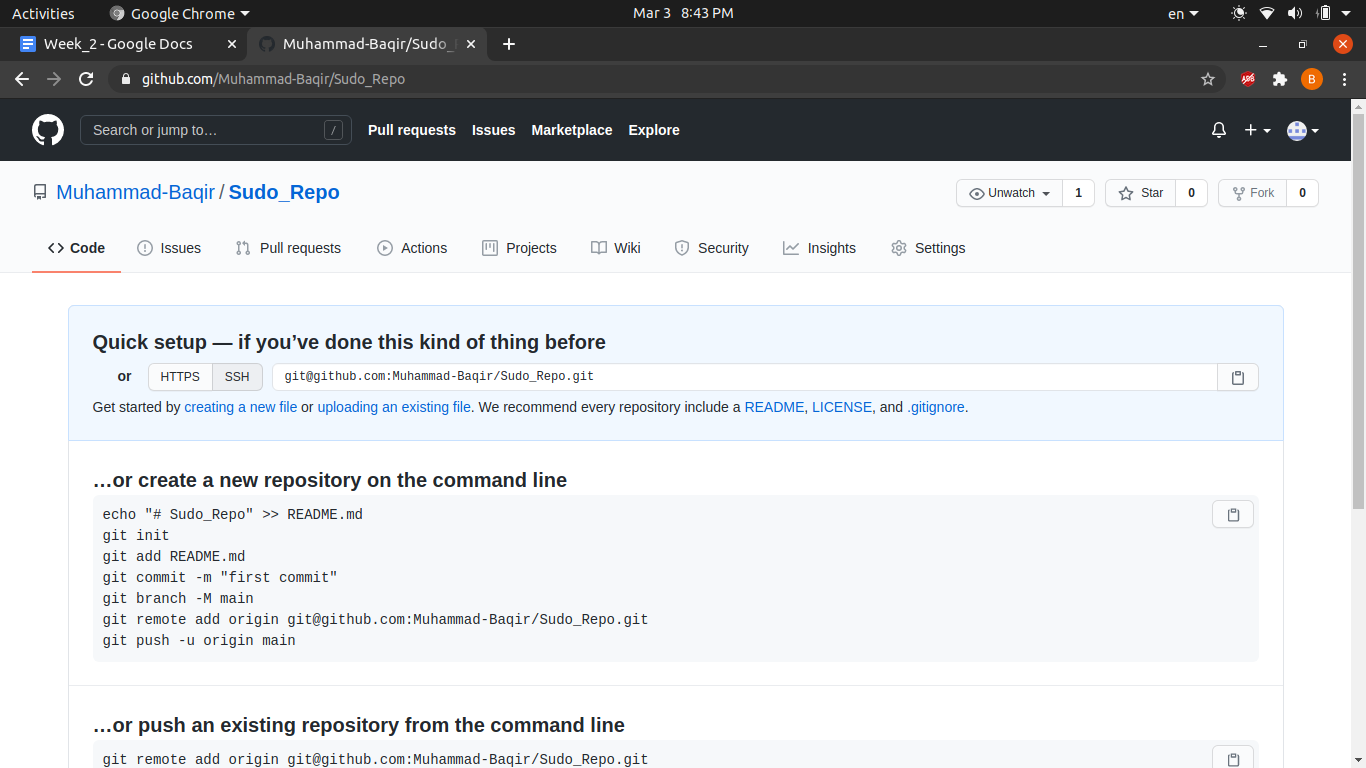


#### **Step 1.2:** Enter repository name



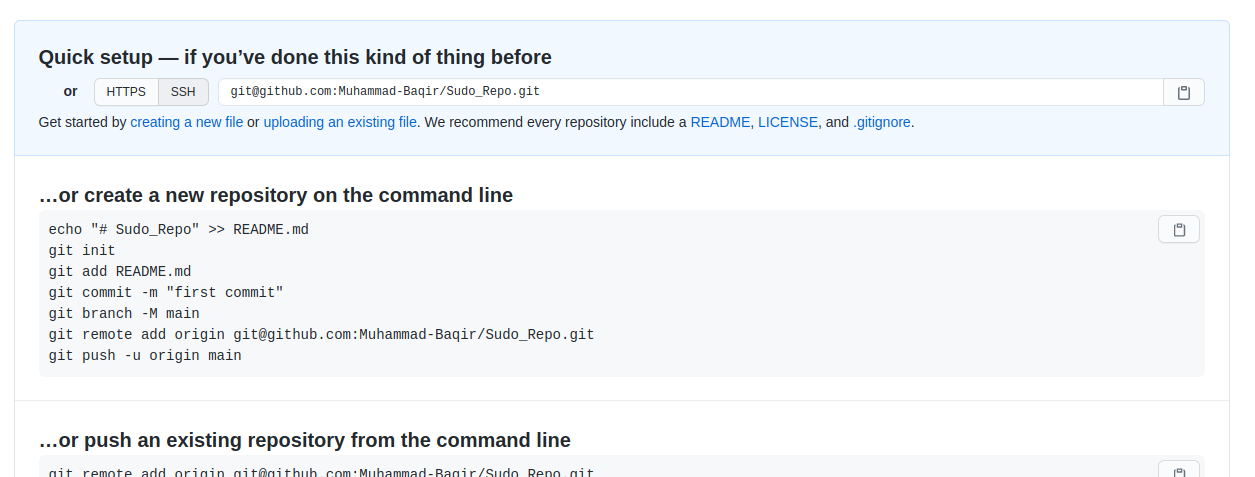
#### **Step 1.3:** Scroll down and click on the **Create Repository** button. You will be redirected to your repository home page.



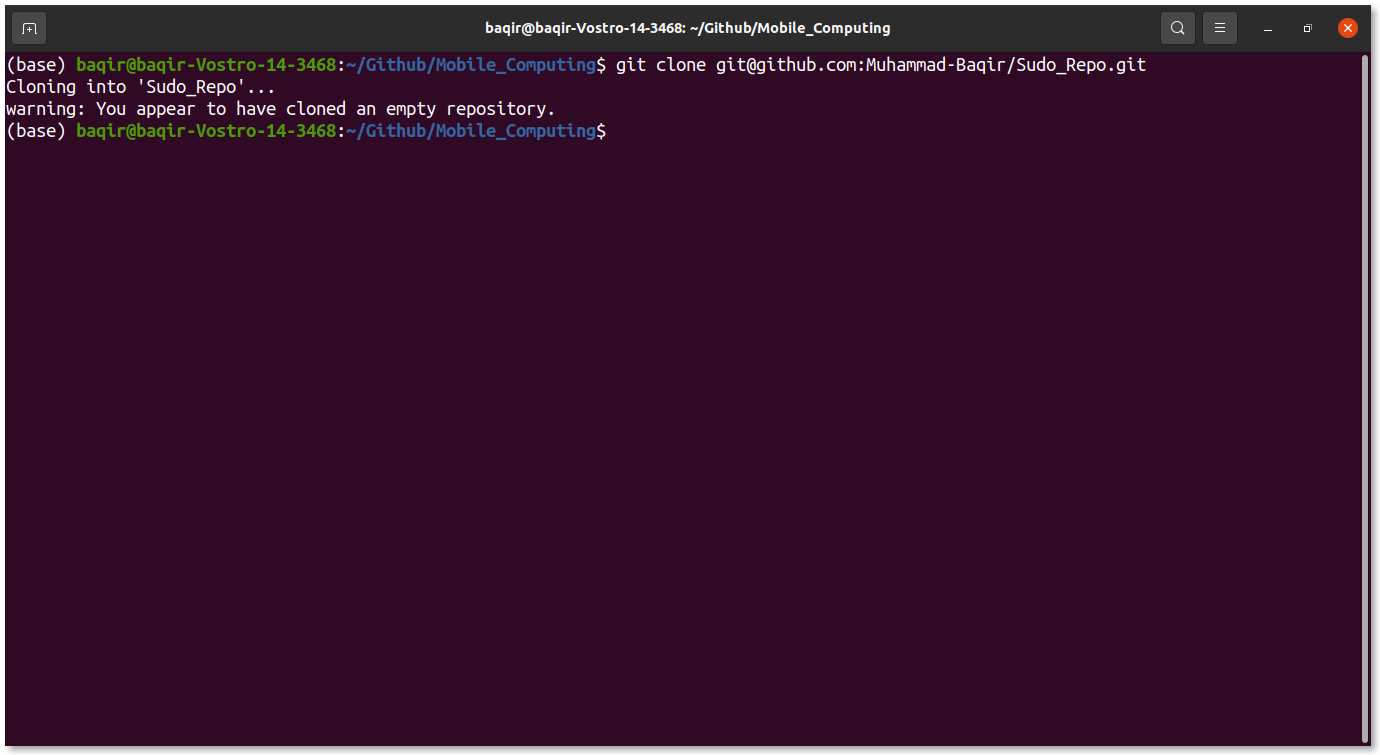


### **Step 2:** After creating a repository online. Let's try to clone it, so that it can be used on your PC.

#### **Step 2.1:** Copy the repository url. You can either copy HTTPS or SSH. I will copy SSH url.

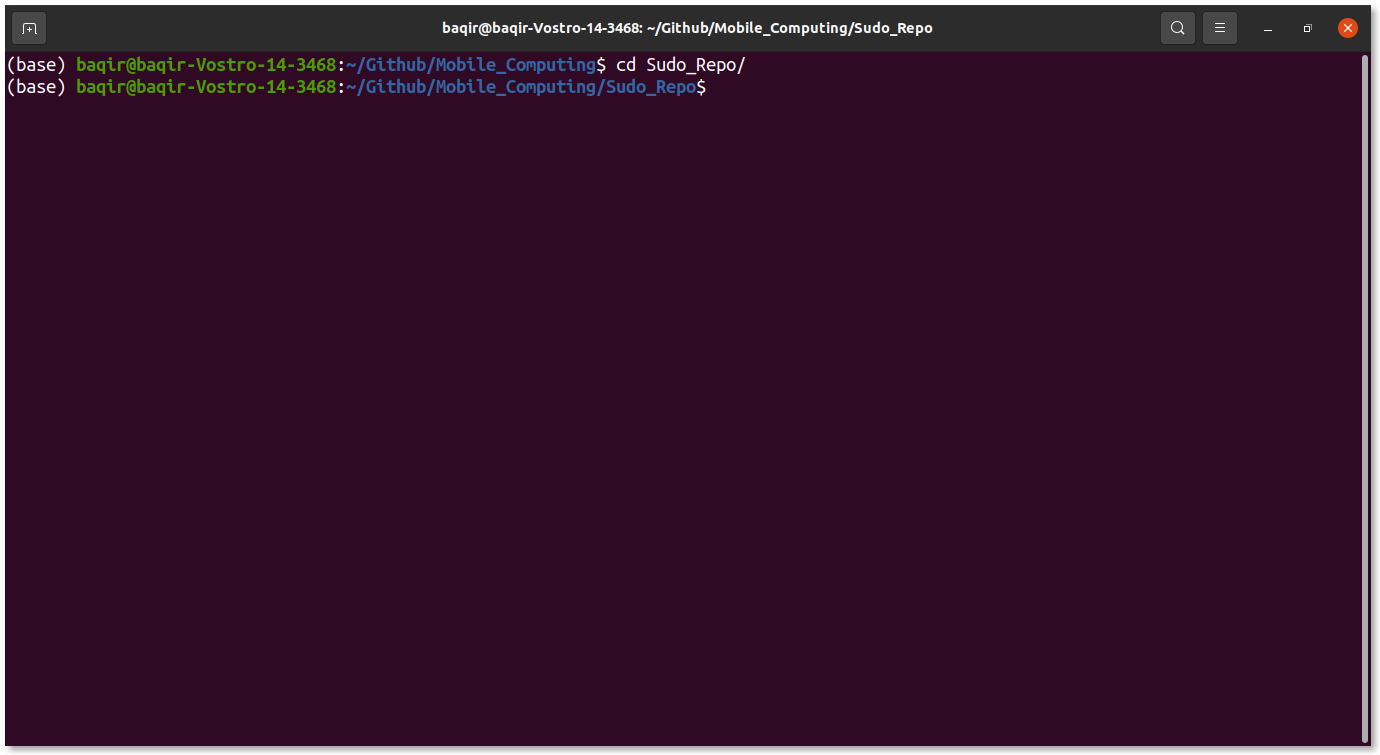


#### **Step 2.2:** Open terminal. Change the directory where you want to clone and enter **git clone repo\_url** command. Git will clone the repository.

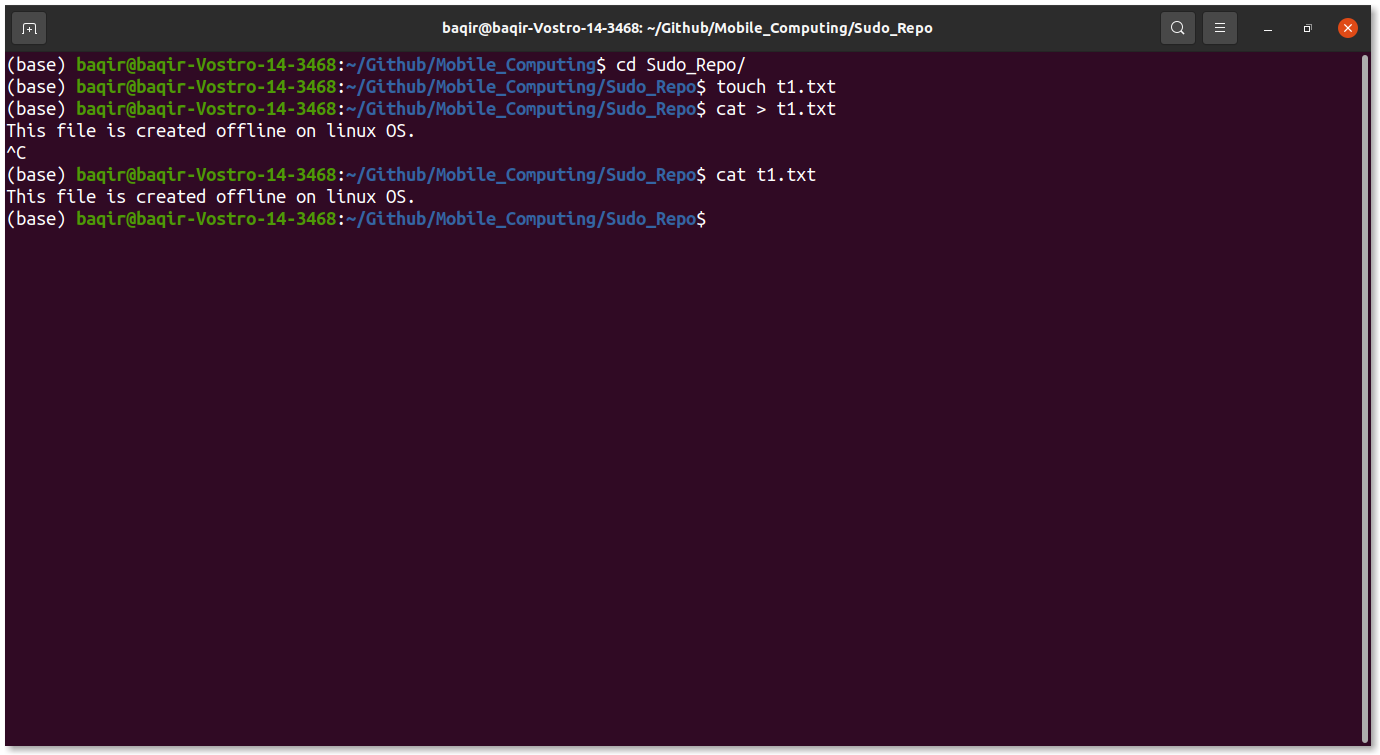


### **Step 3:** Now we have cloned the repository. Let’s add a new file and push it.

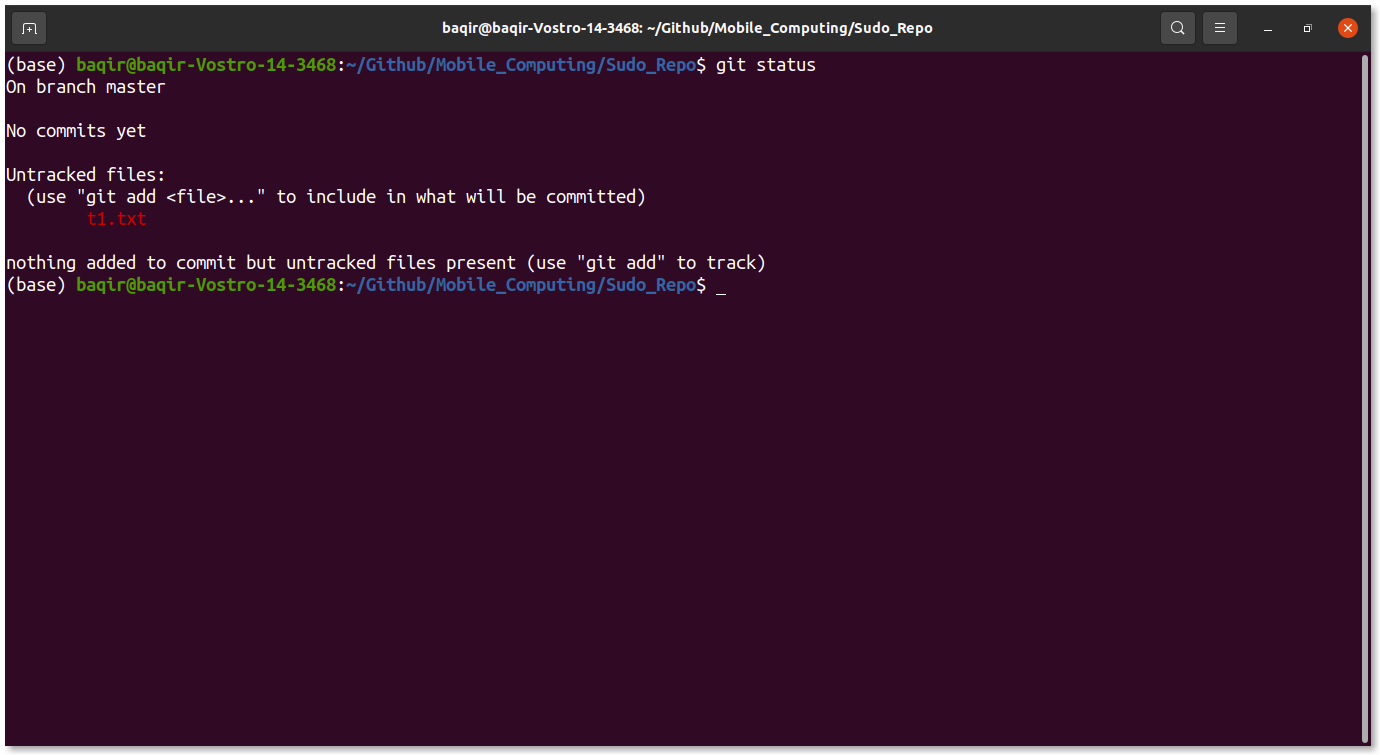
#### **Step 3.1:** Firstly get into the repo directory using **cd directory\_name** command.



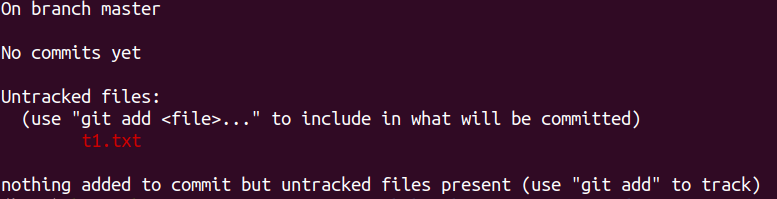
#### **Step 3.2:** Create a new file and add some random stuff.



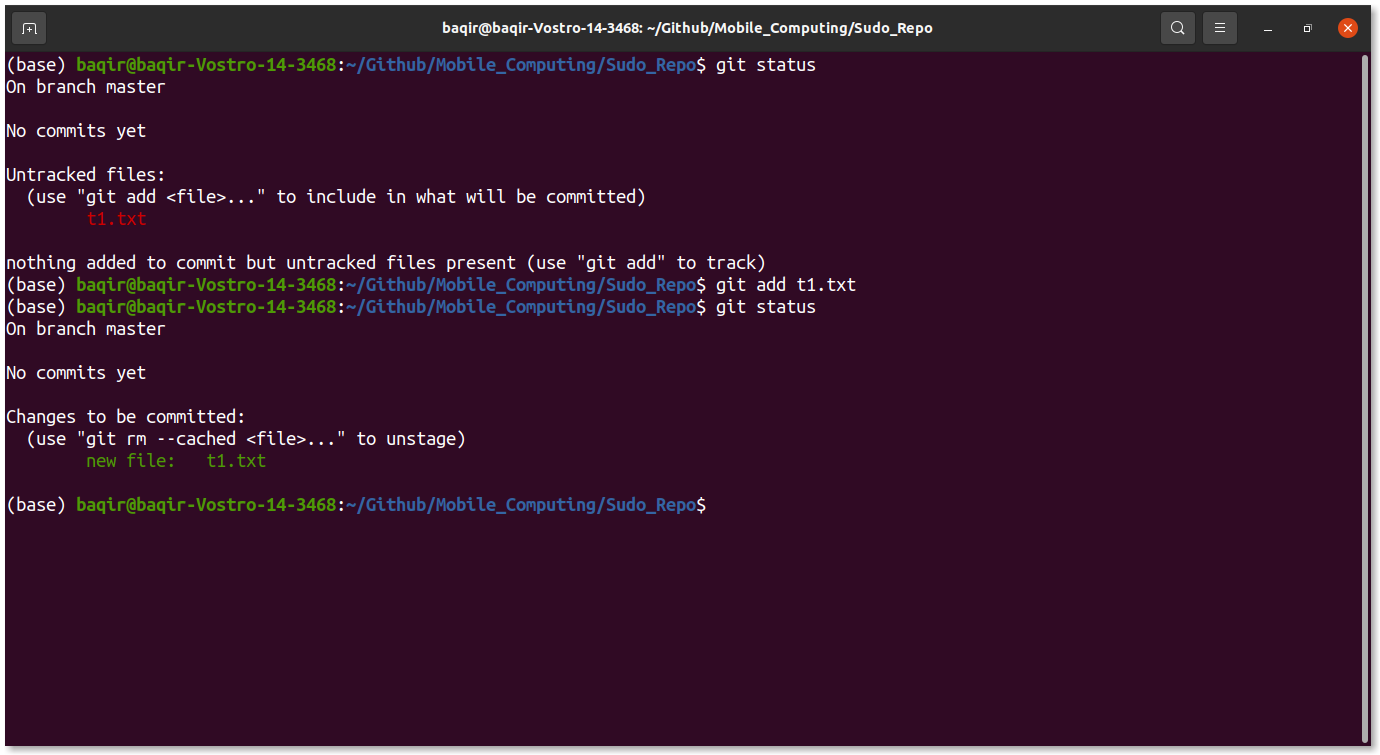
#### **Step 3.3:** Clear the screen and enter **git status** command.



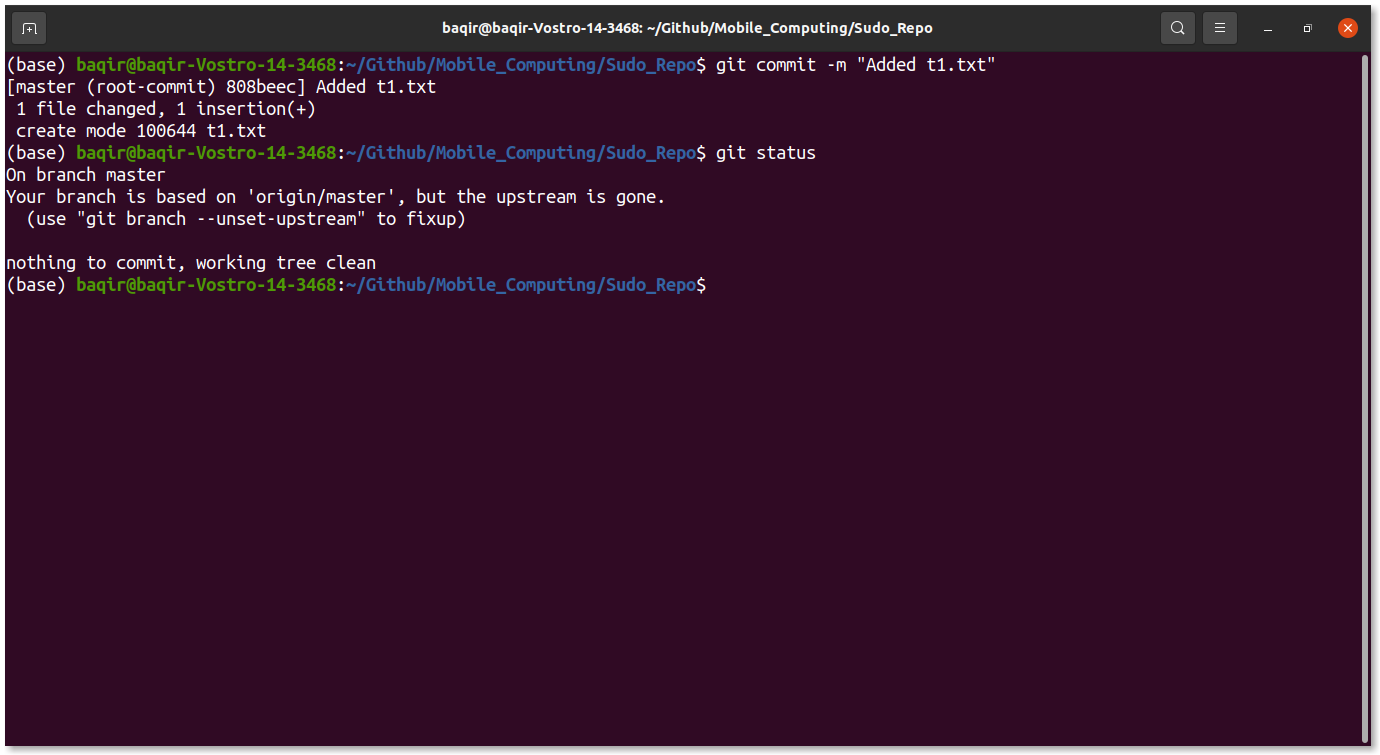
#### **Step 3.4:** Here you can see that under **Untracked files:** t1.txt is highlighted in red. Which means that this file is not yet tracked by git.



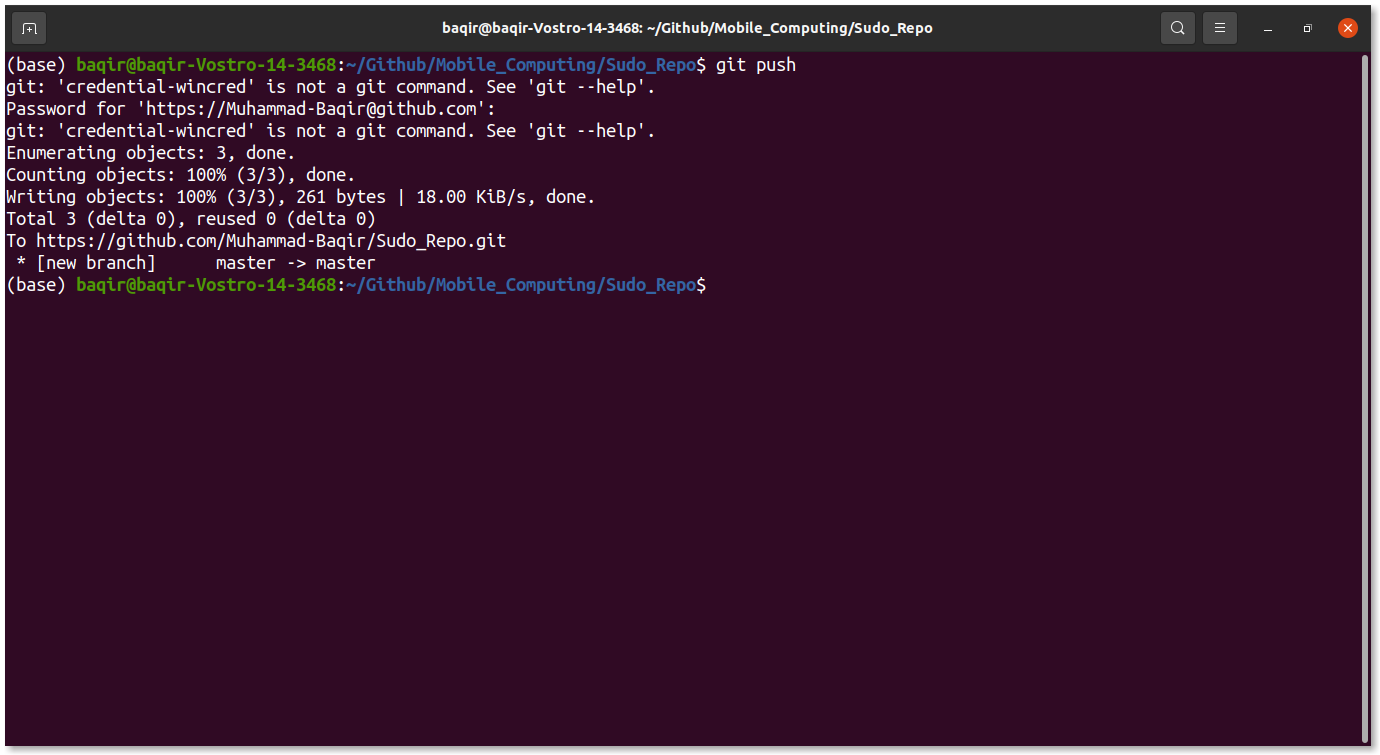
#### **Step 3.5:** So we have to inform git to track the file by entering **git add fileName** command. Enter the command and again see status. Now you can see that git is tracking the file but it is not committed.



#### **Step 3.6:** So clear the screen and enter command **git commit -m “Any meaningful message”**. And again check git status. Now you can see that there is nothing to commit.

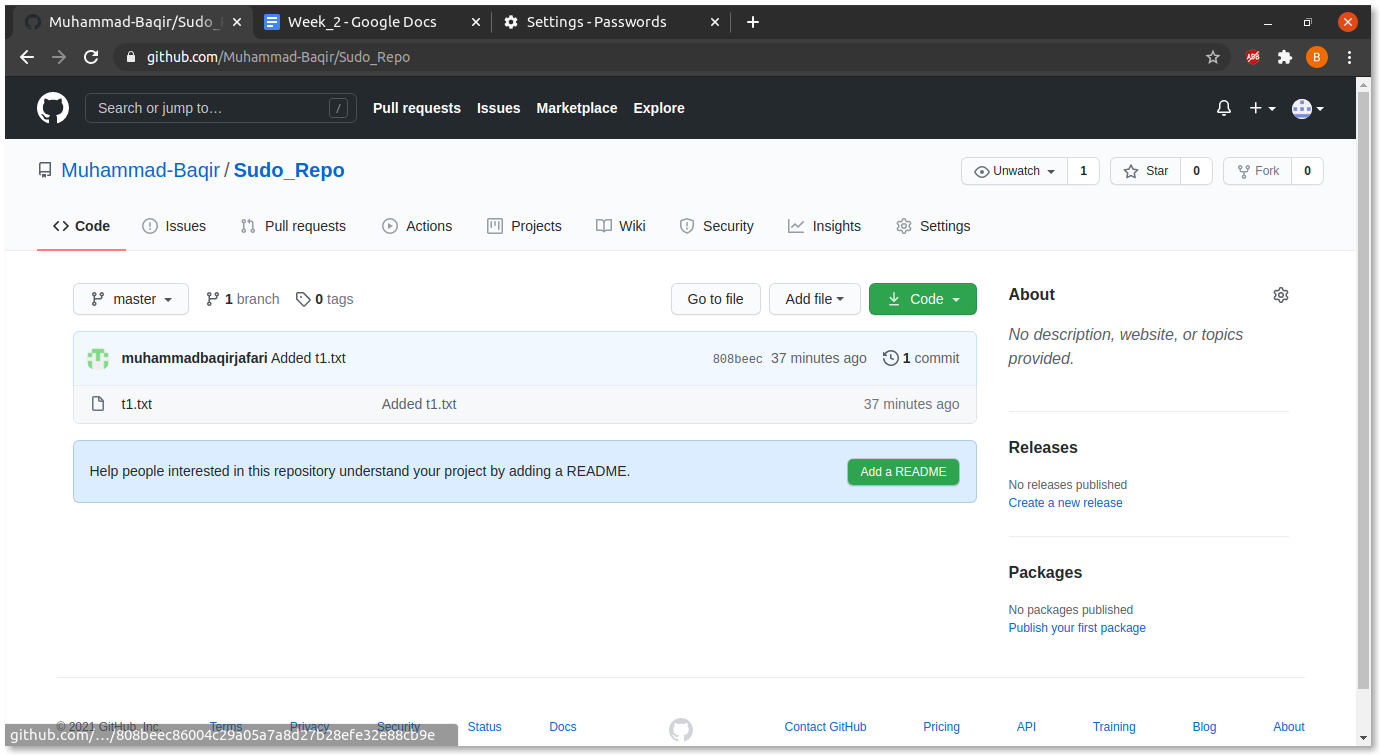


#### **Step 3.7:** Now we have committed the changes. But still these changes are not visible online. The reason is that we have to push the repo. So enter **git push** command. Now you can see changes on Github repo also.

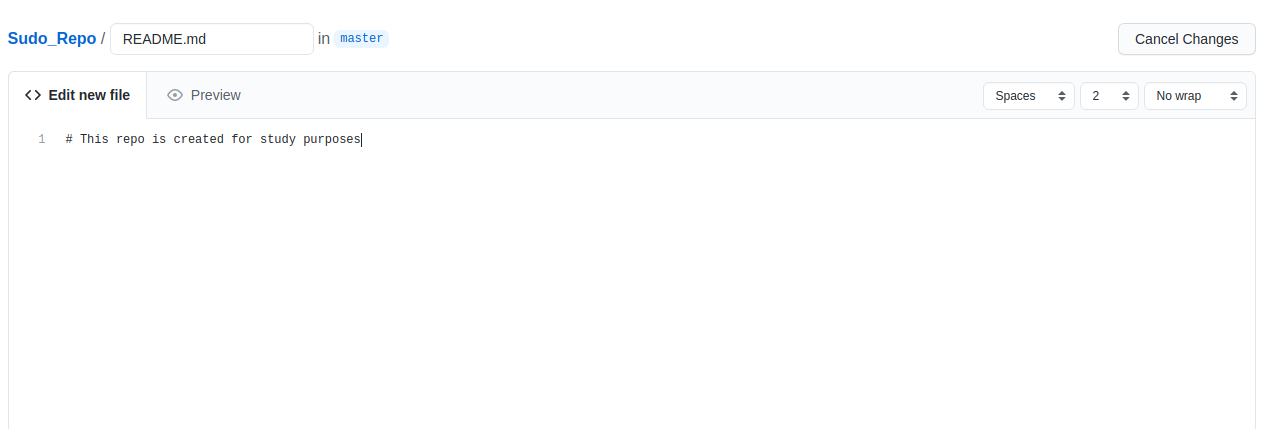


### **Step 4:** Now let’s try to add a README.md file online and pull the changes offline.

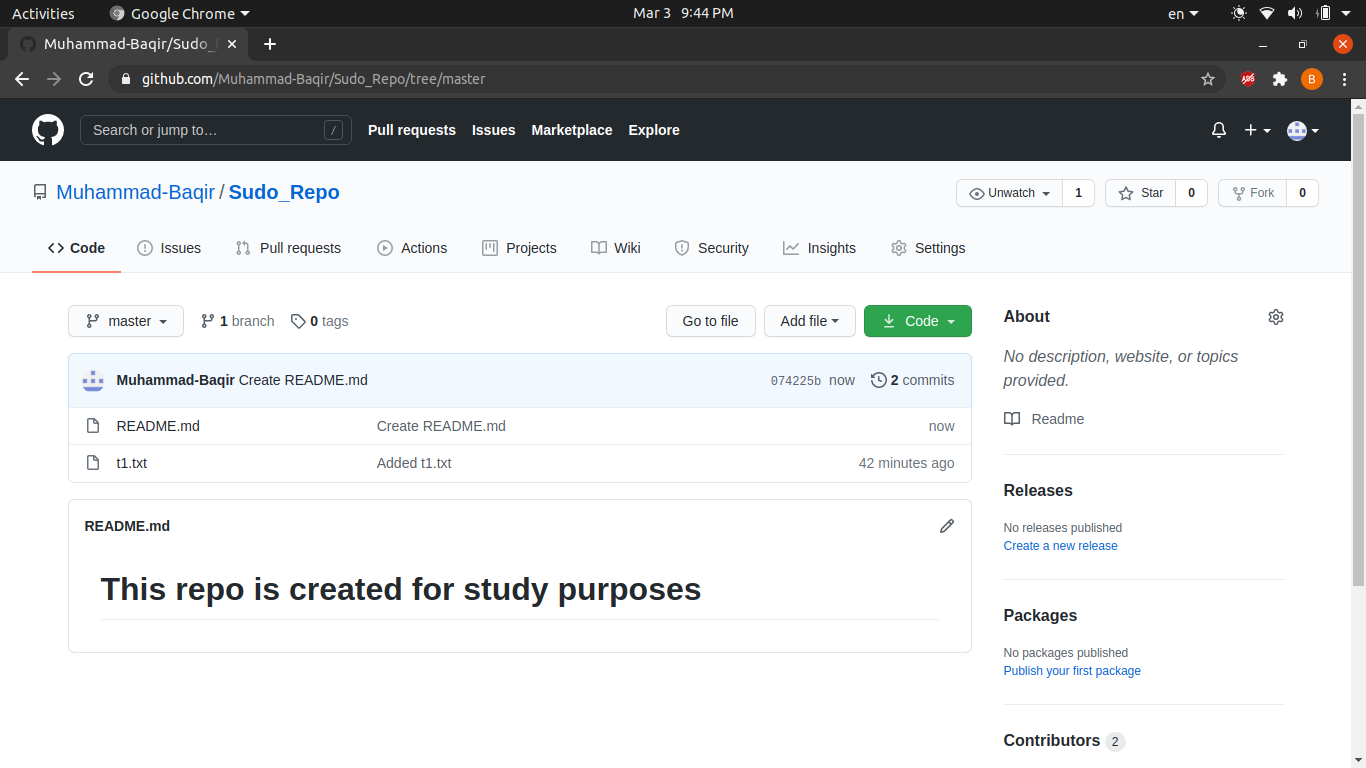
#### **Step 4.1:** Go to your Github repository homepage and click on **Add a README** button.



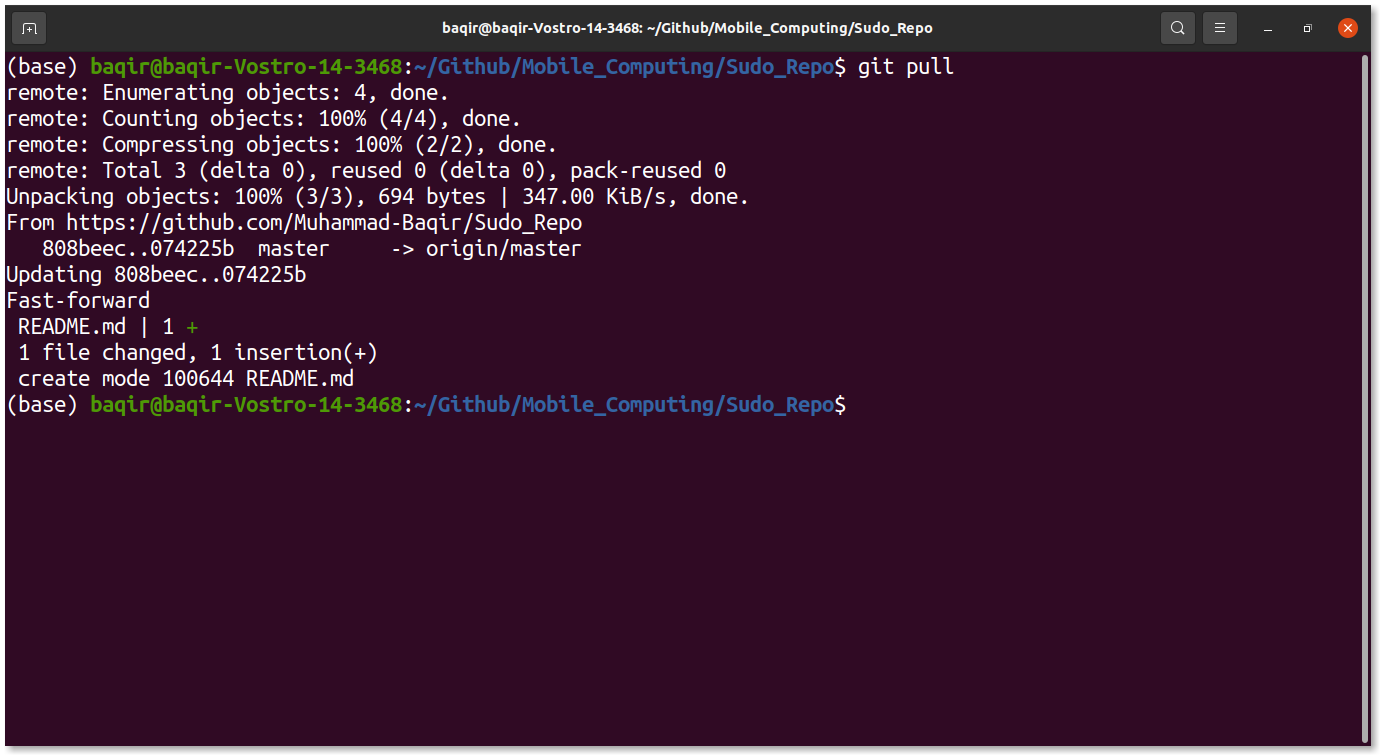
#### **Step 4.2:** You will be redirected to a new url where you can edit your newly created README file. Add some random stuff.



#### **Step 4.3:** Scroll down and click on **Commit new file** button. Now you will be redirected to the repo home page and a new README.md file is also added.



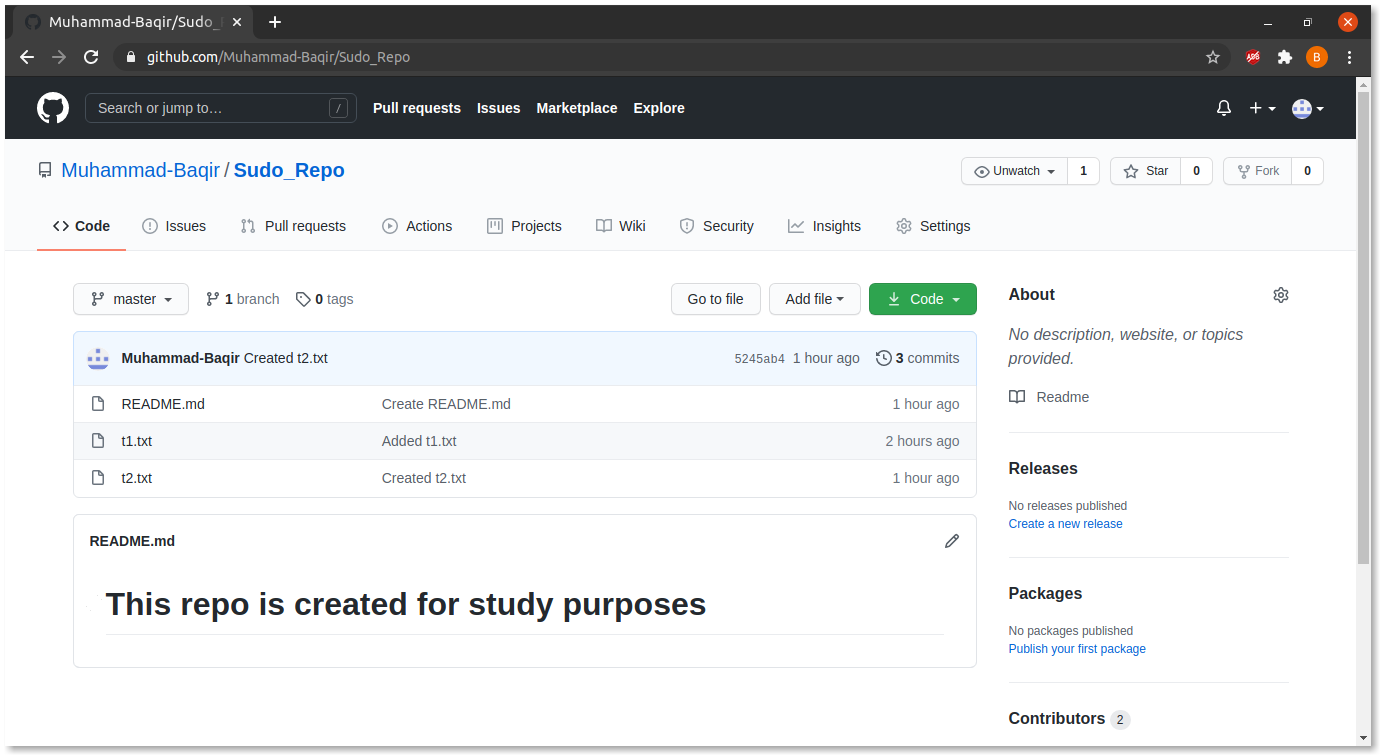
#### **Step 4.4:** To see changes offline. Go to terminal and enter **git pull** command.



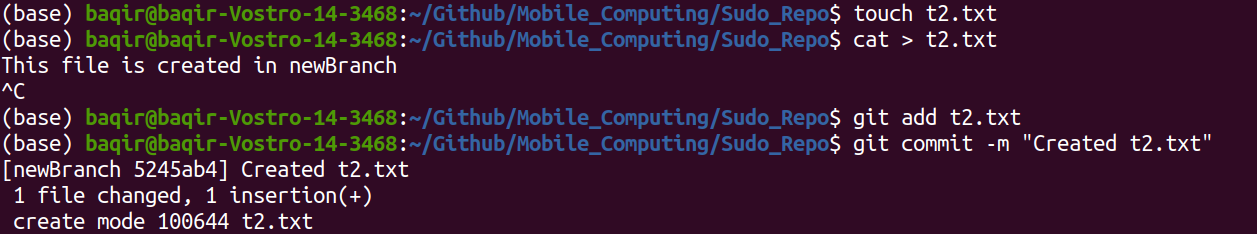
#### **Step 4.5:** Now you can see that the README file is available offline.

### **Step 5:** Now lets play with git branches.

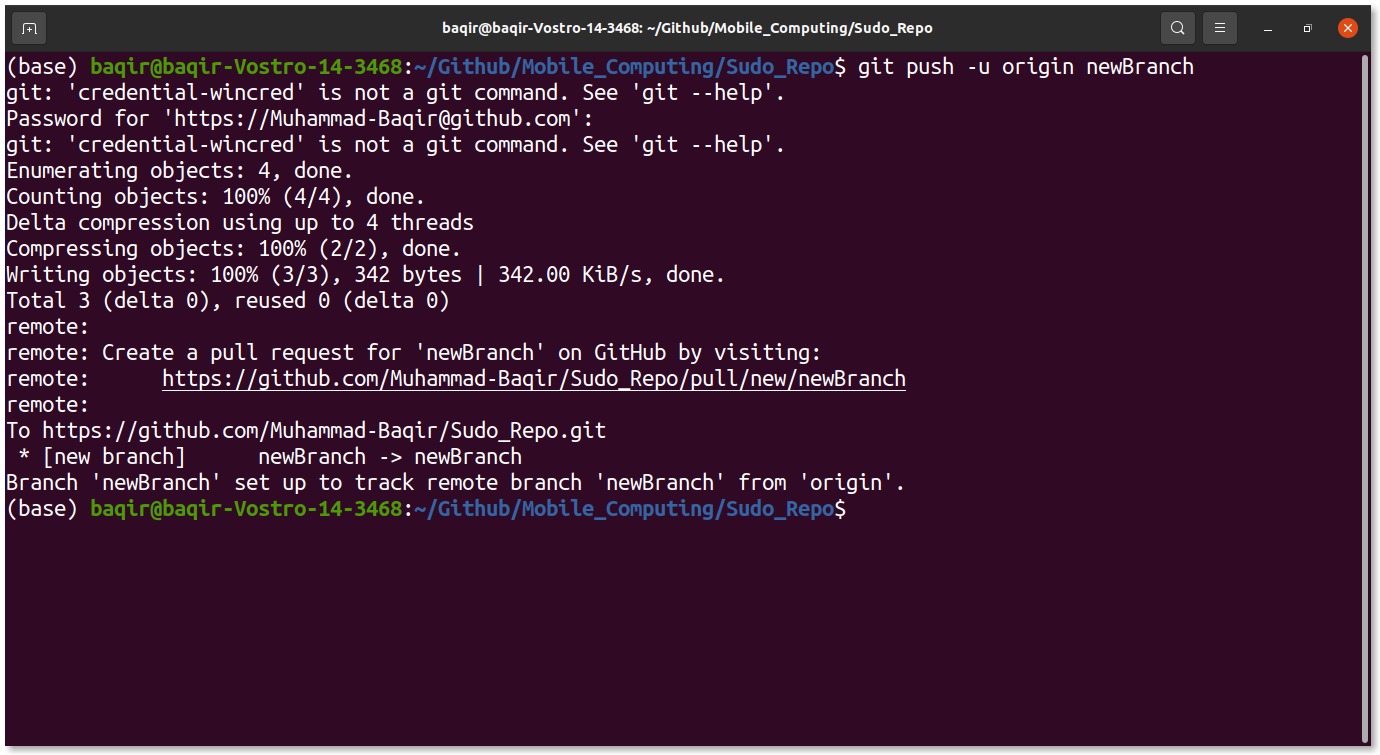
#### **Step 5.1:** In terminal enter **git checkout -b new\_branch\_name**. Now enter the git **branch** and you can see the new branch is created.



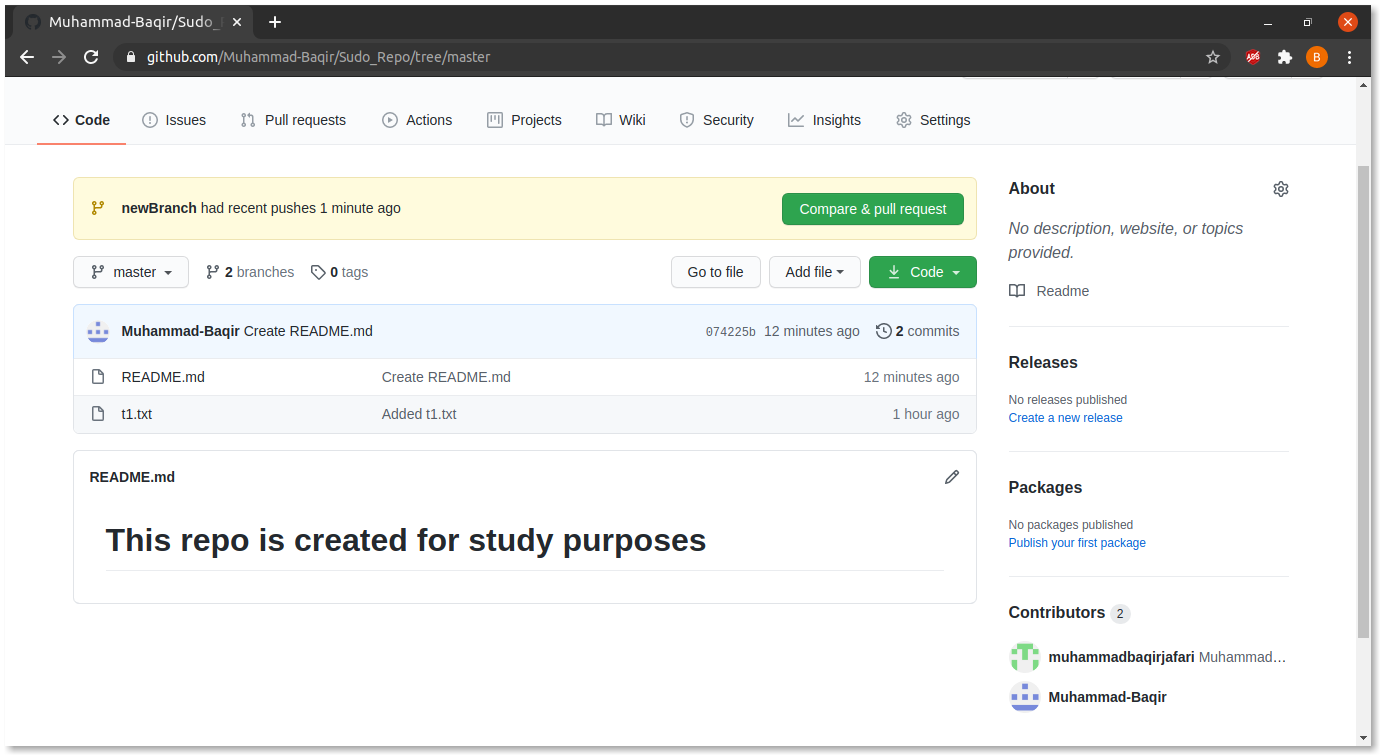
#### **Step 5.2:** Create a new file there and commit the changes.



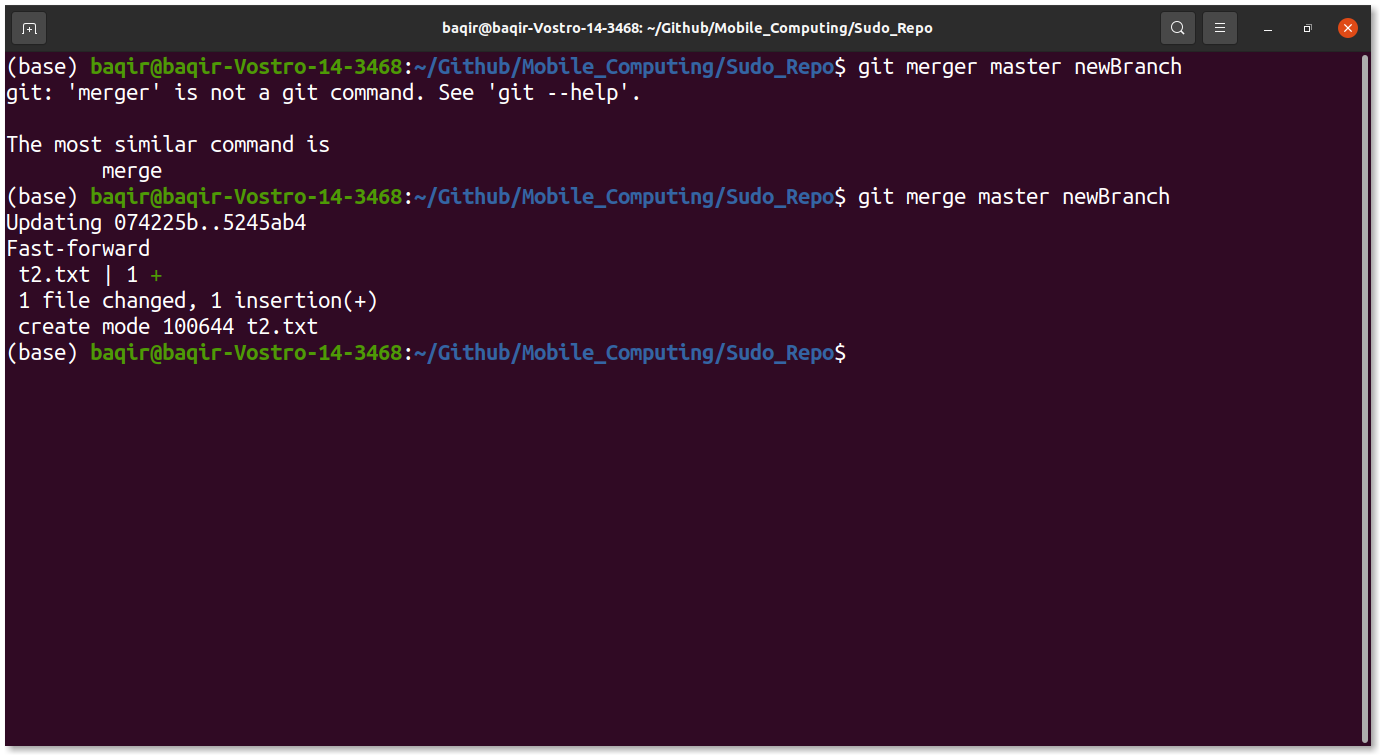
#### **Step 5.3:** Now enter **git push -u origin new\_branch\_name** to push the changes.



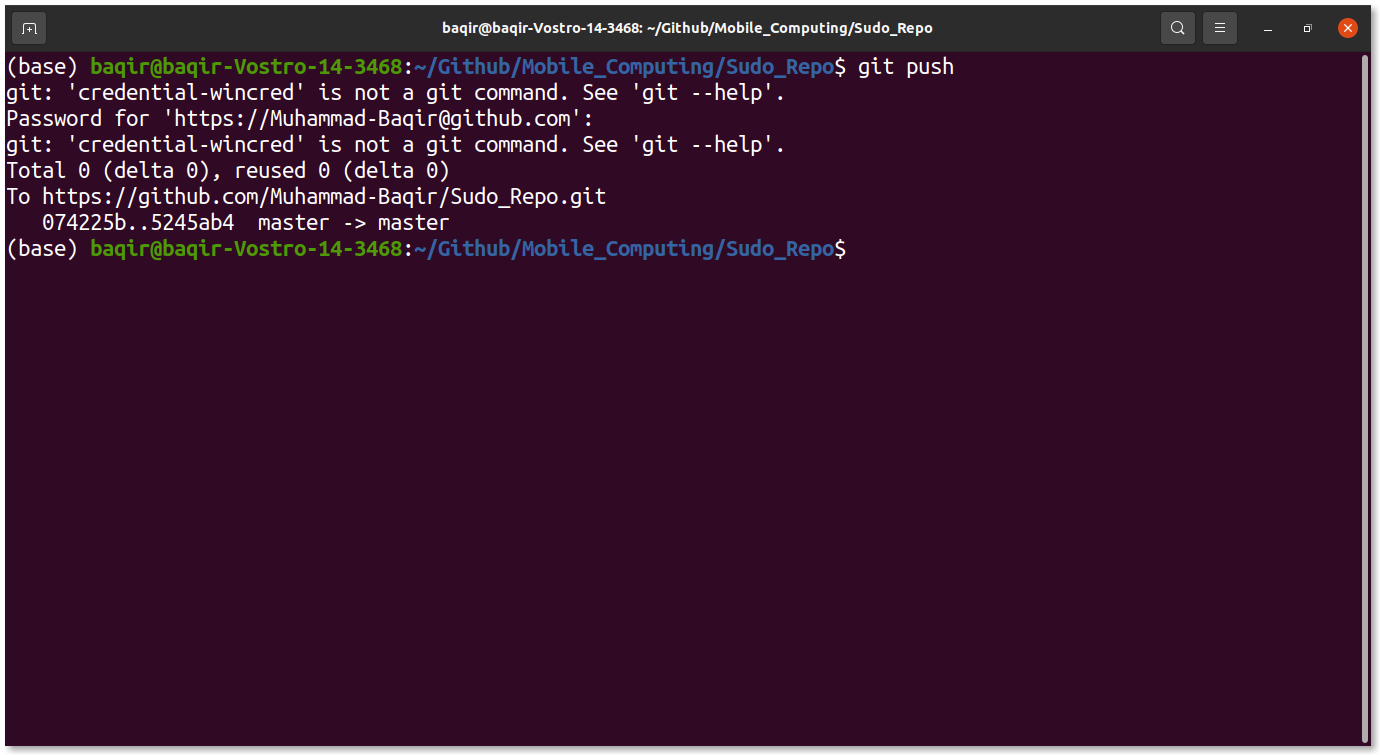
#### Step 5.4: Now you can see that there are two branches available. Now let’s try to merge newly created branch to master branch.



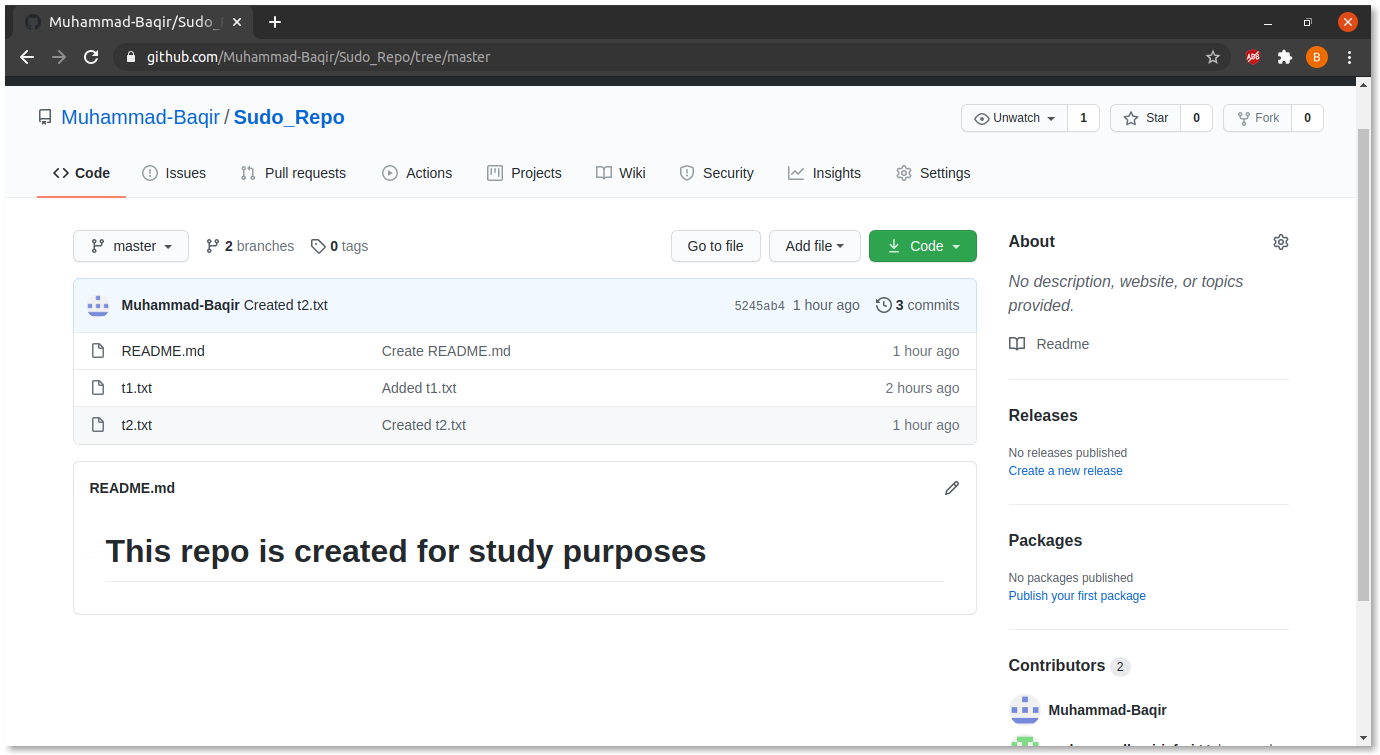
#### **Step 5.5:** In terminal enter command **git checkout master**, in order to switch to master and branch and then enter **git merge origin new\_branch\_name**.



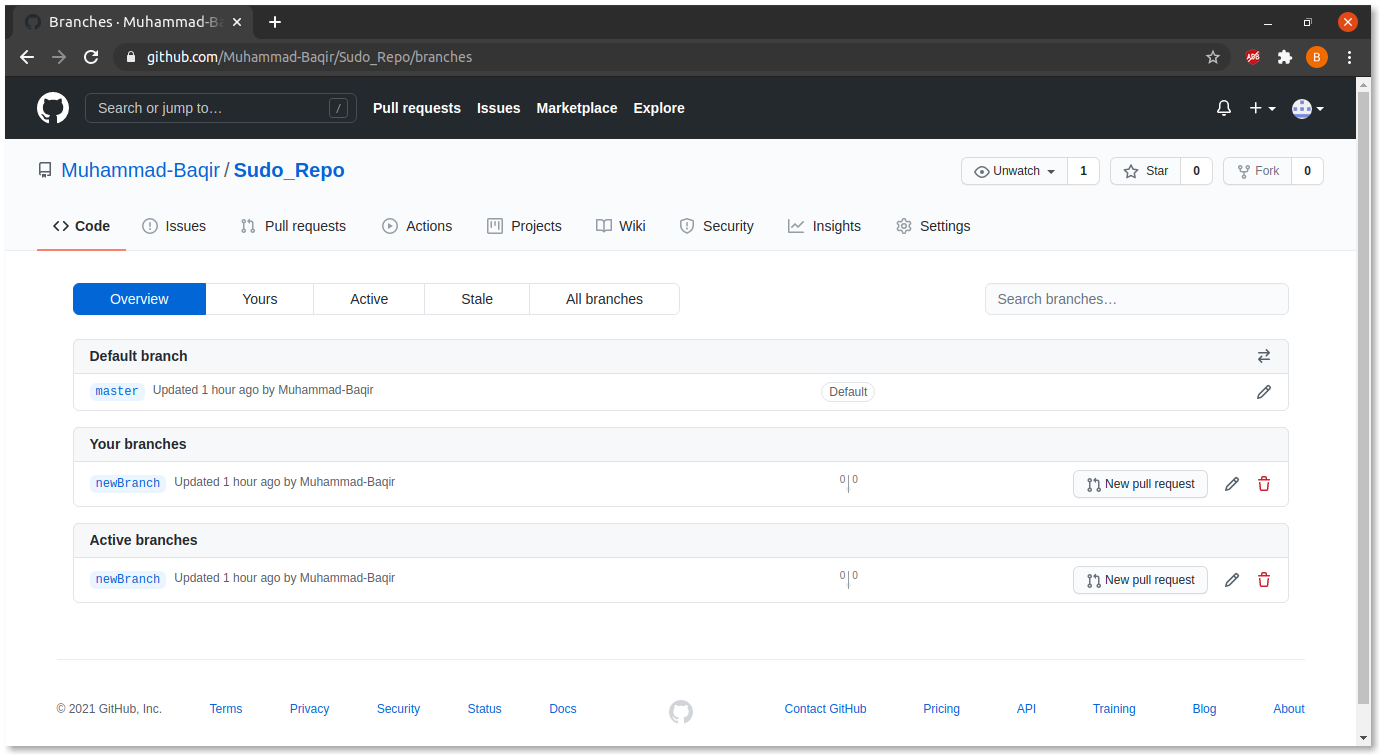
#### **Step 5.6:** Now push the changes so you can see the changes online



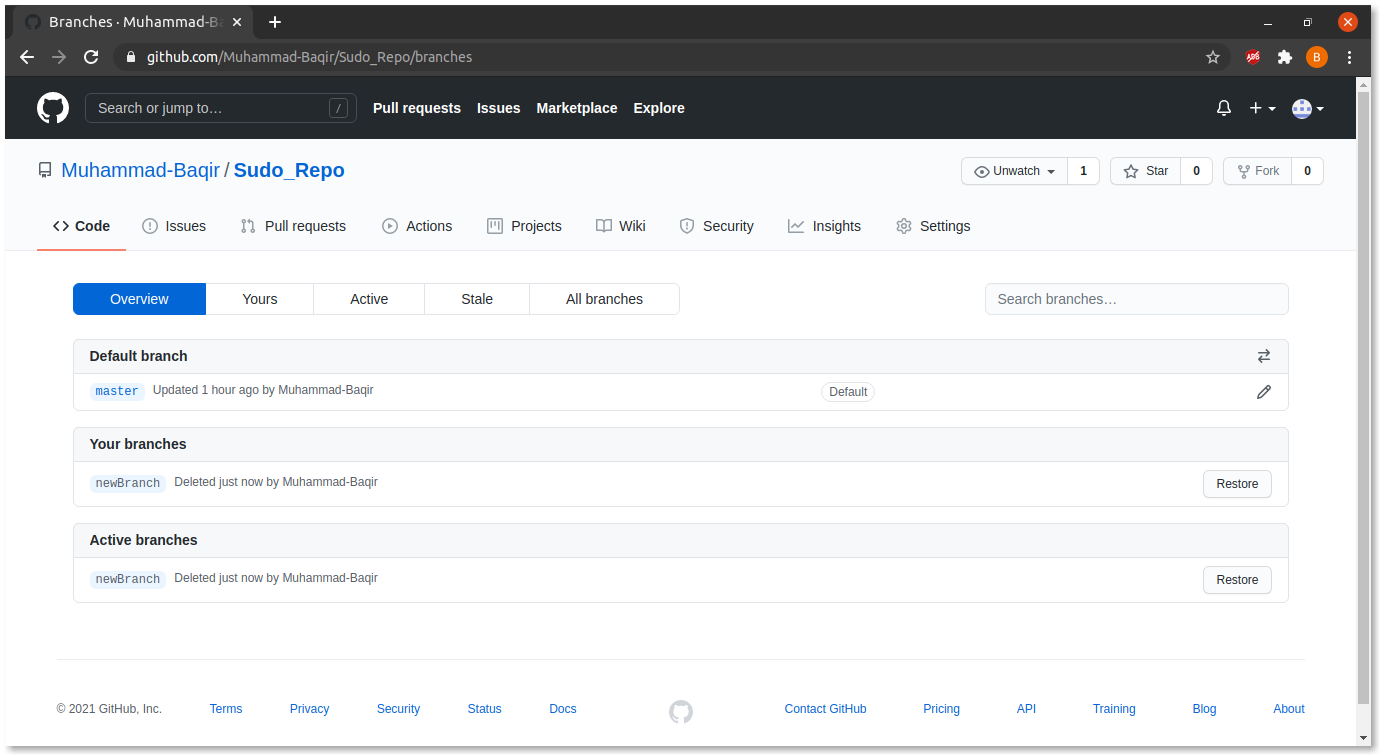
#### **Step 5.7:** You can see that the master branch also has a t2.txt file.



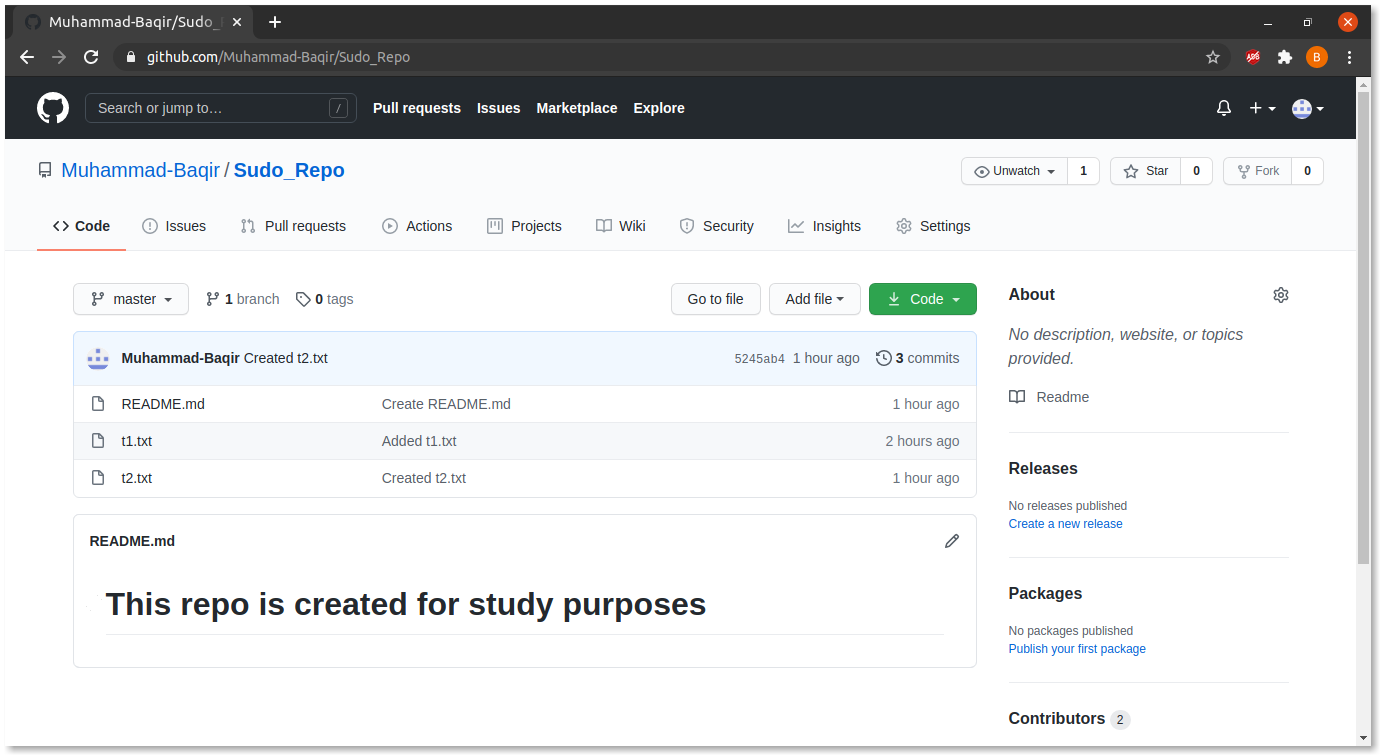
#### **Step 5.8:** You can also delete older branch by clicking on branches icon. And you will be redirected to a new page.



#### **Step 5.9:** Here you can click on delete button and your branch will be deleted.



#### **Step 5.10:** Now go to the home page and you can see only master branch is available.

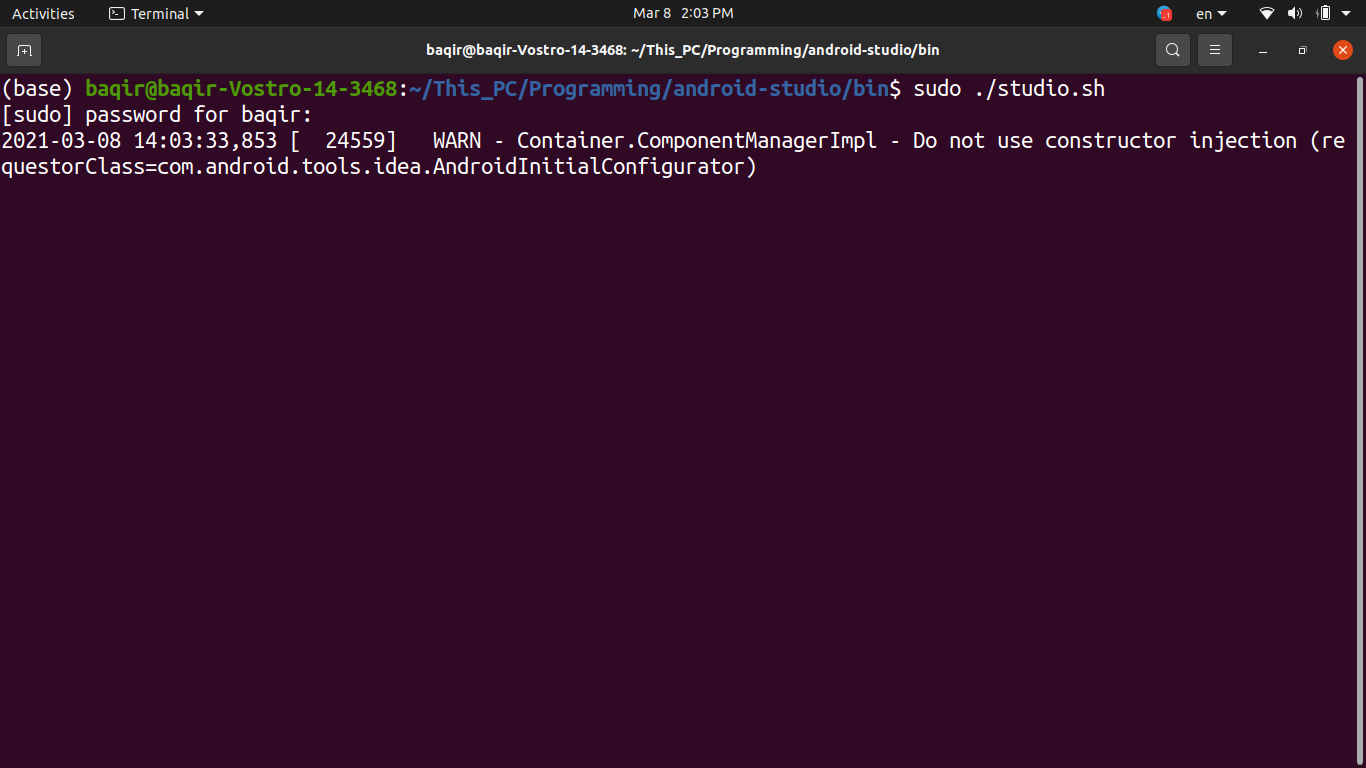


## Lecture 4 Practice

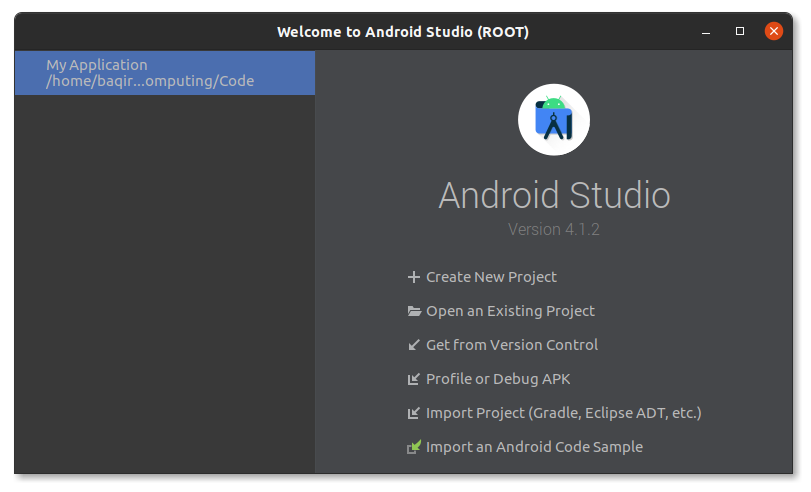
* + Now, let’s play with Android Studio.

### Step 1: Open Android Studio.

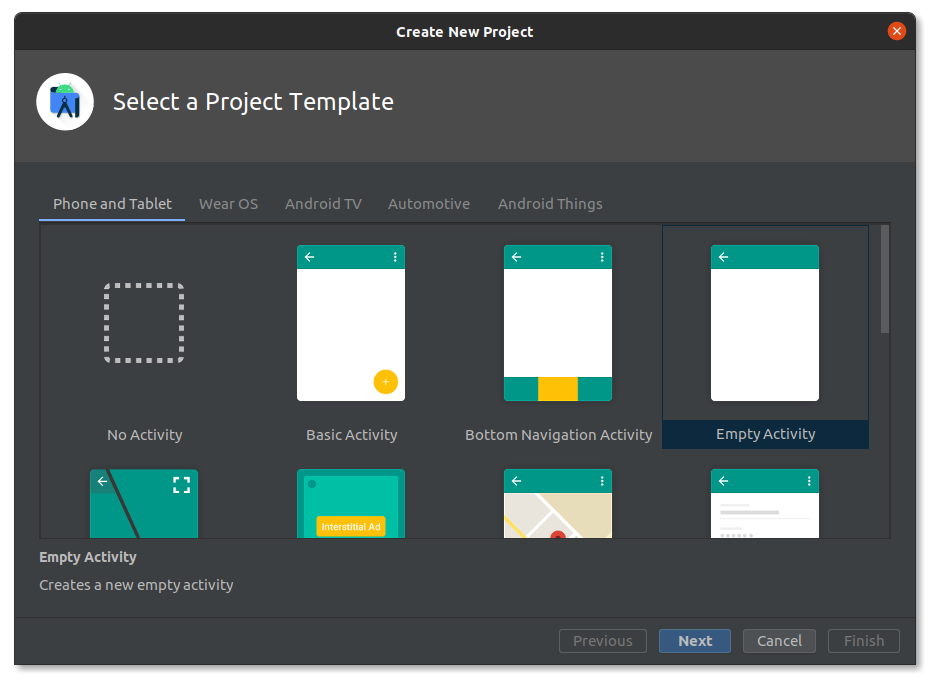
#### Step 1.1: Open the bin folder of android studio and run the following command **sudo ./studio.sh**



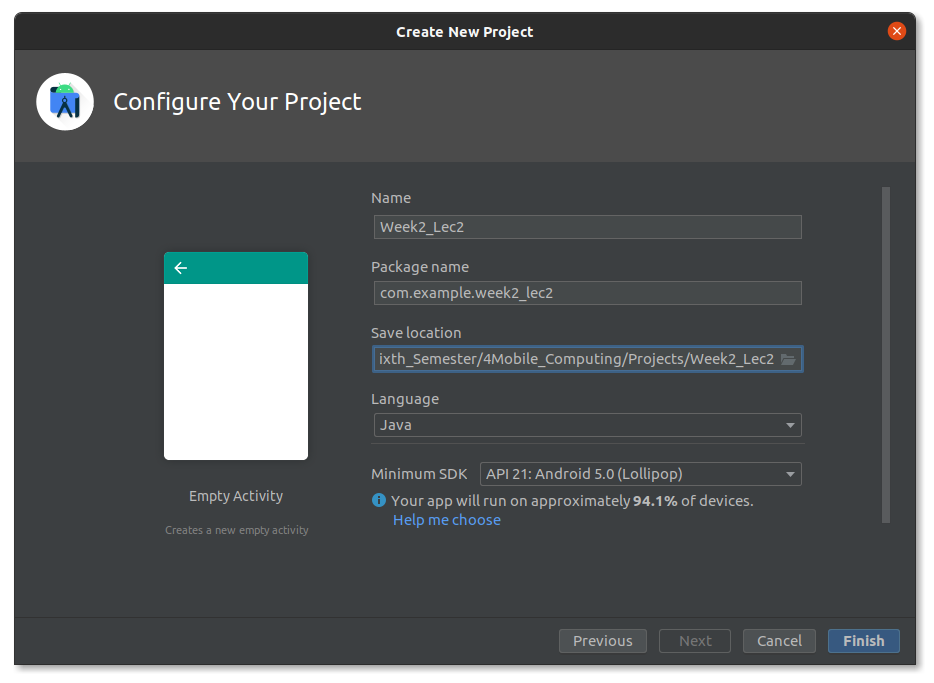
#### Step 1.2: From the menu opened click on **Create New Project**



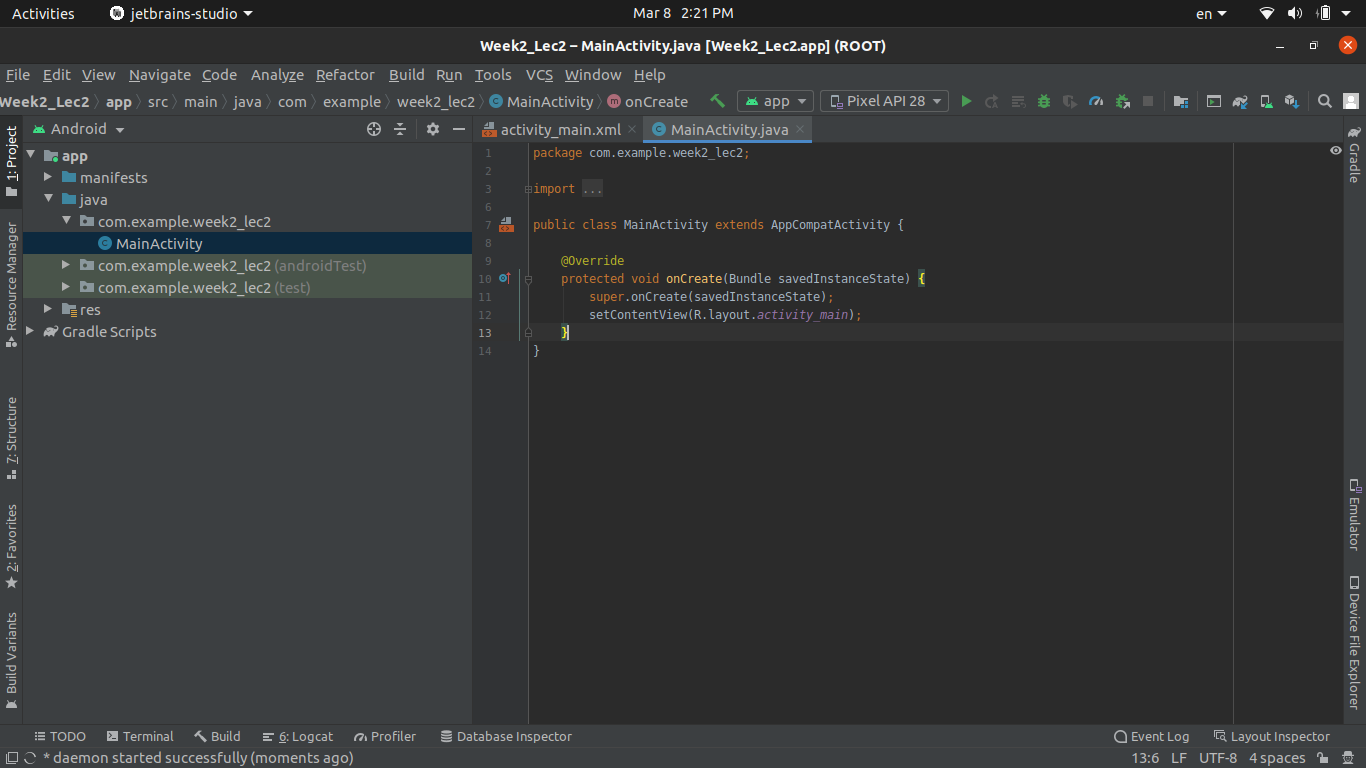
#### Step 1.3: Now select **Empty Activity** template and select **next**.



#### Step 1.4: Enter the name of the application and set its location. And click on **Finish.**

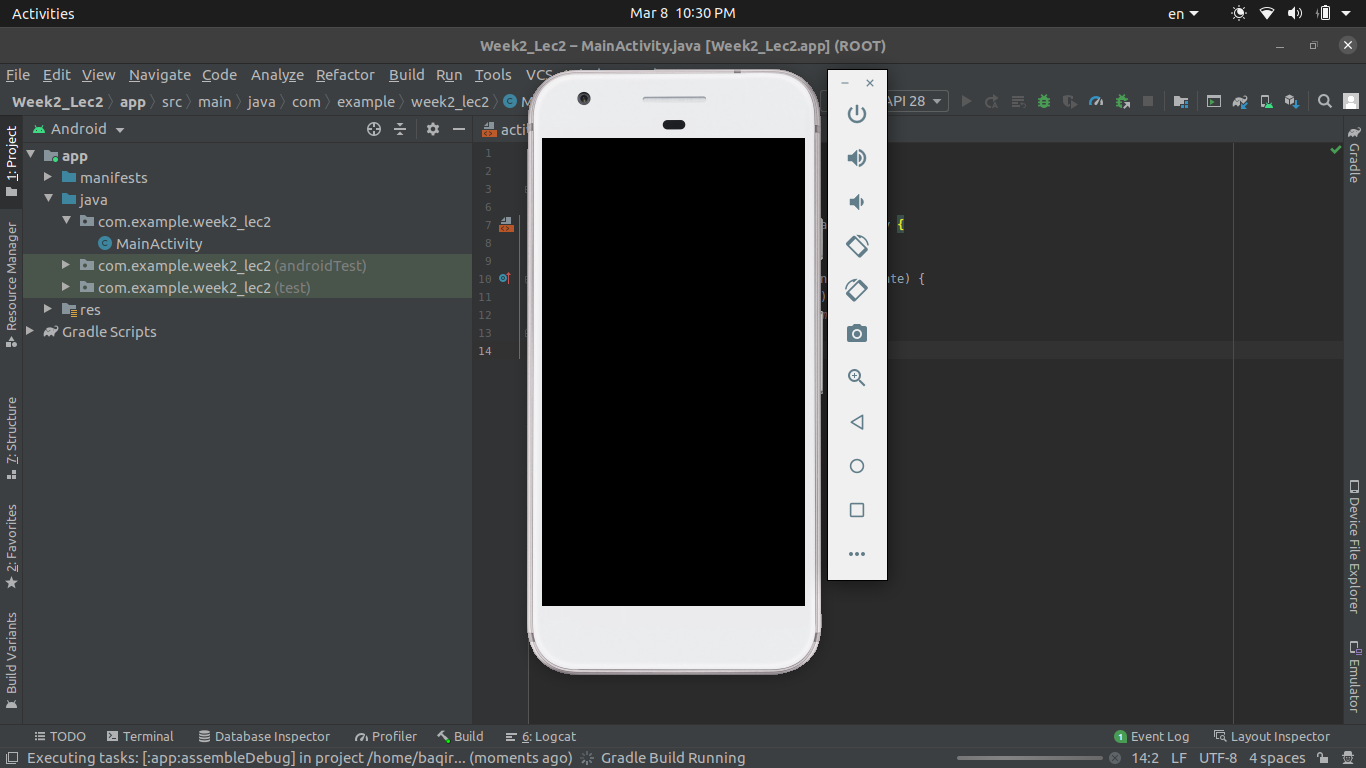


#### Step 1.5: Now wait a moment and Android Studio is ready for coding.

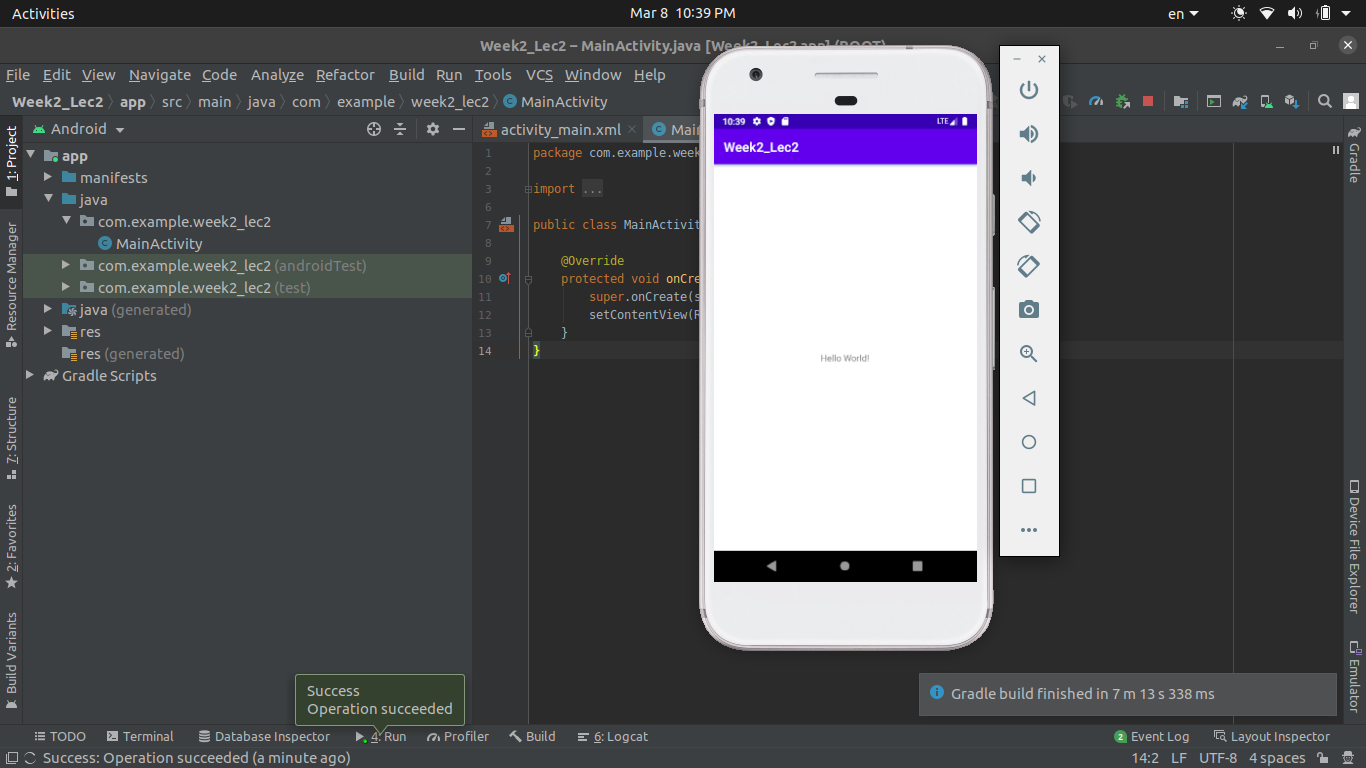


### Step 2: Run a simple Hello World App.

#### Step 2.1: Enter **Shift + F10** or click on green arrow button to build and run the app.

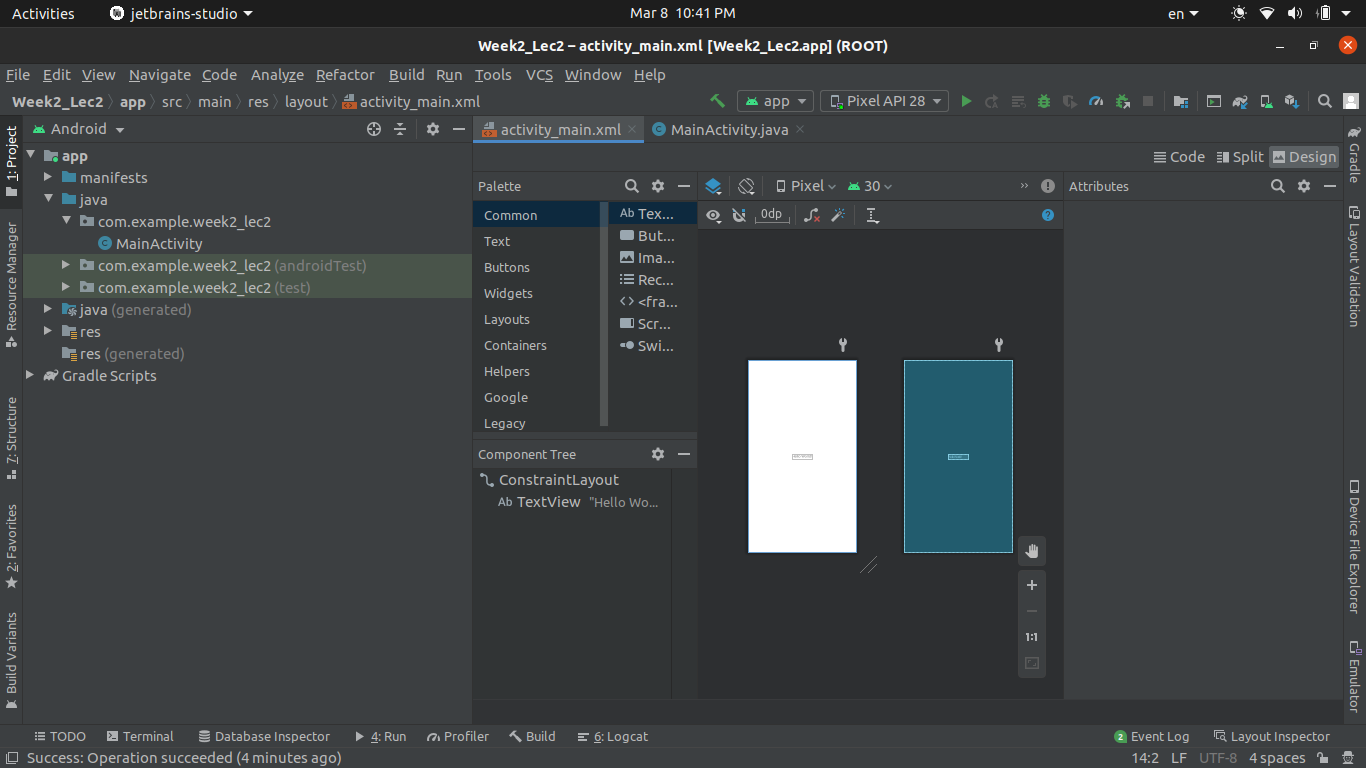


#### Step 2.2: Wait some time and you will see the Hello World app run on virtual android device.

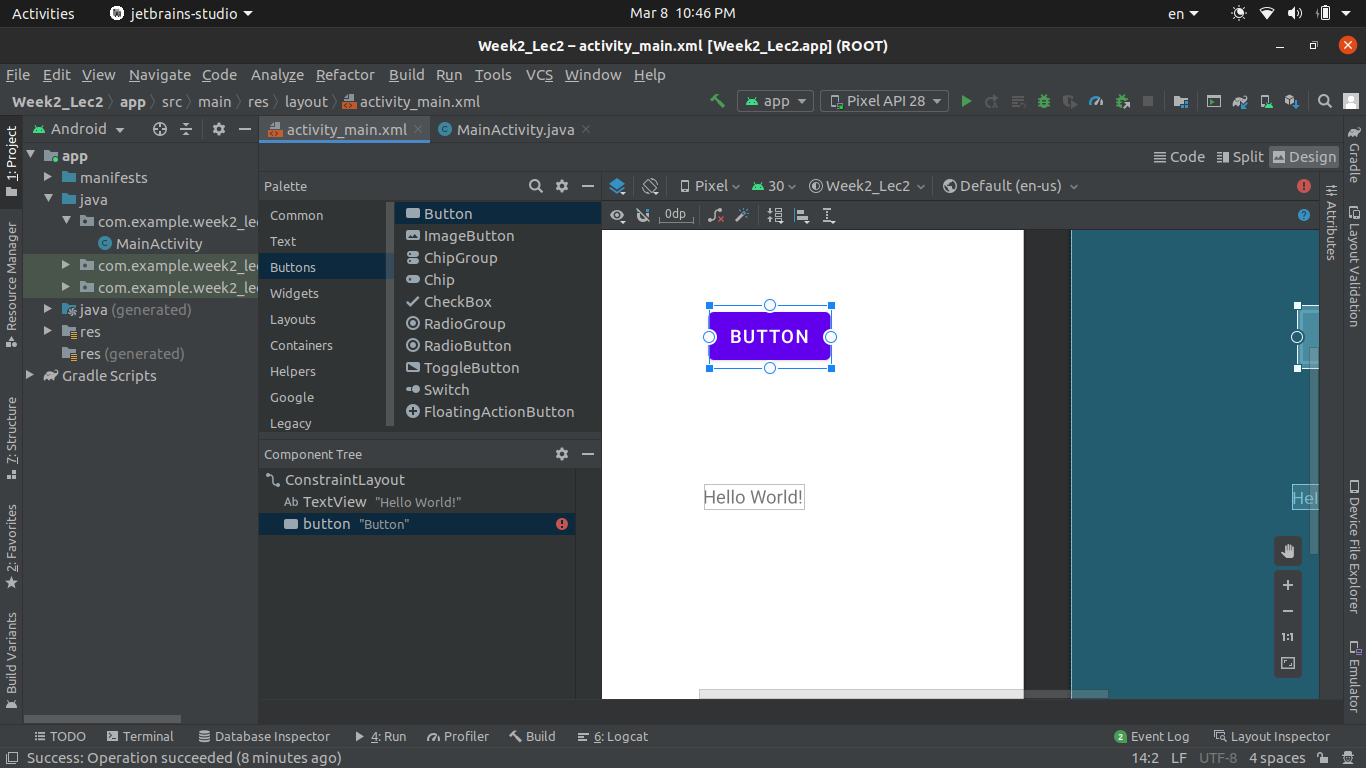


### Step 3: Add some buttons.

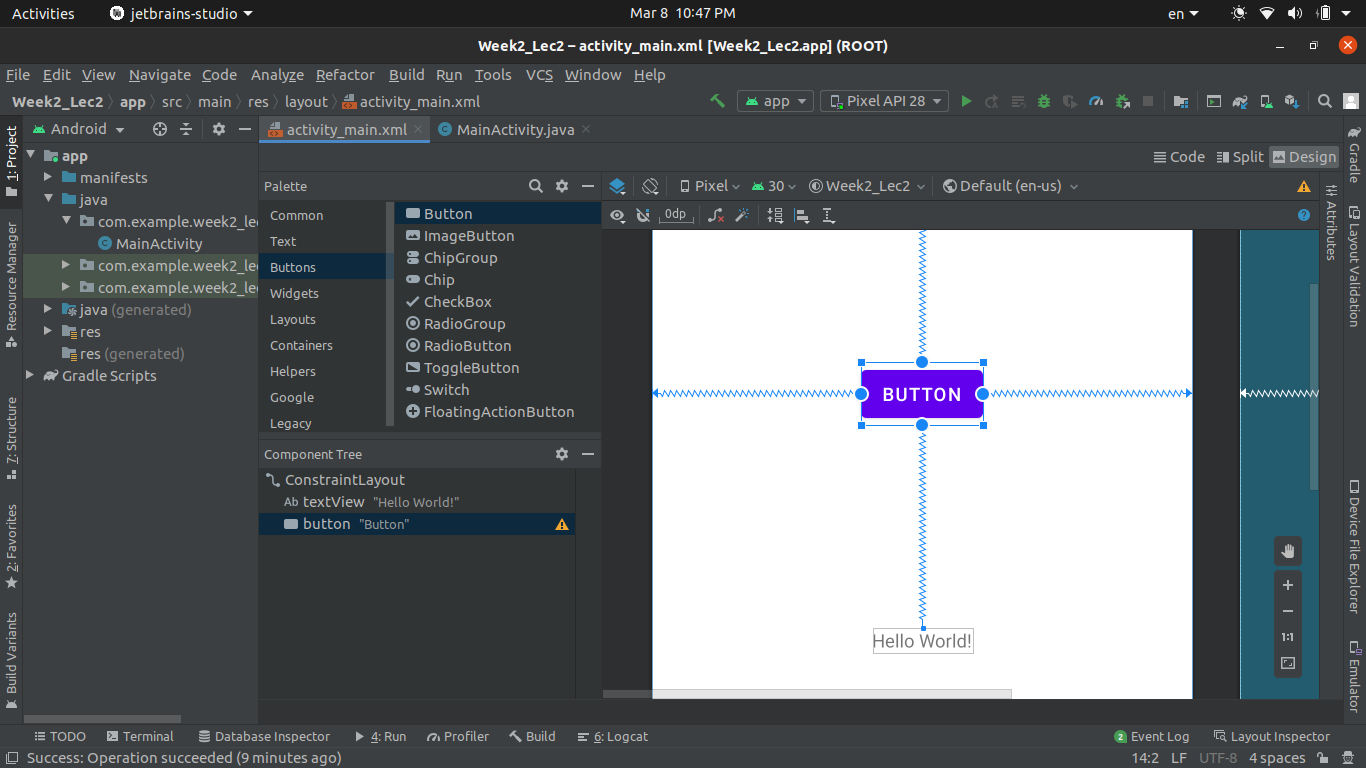
#### Step 3.1: Stop the app. Open **activity\_main.xml.**



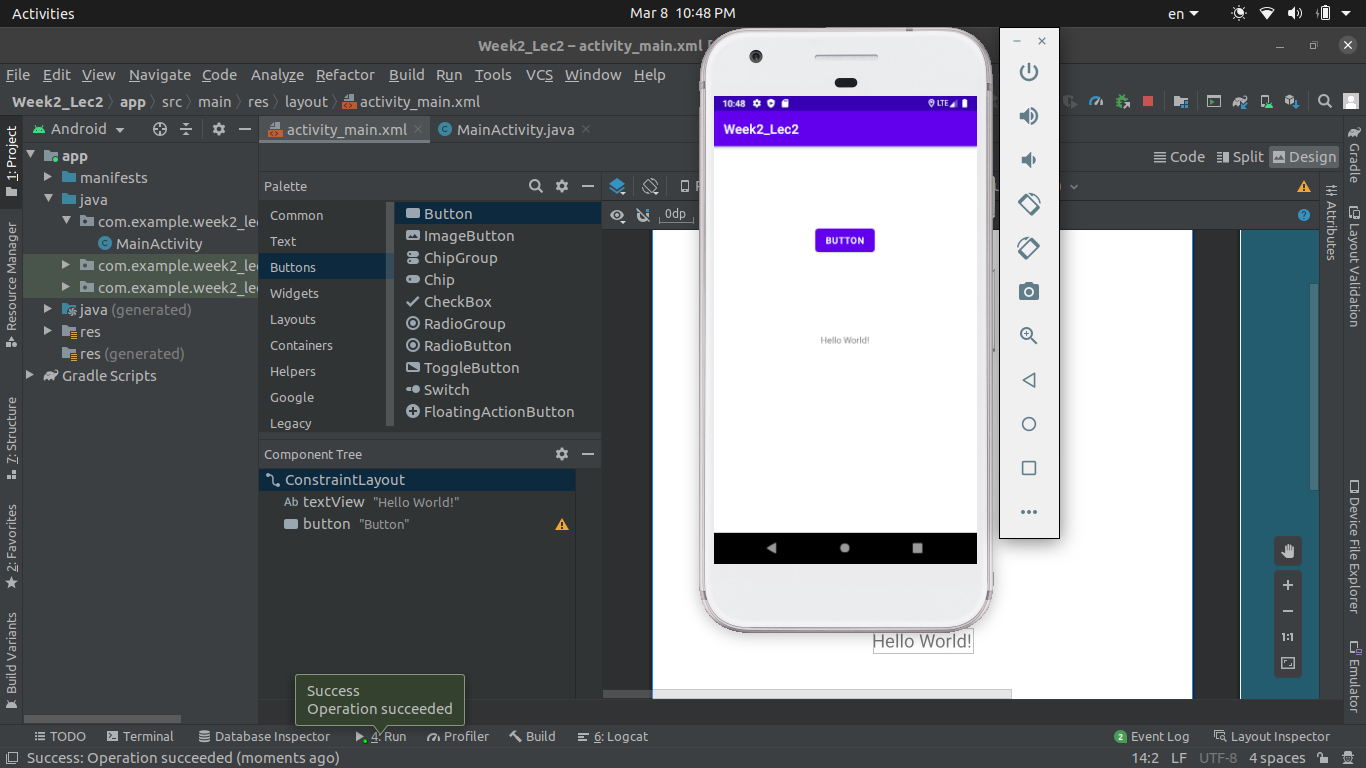
#### Step 3.2: Under the **Palette** menu choose Buttons tab. And now drag and drop simple button from the menu next to it.



#### Step 3.3: Adjust the location of the button.

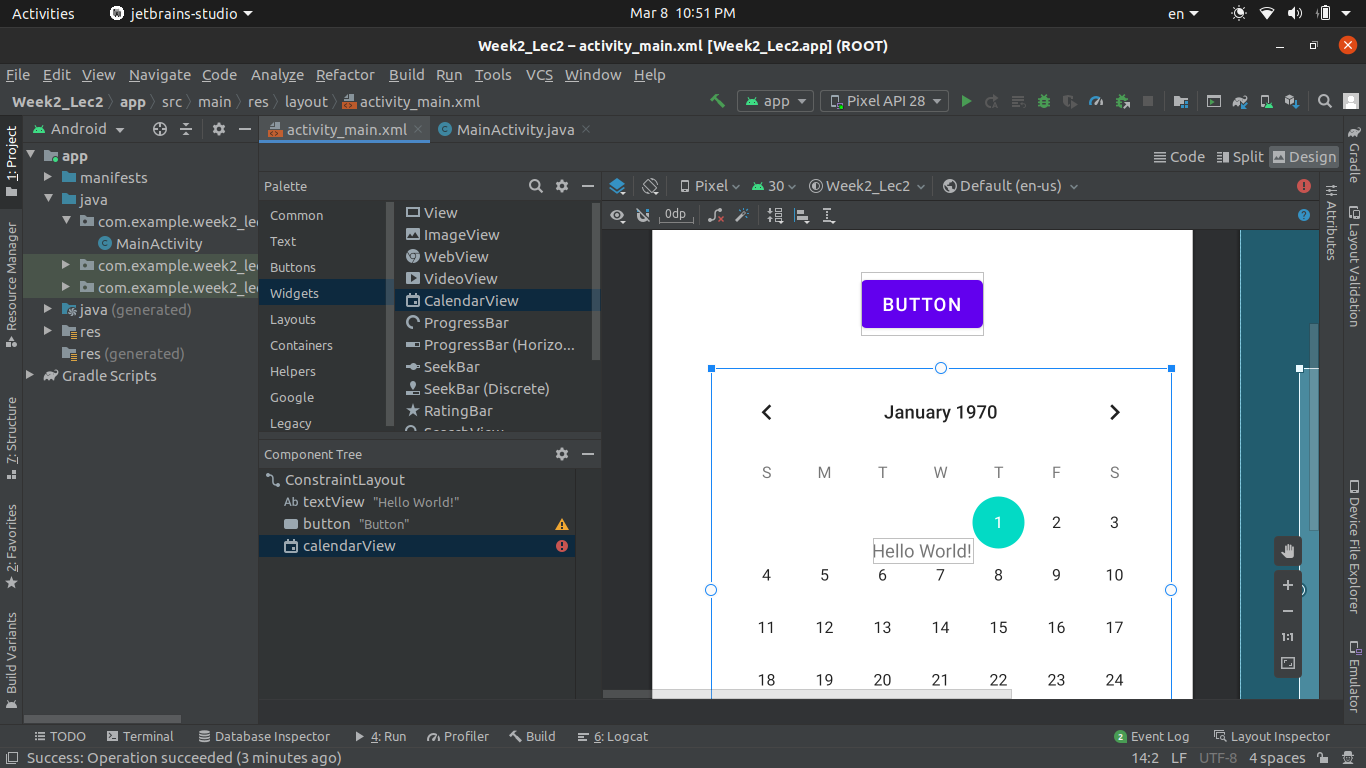


#### Step 3.4: Again run the code and you will see the output.

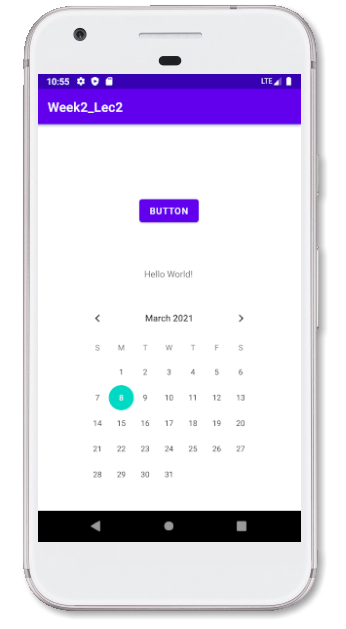


### Step 4: Add some Widgets.

#### Step 4.1: Stop the app. From **Palette** select Widgets. And drag and drop calendar widget.

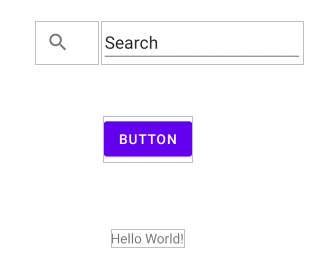


#### Step 4.2: Adjust calendar and run the app.



### Step 5: Add some more Widgets.

#### Step 5.1: From widgets drag and drop search widget.



#### Step 5.2: Build and Run the app and see the output.

