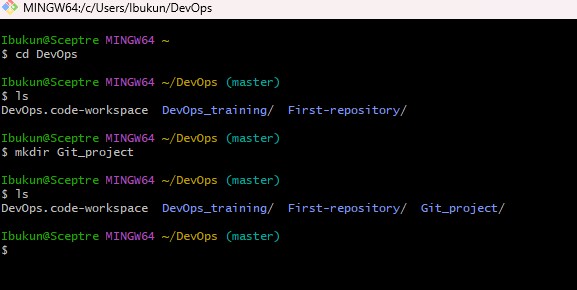
**Git project**

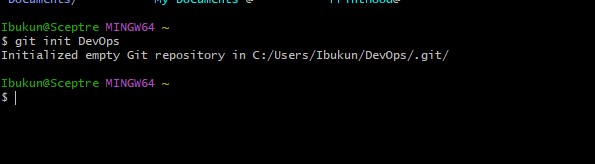
Initializing a git Repository

To start initializing git repo, git must have been installed on our machine.

Follow below order

* 1. Open a terminal on the computer e.g Git bash.
  2. Enter cd DevOps from home.
  3. On open terminal create a folder or directory e.g Git\_Project with mkdir command {**mkdir Git\_Project**} inside existing DevOps folder.
  4. On the terminal, press **git init** for initialization of our repository





From above screenshot, we see have been able to create working directory and initialized git repository. With this we can proceed and make our first commit in git.

Commit means saving any changes made to my file, changes like add, delete and file or text modification.

When we make a commit, Git takes a snapshot of the current state of our repository and save a copy in git folder inside our current working directory.

We can create our First commit now follow this order.

Inside my working directory, I create a new file called index.txt using touch command

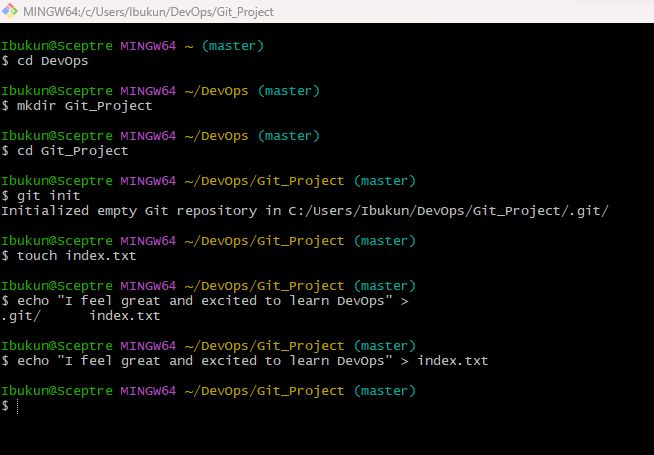
{**touch index.txt**}.

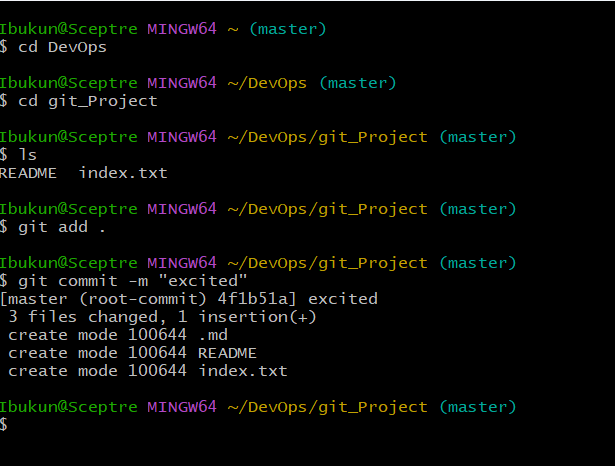
Then inside the index file created input some words on the blank file, then save.

Next step is to add our change git to staging area by using **git add** . command.

To commit the change we made to git, use the git commit -m command “initial commit”

{**git commit -m “initial commit”}** -m is a flag to add a commit message that gives us context of commit or hint about commit.





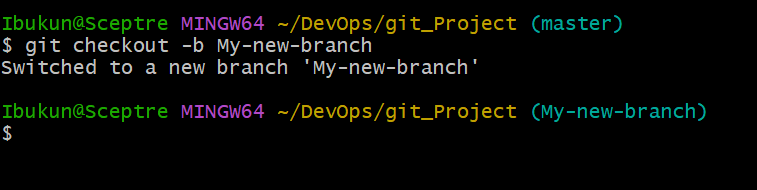
working with branches

Git branch Is an important tool for collaboration within remote team

Git branch is commonly used to develop new feature of our application without altering the main code

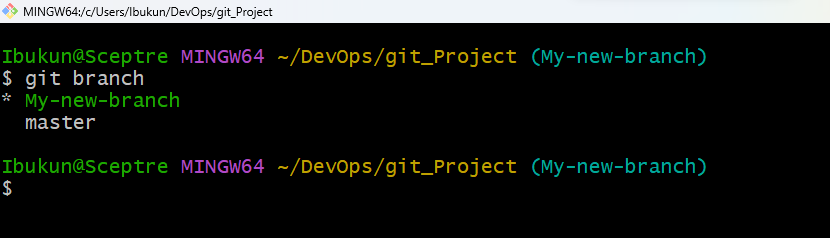
Make our first branch code, follow below steps

The git branch run on this command **git checkout -b my- new- branch**



-b is flag that help us to create and change into new directory

Listing our git branch, to show the list of our git branch run this command **git branch**



To change into an existing or old branch use this command **git checkout {branch name**}

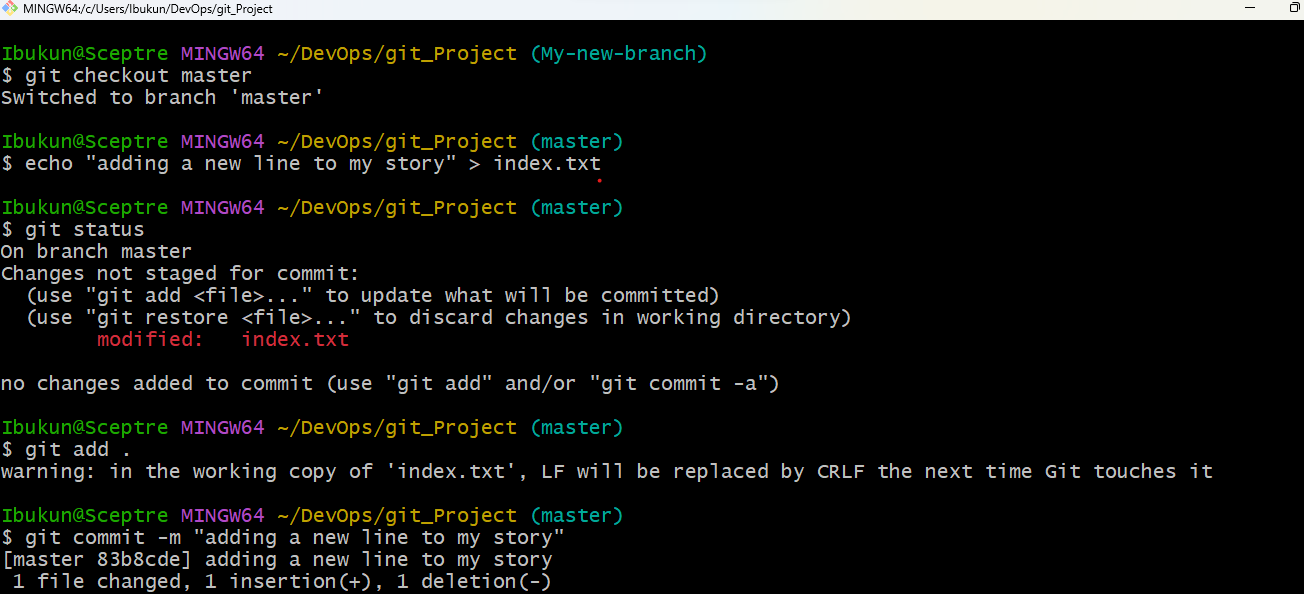


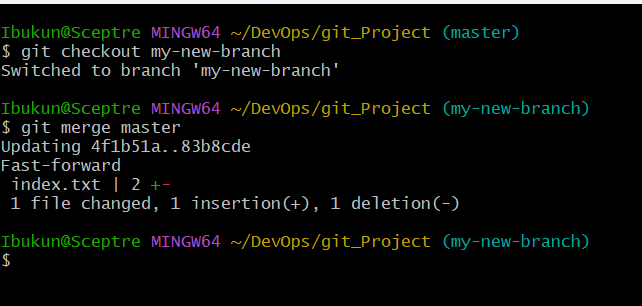
Merging a branch into another branch

Assuming we have 2 branches A & B and we want to add the content of branch B into A

First thing is to change into branch A and run the git command. Run this command

**Git merge b**

****

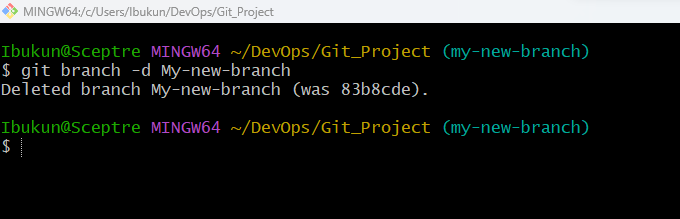
****

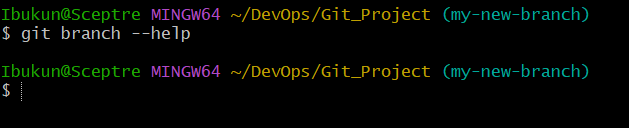
Deleting a git branch

When a new feature is added to an application it is mostly done in a feature branch (clone format) and the cloned branch is deleted once the feature is tested and merged to main branch.

Git branch is deleted with this command git branch -d {branch name}

**Git branch -d My-new-branch**

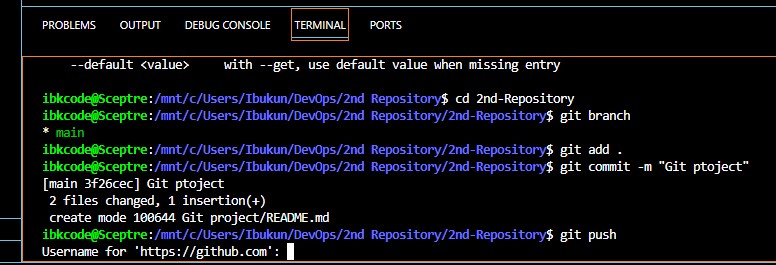




Git help shows us the different ways of using command shows all the available options for the specific commands

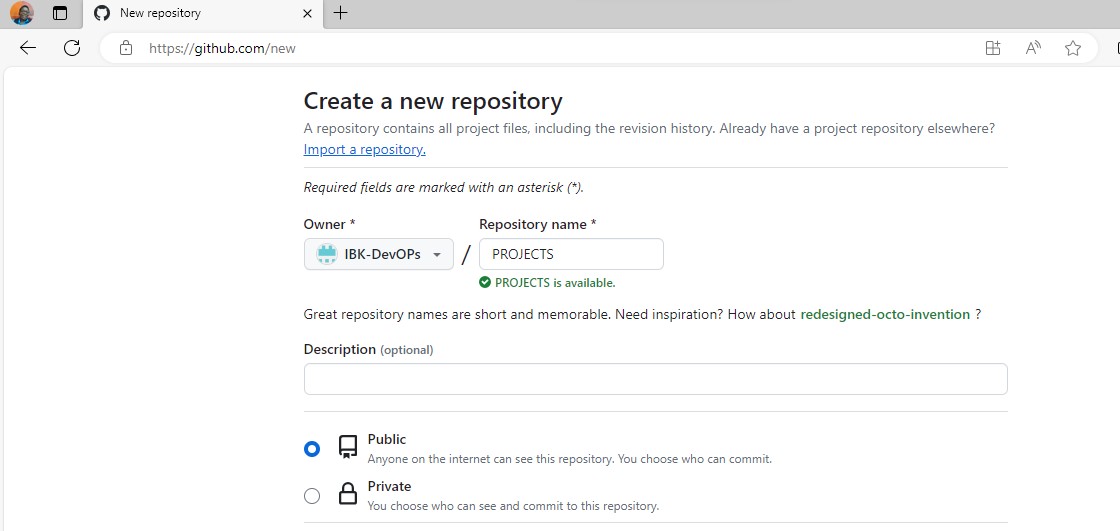
## PUSHING LOCAL GIT REPOSITORY TO REMOTE GITHUB REPOSITORY

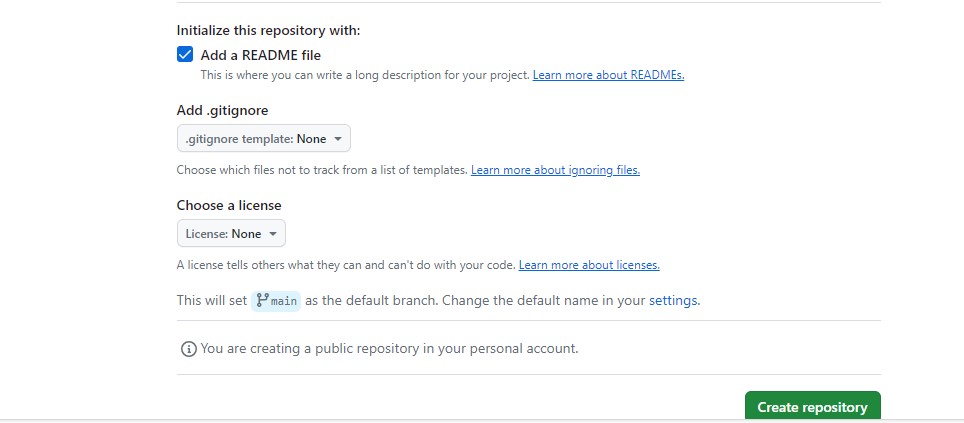
After writing and commit our story on local repository we must push orr changes to the remote repository with the command. git remote add origin [link-to-github-repo]

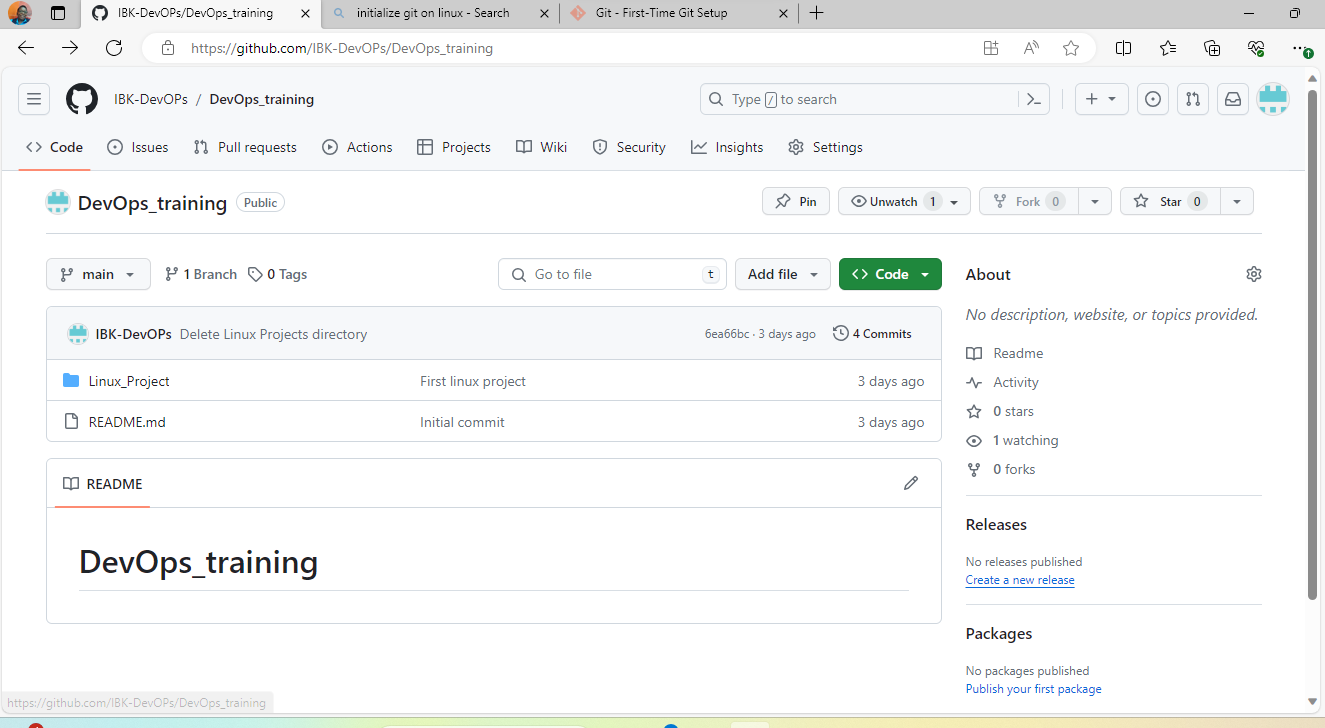


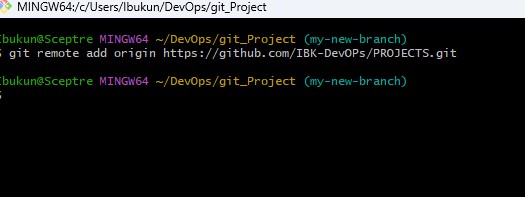


## CLONING REMOTE GIT REPOSITORY

* To clone a git repository, we use the command Git clone { link-to-remote-repository}
* COLLABORATION AND REMOTE REPOSITORIES
* Creating a git hub account
* Follow this step to create github account.
* Join github.com to create a Github account.
* Enter our username, password and email.
* Next you verify your identity with verify button.
* Follow by create a user account.
* Input activation code sent to our provided email into the box on the portal
* Select our preferences and click continue.
* Click continue for free for the free tier.
* Create new repository
* 





* PUSHING LOCAL GIT REPOSITORY TO REMOTE GITHUB REPOSITORY
* After writing all our story on out local git repository, for our friend or team to contribute to our story we have to send a copy to our repository on git hub
* All the changes made on our local repository must be pushed to the remote repository with the command this command; git remote add origin {link-to-git hub-repo}
* To get a remote link, click on the green button code and copy http link.
* 

After committing our changes in my local repo, we have to push the content to the remote repo using below command. Git push origin < branch name>

**Input git push origin main**

* 

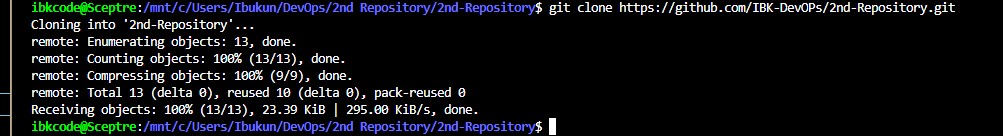
CLONING REMOTE REPOSITORY

To clone a git repository, we use the below command

Git clone is used to copy what we have on our remote repository in our local machine.

It also serves as tools used for downloading remote repository into our local machine.

**git clone [link-to-remote-repository]**



BRANCH MANAGEMENT AND TAGGING

Markdown syntax is commonly used for creating document, README files, forum posts and even web pages.

1. Heading: To create headings, we use the hash (#) symbol, e.g.

# Heading 1

## ## Heading 2

### Heading 3



1. Emphasis: To create emphasis, the asterisks (\*) or underscore (\_) is used e.g.

\*Italic\* or \_Italic Text\_

\*\*Bold Text\*\* \_Bold Text\_

1. Lists: To create an unordered list, we use the hyphen (-) symbol e.g.

unordered list

-Unordered list 1

-Unordered list 2

-Unordered list 3

ordered list

1. First item

2. Second item

3. Third item

4. Links: To create hyperlinks, we use square brackets [ ] for the link text, followed by the parenthesis () containing the URL e.g. [visit darey.io] (https://www.darey.io)

5. Images: To display an image, use an exclamation mark (!) followed by a square bracket [ ] for the alt text and the parenthesis for the image URL

! [ git 9]( <https://github.com/IBK-DevOPs/2nd-Repository.git>).

1. Code: To display codes or snippets, we use backicks (``) to enclose the code e.g. console.log('Welcome to darey.io')