Creating a account in Git Hub

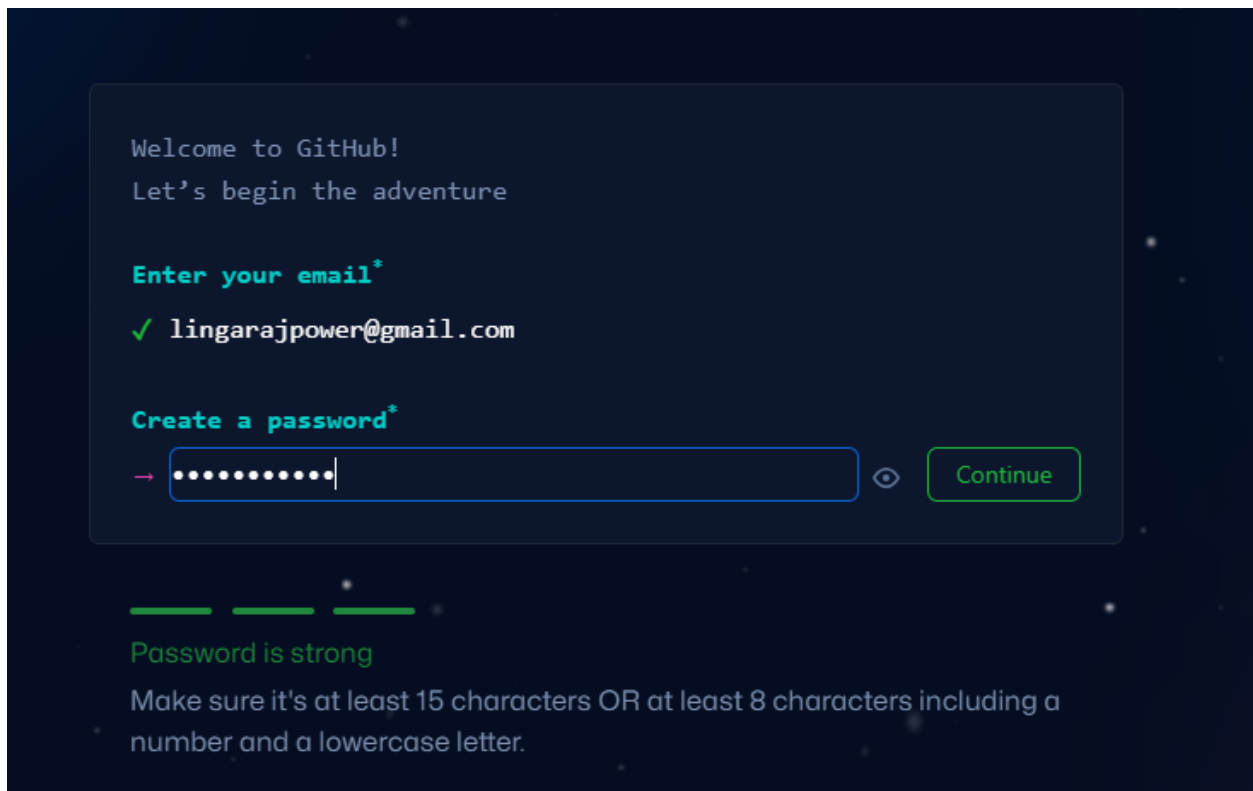Step 1:

Step 2:

Step 3:

```
Welcome to GitHub!
Let's begin the adventure

Enter your email*
✓ lingarajpower@gmail.com

Create a password*
✓ ••••••••••••

Enter a username*
✓ lingarajpower

Email preferences
☐ Receive occasional product updates and       [ Continue ]
  announcements.
```

Step 4:

Verification in mail and OTP
Step 5:

This will help us guide you to the tools that are best suited for your projects.

How many team members will be working with you?

○ Just me   ○ 2-5   ○ 5-10

○ 10-20   ○ 20-50   ○ 50+

Are you a student or teacher?

○ N/A   ● Student   ○ Teacher
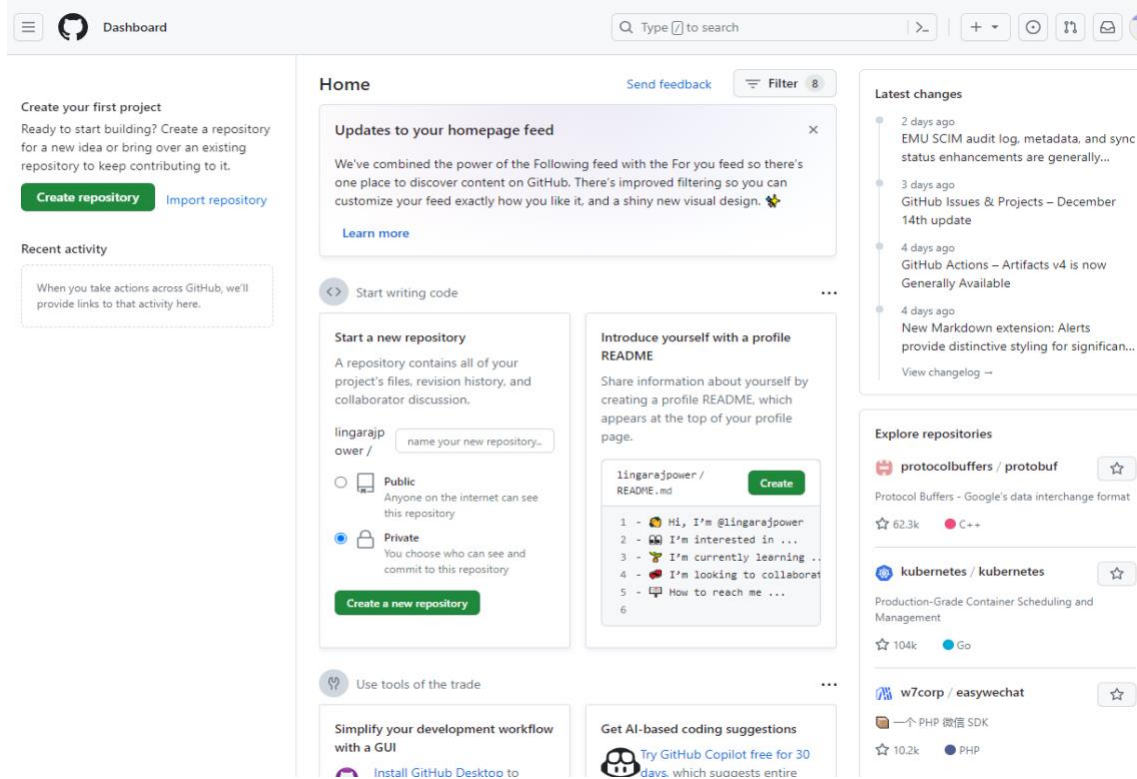
Continue

Step 7:

Select Collaborative coding and community

Step 8: Select free plan.

Step 9:

Step 10: Click your Profile ⭢ overview⭢ Repositories.

Step 11: Create repositories

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? Import a repository.

Required fields are marked with an asterisk (*).

Owner *
lingarajpower ▾

/

Repository name *
3VC15CS065

✔ 3VC15CS065 is available.

Great repository names are short and memorable. Need inspiration? How about shiny-octo-guide ?

Description (optional)
This MY First Reoositry

● 🖥 **Public**
    Anyone on the internet can see this repository. You choose who can commit.

○ 🔒 **Private**
    You choose who can see and commit to this repository.

Initialize this repository with:
☑ **Add a README file**
    This is where you can write a long description for your project. Learn more about READMEs.

Add .gitignore
.gitignore template: None ▾

Choose which files not to track from a list of templates. Learn more about ignoring files.

Choose a license
License: None ▾

A license tells others what they can and can't do with your code. Learn more about licenses.

This will set 🔀 main as the default branch. Change the default name in your settings.

ⓘ You are creating a public repository in your personal account.

🔹 **3VC15CS065** Public   📌 Pin   👁 Unwatch 1 ▾

🔀 main ▾   🔀 1 Branch   🏷 0 Tags   🔍 Go to file   t   Add file ▾   <> Code ▾

🔹 lingarajpower Initial commit   d15bc68 · now   🕐 1 Commits

📄 README.md   Initial commit   now

📖 **README**   ✏️

3VC15CS065

This MY First Reoositry

Step 12: Click Read MD file

Step 13: to edit Read me file go to EDIT

3VC15CS065 / README.md   in main   Cancel changes   Commit changes...

Edit   Preview   🐙 Code 55% faster with GitHub Copilot   Spaces ⬍   2 ⬍   Soft wrap ⬍

```
1    # 3VC15CS065
2    This MY First Reoositry
3    Author - Lingaraj K
4
```
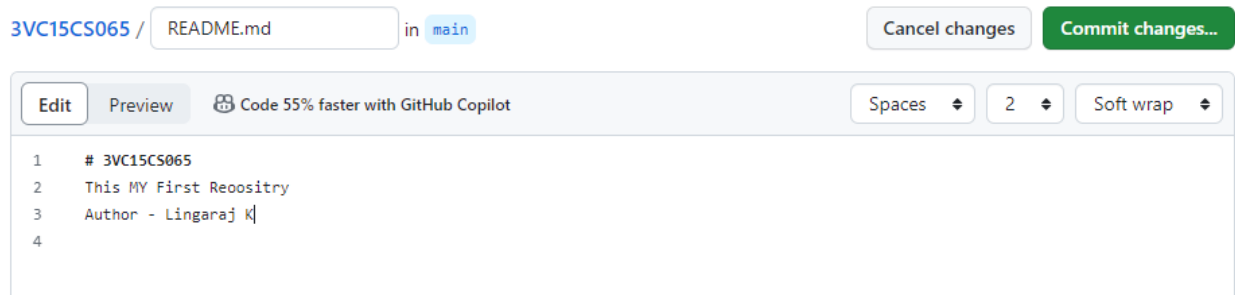
Step 14: Click Commit Changes
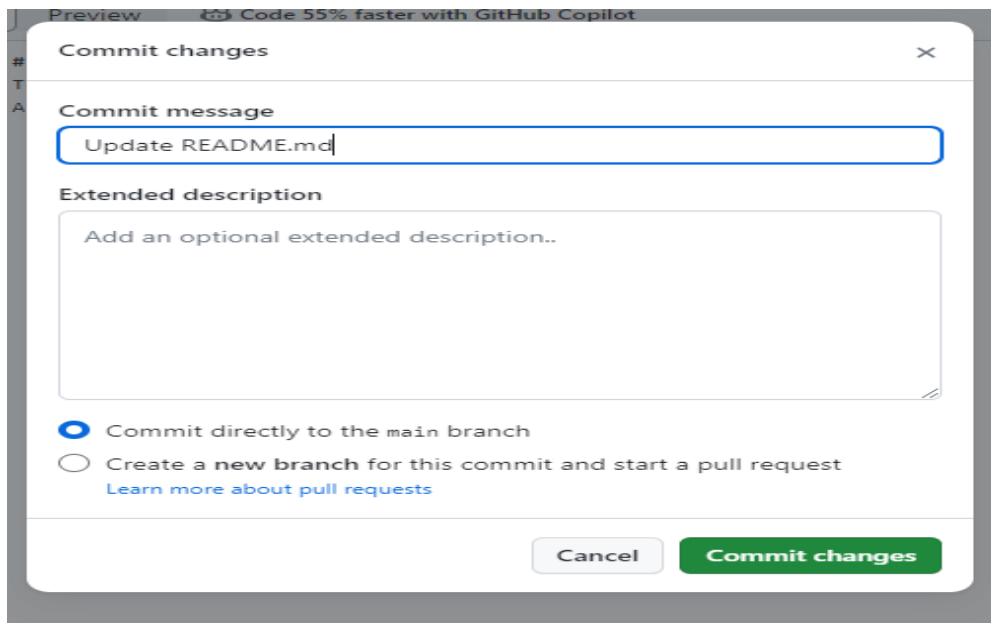
Preview   🐙 Code 55% faster with GitHub Copilot

Commit changes   ✕

Commit message

Update README.md

Extended description

Add an optional extended description..

⬤ Commit directly to the main branch
◯ Create a new branch for this commit and start a pull request
Learn more about pull requests

                              Cancel   Commit changes

Step 15: Cilck code

Pin   Unwatch 1

main   1 Branch   0 Tags   Go to file t   Add file   <> Code

lingarajpower Update README.md   b67b488 · now   2 Commits

README.md   Update README.md   now

README

3VC15CS065

This MY First Reoositry Author - Lingaraj K

To code to better

 in code to get in next line.

Sl. No	Experiments
1	**Setting Up and Basic Commands** Initialize a new Git repository in a directory. Create a new file and add it to the staging area and commit the changes with an appropriate commit message.

Step1 : mkdir 3VC20CS065

Step 2: ls

Step 3: cd 3VC20CS065

Step 4: To create a repository

Git init

Git config-list

```
User@DESKTOP-KKSONE6 MINGW64 ~/3VC20CS065 (master)
$ git config --list
diff.astextplain.textconv=astextplain
filter.lfs.clean=git-lfs clean -- %f
filter.lfs.smudge=git-lfs smudge -- %f
filter.lfs.process=git-lfs filter-process
filter.lfs.required=true
http.sslbackend=openssl
http.sslcainfo=C:/Program Files/Git/mingw64/ssl/certs/ca-bundle.crt
core.autocrlf=true
core.fscache=true
core.symlinks=false
pull.rebase=false
credential.helper=manager
credential.https://dev.azure.com.usehttppath=true
init.defaultbranch=master
core.repositoryformatversion=0
core.filemode=false
core.bare=false
core.logallrefupdates=true
core.symlinks=false
core.ignorecase=true
```

git config –global user.name "lingarajpower"

git config – global user.email " lingarajpower@gmail.com"

```
User@DESKTOP-KKSONE6 MINGW64 ~/3VC20CS065 (master)
$ git config --global user.name "lingarajpower"

User@DESKTOP-KKSONE6 MINGW64 ~/3VC20CS065 (master)
$ git config --global user.email "lingarajpower@gmail.com"

User@DESKTOP-KKSONE6 MINGW64 ~/3VC20CS065 (master)
$ |
```

Git clone https://github.com/lingarajpower/3VC15CS065.git

```
User@DESKTOP-KKSONE6 MINGW64 ~/3VC20CS065 (master)
$ git clone https://github.com/lingarajpower/3VC15CS065.git
Cloning into '3VC15CS065'...
remote: Enumerating objects: 9, done.
remote: Counting objects: 100% (9/9), done.
remote: Compressing objects: 100% (5/5), done.
remote: Total 9 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (9/9), done.
```

cd 3VC15CS065

git status

$ echo "hello git and git Hub" >> lab1.txt

```
User@DESKTOP-KKSONE6 MINGW64 ~/3VC20CS065/3VC15CS065 (main)
$ echo "hello git and git Hub" >> lab1.txt

User@DESKTOP-KKSONE6 MINGW64 ~/3VC20CS065/3VC15CS065 (main)
$ ls
README.md  lab1.txt
```

$ git add lab1.txt

$ git status

```
User@DESKTOP-KKSONE6 MINGW64 ~/3VC20CS065/3VC15CS065 (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   lab1.txt
```

Git commit –m "first msg"

```
User@DESKTOP-KKSONE6 MINGW64 ~/3VC20CS065/3VC15CS065 (main)
$ git commit -m "first msg"
[main a7e4466] first msg
 1 file changed, 1 insertion(+)
 create mode 100644 lab1.txt

User@DESKTOP-KKSONE6 MINGW64 ~/3VC20CS065/3VC15CS065 (main)
$ git push origin main
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 290 bytes | 290.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/lingarajpower/3VC15CS065.git
   40f7c5c..a7e4466  main -> main
```

Git push origin main

```
3VC15CS065  Public                                    Pin    Unwatch 1

main          1 Branch    0 Tags        Q Go to file     t    Add file    <> Code

lingarajpower  first msg                a7e4466 · 8 minutes ago   4 Commits

README.md             Update README.md                    2 hours ago
lab1.txt              first msg                           8 minutes ago

README

3VC15CS065

This MY First Reoositry
Author - Lingaraj K
```

Sl. No	Experiments
2	**Setting Up and Basic Commands** Initialize a new Git repository in a directory. Create a new file and add it to the staging area and commit the changes with an appropriate commit message.

$ vi one.c

```
include<stdio.h>
void Main()
{
        printf("meging");
}<F2>
```

```
one.c[+] [unix] (14:00 18/12/2023)                                            5,5 All
:wq!
```

$ git add one.c
Git commit –m "create one.c"

```
User@DESKTOP-KKS0NE6 MINGW64 ~/3VC20CS065/3VC15CS065 (main)
$ git add one.c
warning: in the working copy of 'one.c', LF will be replaced by CRLF the next time Git touches it

User@DESKTOP-KKS0NE6 MINGW64 ~/3VC20CS065/3VC15CS065 (main)
$ git commit -m "create one.c"
[main b616e55] create one.c
 1 file changed, 5 insertions(+)
 create mode 100644 one.c
```

git add two.c

git commit -m "create two.c"

```
User@DESKTOP-KKSONE6 MINGW64 ~/3VC20CS065/3VC15CS065 (main)
$ vi two.c

User@DESKTOP-KKSONE6 MINGW64 ~/3VC20CS065/3VC15CS065 (main)
$ ls
README.md  lab1.txt  one.c  two.c

User@DESKTOP-KKSONE6 MINGW64 ~/3VC20CS065/3VC15CS065 (main)
$ git add two.c
warning: in the working copy of 'two.c', LF will be replaced by CRLF the next time Git touches it

User@DESKTOP-KKSONE6 MINGW64 ~/3VC20CS065/3VC15CS065 (main)
$ git commit -m "create two.c"
[main a280c93] create two.c
 1 file changed, 5 insertions(+)
 create mode 100644 two.c
```

git log

git status

```
$ git log
commit a280c93d3bf365599568dcba134a4ccaae58e842 (HEAD -> main)
Author: lingarajpower <lingarajpower@gmail.com>
Date:   Mon Dec 18 14:11:21 2023 +0530

    create two.c

commit b616e55177927154575a4a837d0c105799c9cdbb
Author: lingarajpower <lingarajpower@gmail.com>
Date:   Mon Dec 18 14:09:17 2023 +0530

    create one.c

commit a7e4466b06caffaf024e977c19bf7223c29120e4 (origin/main, origin/HEAD)
Author: lingarajpower <lingarajpower@gmail.com>
Date:   Mon Dec 18 13:48:01 2023 +0530

    first msg

commit 40f7c5cdcceb99dc061e099d1c03e7c78d88328b
Author: lingarajpower <154112805+lingarajpower@users.noreply.github.com>
Date:   Mon Dec 18 11:23:03 2023 +0530

    Update README.md

commit b67b488fcb54d8f7f268f022cf436310f37c0536
Author: lingarajpower <154112805+lingarajpower@users.noreply.github.com>
Date:   Mon Dec 18 11:20:31 2023 +0530

    Update README.md

commit d15bc68e50c3d6032bb111ac7af2e248509a3006
Author: lingarajpower <154112805+lingarajpower@users.noreply.github.com>
Date:   Mon Dec 18 11:10:57 2023 +0530

    Initial commit

User@DESKTOP-KKSONE6 MINGW64 ~/3VC20cs065/3VC15CS065 (main)
$ git status
On branch main
Your branch is ahead of 'origin/main' by 2 commits.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean
```

$ git branch b1

git branch b2

git branch

```
User@DESKTOP-KKS0NE6 MINGW64 ~/3VC20cs065/3VC15CS065 (main)
$ git branch b1

User@DESKTOP-KKS0NE6 MINGW64 ~/3VC20cs065/3VC15CS065 (main)
$ git branch b2

User@DESKTOP-KKS0NE6 MINGW64 ~/3VC20cs065/3VC15CS065 (main)
$ git branch
  b1
  b2
* main
```

Git checkout b1

Git status

```
User@DESKTOP-KKS0NE6 MINGW64 ~/3VC20cs065/3VC15CS065 (main)
$ git checkout b1
Switched to branch 'b1'

User@DESKTOP-KKS0NE6 MINGW64 ~/3VC20cs065/3VC15CS065 (b1)
$ git status
On branch b1
nothing to commit, working tree clean
```

$ echo " in branch b1 hello world " >> lab3.c

$ echo " in branch b1 branching world " >> lab4.c

git add lab3.c

git add lab4.c

git commit -m " lab3 for b1"

```
User@DESKTOP-KKSONE6 MINGW64 ~/3VC20cs065/3VC15CS065 (b1)
$ echo " in branch b1 hello world " >> lab3.c

User@DESKTOP-KKSONE6 MINGW64 ~/3VC20cs065/3VC15CS065 (b1)
$ echo " in branch b1 branching world " >> lab4.c

User@DESKTOP-KKSONE6 MINGW64 ~/3VC20cs065/3VC15CS065 (b1)
$ ^C

User@DESKTOP-KKSONE6 MINGW64 ~/3VC20cs065/3VC15CS065 (b1)
$ git add lab3.c
warning: in the working copy of 'lab3.c', LF will be replaced by CRLF the next time Git touches it

User@DESKTOP-KKSONE6 MINGW64 ~/3VC20cs065/3VC15CS065 (b1)
$ git add lab4.c
warning: in the working copy of 'lab4.c', LF will be replaced by CRLF the next time Git touches it

User@DESKTOP-KKSONE6 MINGW64 ~/3VC20cs065/3VC15CS065 (b1)
$ git commit -m " lab3 for b1"
[b1 d0f9cc3]  lab3 for b1
 2 files changed, 2 insertions(+)
 create mode 100644 lab3.c
 create mode 100644 lab4.c
```

git checkout b2

git branch

```
User@DESKTOP-KKSONE6 MINGW64 ~/3VC20cs065/3VC15CS065 (b1)
$ git checkout b2
Switched to branch 'b2'

User@DESKTOP-KKSONE6 MINGW64 ~/3VC20cs065/3VC15CS065 (b2)
$ git branch
  b1
* b2
  main
```

echo " in branch b2 hello world" >> lab5.c

echo " in branch b2 branching world" >> lab6.c

```
User@DESKTOP-KKSONE6 MINGW64 ~/3VC20cs065/3VC15CS065 (b2)
$ echo " in branch b2 hello world" >> lab5.c

User@DESKTOP-KKSONE6 MINGW64 ~/3VC20cs065/3VC15CS065 (b2)
$ echo " in branch b2 branching world" >> lab6.c
```

git add lab5.c

git add lab6.c

git commit -m " lab5 and lab 6 in branch b2"

```
User@DESKTOP-KKSONE6 MINGW64 ~/3VC20cs065/3VC15CS065 (b2)
$ echo " in branch b2 hello world" >> lab5.c

User@DESKTOP-KKSONE6 MINGW64 ~/3VC20cs065/3VC15CS065 (b2)
$ echo " in branch b2 branching world" >> lab6.c

User@DESKTOP-KKSONE6 MINGW64 ~/3VC20cs065/3VC15CS065 (b2)
$ git add lab5.c
warning: in the working copy of 'lab5.c', LF will be replaced by CRLF the next time Git touches it

User@DESKTOP-KKSONE6 MINGW64 ~/3VC20cs065/3VC15CS065 (b2)
$ git add lab6.c
warning: in the working copy of 'lab6.c', LF will be replaced by CRLF the next time Git touches it

User@DESKTOP-KKSONE6 MINGW64 ~/3VC20cs065/3VC15CS065 (b2)
$ git commit -m " lab5 and lab 6 in branch b2"
[b2 004b149]  lab5 and lab 6 in branch b2
 2 files changed, 2 insertions(+)
 create mode 100644 lab5.c
 create mode 100644 lab6.c
```

git checkout master

git diff main..b1

```
User@DESKTOP-KKSONE6 MINGW64 ~/3VC20cs065/3VC15CS065 (b2)
$ git checkout main
Switched to branch 'main'
Your branch is ahead of 'origin/main' by 2 commits.
  (use "git push" to publish your local commits)

User@DESKTOP-KKSONE6 MINGW64 ~/3VC20cs065/3VC15CS065 (main)
$ git diff main..b1
diff --git a/lab3.c b/lab3.c
new file mode 100644
index 0000000..2e8d8d2
--- /dev/null
+++ b/lab3.c
@@ -0,0 +1 @@
+ in branch b1 hello world
diff --git a/lab4.c b/lab4.c
new file mode 100644
index 0000000..8d73116
--- /dev/null
+++ b/lab4.c
@@ -0,0 +1 @@
+ in branch b1 branching world
```

git merge b1

```
User@DESKTOP-KKSONE6 MINGW64 ~/3VC20cs065/3VC15CS065 (main)
$ git merge b1
Updating a280c93..d0f9cc3
Fast-forward
 lab3.c | 1 +
 lab4.c | 1 +
 2 files changed, 2 insertions(+)
 create mode 100644 lab3.c
 create mode 100644 lab4.c
```

git branch –merged

```
User@DESKTOP-KKS0NE6 MINGW64 ~/3VC20cs065/3VC15CS065 (main)
$ git branch --merged
  b1
* main
```

git branch -d b1

git branch -d b2

$ git status

```
User@DESKTOP-KKS0NE6 MINGW64 ~/3VC20cs065/3VC15CS065 (main)
$ git branch --merged
  b1
* main

User@DESKTOP-KKS0NE6 MINGW64 ~/3VC20cs065/3VC15CS065 (main)
$ git branch -d b1
Deleted branch b1 (was d0f9cc3).

User@DESKTOP-KKS0NE6 MINGW64 ~/3VC20cs065/3VC15CS065 (main)
$ git branch -d b2
error: the branch 'b2' is not fully merged.
If you are sure you want to delete it, run 'git branch -D b2'

User@DESKTOP-KKS0NE6 MINGW64 ~/3VC20cs065/3VC15CS065 (main)
$ git status
On branch main
Your branch is ahead of 'origin/main' by 3 commits.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean
```

git merge b2

```
$ git merge b2
Merge made by the 'ort' strategy.
 lab5.c | 1 +
 lab6.c | 1 +
 2 files changed, 2 insertions(+)
 create mode 100644 lab5.c
 create mode 100644 lab6.c
```

git branch –-merged

git branch -d b2

```
User@DESKTOP-KKSONE6 MINGW64 ~/3VC20cs065/3VC15CS065 (main)
$ git branch -- merged

User@DESKTOP-KKSONE6 MINGW64 ~/3VC20cs065/3VC15CS065 (main)
$ git branch --merged
  b2
* main
  merged

User@DESKTOP-KKSONE6 MINGW64 ~/3VC20cs065/3VC15CS065 (main)
$ git branch -d b2
Deleted branch b2 (was 004b149).
```

Sl. No	Experiments
3	Creating and Managing Branches Write the commands to stash your changes, switch branches, and then apply the stashed changes

vi index.txt

git add .

$ git commit -m "create indes file"


```
RYMEC@RYMEC-PC MINGW32 ~/3VC15IS067/3VC15CS065 (main)
$ vi index.txt

RYMEC@RYMEC-PC MINGW32 ~/3VC15IS067/3VC15CS065 (main)
$ git add .

RYMEC@RYMEC-PC MINGW32 ~/3VC15IS067/3VC15CS065 (main)
$ git commit -m "create indes file"
[main 8e08a12] create indes file
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 index.txt
```

$ git branch feature

$ git checkout feature

```
RYMEC@RYMEC-PC MINGW32 ~/3VC15IS067/3VC15CS065 (main)
$ git branch feature

RYMEC@RYMEC-PC MINGW32 ~/3VC15IS067/3VC15CS065 (main)
$ git checkout feature
Switched to branch 'feature'
```

vi feature.txt

git add.

$ git commit -m "work in progress"

```
RYMEC@RYMEC-PC MINGW32 ~/3VC15IS067/3VC15CS065 (feature)
$ git add .
warning: in the working copy of 'feature.txt', LF will be replaced by CRLF the next time Git touches it

RYMEC@RYMEC-PC MINGW32 ~/3VC15IS067/3VC15CS065 (feature)
$ git commit -m "work in progress"
[feature 0bc5c24] work in progress
 1 file changed, 1 insertion(+)
 create mode 100644 feature.txt
```

$ vi index.txt

$ git status

git checkout main

```
RYMEC@RYMEC-PC MINGW32 ~/3VC15IS067/3VC15CS065 (feature)
$ git checkout main
error: Your local changes to the following files would be overwritten by checkou
t:
        index.txt
Please commit your changes or stash them before you switch branches.
Aborting
```

git stash

```
RYMEC@RYMEC-PC MINGW32 ~/3VC15IS067/3VC15CS065 (feature)
$ git stash
warning: in the working copy of 'index.txt', LF will be replaced by CRLF the next time Git touches it
Saved working directory and index state WIP on feature: 0bc5c24 work in progress
```

Git status

```
RYMEC@RYMEC-PC MINGW32 ~/3VC15IS067/3VC15CS065 (feature)
$ git status
On branch feature
nothing to commit, working tree clean
```

git stash list

```
RYMEC@RYMEC-PC MINGW32 ~/3VC15IS067/3VC15CS065 (feature)
$ git stash list
stash@{0}: WIP on feature: 0bc5c24 work in progress
```

git checkout main

```
RYMEC@RYMEC-PC MINGW32 ~/3VC15IS067/3VC15CS065 (feature)
$ git checkout main
Switched to branch 'main'
Your branch is ahead of 'origin/main' by 2 commits.
  (use "git push" to publish your local commits)
```

git checkout feature

```
RYMEC@RYMEC-PC MINGW32 ~/3VC15IS067/3VC15CS065 (main)
$ git checkout feature
Switched to branch 'feature'
```

git stash pop

```
RYMEC@RYMEC-PC MINGW32 ~/3VC15IS067/3VC15CS065 (feature)
$ git stash pop
On branch feature
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   index.txt

no changes added to commit (use "git add" and/or "git commit -a")
Dropped refs/stash@{0} (b619c298941172adf9563a9ba16a28dccf8bb706)
```

$ git stash list

$ git add .

git commit -m "create indes file changed in feature"

```
RYMEC@RYMEC-PC MINGW32 ~/3VC15IS067/3VC15CS065 (feature)
$ git commit -m "create indes file changed in feature"
[feature 9c13bc4] create indes file changed in feature
 1 file changed, 1 insertion(+)
```

Sl. No	Experiments
4	Collaboration and Remote Repositories Clone a remote Git repository to your local machine

lingarajpower

Edit profile

Popular repositories

3VC15CS065   Public
This MY First Reoositry

Customize your pins

6 contributions in the last year   Contribution settings ▾   2024

2023

Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec Jan
Mon
Wed
Fri
Learn how we count contributions   Less ☐☐☐☐☐ More

Contribution activity

January 2024

lingarajpower has no activity yet for this period.

Show more activity

Seeing something unexpected? Take a look at the GitHub profile guide.

Step 2:
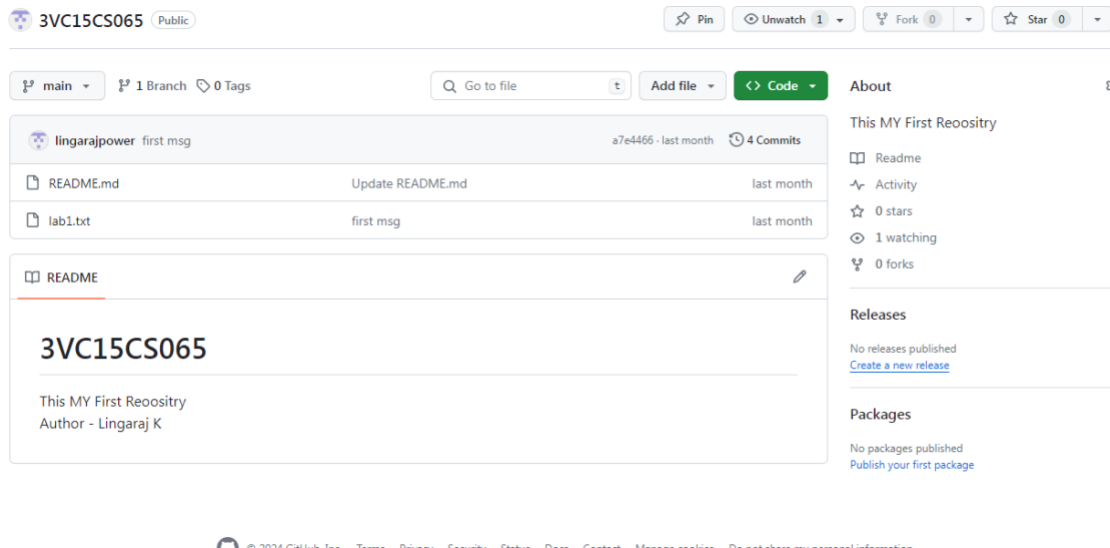
Step 3:

Add file → create new file
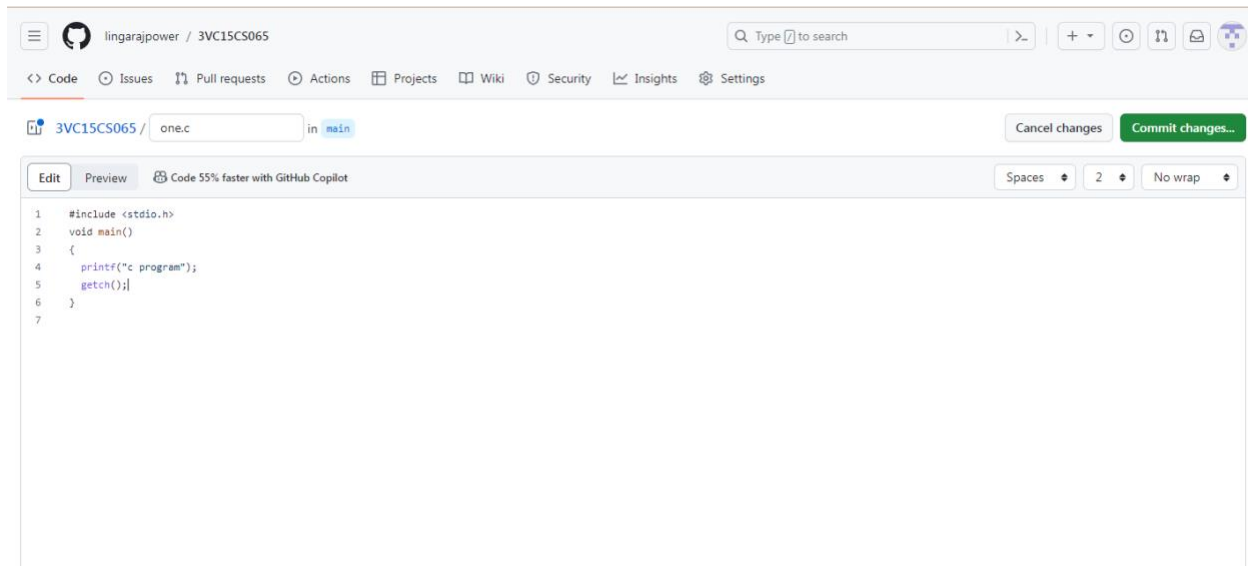
Step 4: commit changes →commit changes

Name	Last commit message	Last commit date
README.md	Update README.md	last month
lab1.txt	first msg	last month
one.c	Create one.c	now

README.md

3VC15CS065

This MY First Reoositry
Author - Lingaraj K

Step 5: go to git bash

Step 6:

Git clone https://github.com/lingarajpower/3VC15CS065.git

Sl. No	Experiments
5	Collaboration and Remote Repositories Fetch the latest changes from a remote repository and rebase your local branch onto the updated remote branch.

Step 1:

vi index.txt

git add .

$ git commit -m "create indes file"

```
RYMEC@RYMEC-PC MINGW32 ~/3VC15IS067/3VC15CS065 (main)
$ vi index.txt

RYMEC@RYMEC-PC MINGW32 ~/3VC15IS067/3VC15CS065 (main)
$ git add .

RYMEC@RYMEC-PC MINGW32 ~/3VC15IS067/3VC15CS065 (main)
$ git commit -m "create indes file"
[main 8e08a12] create indes file
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 index.txt
```

$ git pull https://github.com/lingarajpower/3VC15CS065

```
$ git pull https://github.com/lingarajpower/3VC15CS065
remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 1017 bytes | 5.00 KiB/s, done.
From https://github.com/lingarajpower/3VC15CS065
 * branch            HEAD        -> FETCH_HEAD
Merge made by the 'ort' strategy.
 one.c | 6 ++++++
 1 file changed, 6 insertions(+)
 create mode 100644 one.c
```

ls

```
RYMEC@RYMEC-PC MINGW32 ~/3VC15IS067/3VC15CS065 (main)
$ ls
3VC15CS065/  README.md  index.txt  lab1.txt  one.c
```

Step 2: settings→ developers setting →personal access tokens→personal acess tokens(classic)→genetrate new token

Step 3: name →mytoken

Expiration → customize to 6 months

Select all box and submit

Copy token in

Step 4: go to git bash paste

git remote set-url origin
https://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@github.com/lingarajpower/3VC15CS065

git push

```
RYMEC@RYMEC-PC MINGW32 ~/3VC15IS067/3VC15CS065 (main)
$ git  remote set-url origin https://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@github.com/lingarajpower/3VC15CS065

RYMEC@RYMEC-PC MINGW32 ~/3VC15IS067/3VC15CS065 (main)
$ git push
Enumerating objects: 10, done.
Counting objects: 100% (10/10), done.
Delta compression using up to 4 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (8/8), 840 bytes | 210.00 KiB/s, done.
Total 8 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), done.
To https://github.com/lingarajpower/3VC15CS065
   2e284b2..06861d0  main -> main
```

Sl. No	Experiments
6	Collaboration and Remote Repositories Write the command to merge "feature-branch" into "master" while providing a custom commit message for the merge.

Vi pgm6.c

```
#include<stdio.h>
void main()
{
        printf("inside master branch");
}
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
pgm6.c[+] [unix] (05:29 01/01/1970)
:wq
```

Git add pgm6.c

Git commit –m "program 6"

Git branch feature-branch

Git checkout feature-branch

```
RYMEC@RYMEC-PC MINGW32 ~/3VC15IS067 (master)
$ git add pgm6.c
warning: in the working copy of 'pgm6.c', LF will be replaced by CRLF the next time Git touches it

RYMEC@RYMEC-PC MINGW32 ~/3VC15IS067 (master)
$ git commit -m "pgm6.c"
[master (root-commit) 5f31aff] pgm6.c
 1 file changed, 6 insertions(+)
 create mode 100644 pgm6.c

RYMEC@RYMEC-PC MINGW32 ~/3VC15IS067 (master)
$ git branch feature-branch
```

Vi pgm6b.c

```
#include<stdio.h>
void main()
{
        printf(" In feature-Branch");
}

~
~
~
~
~
~
~
~
~
```

Git add pgm6b.c

Git commit –m "pgm 6b feature branch"

```
$ git add pgm6b.c
warning: in the working copy of 'pgm6b.c', LF will be replaced by CRLF the next time Git touches it

RYMEC@RYMEC-PC MINGW32 ~/3VC15IS067 (feature-branch)
$ git commit -m "pgm 6b"
[feature-branch 1d9cf5c] pgm 6b
 1 file changed, 6 insertions(+)
 create mode 100644 pgm6b.c
```

git checkout master

git merge --no-ff feature-branch -m "Your custom merge commit message"

```
RYMEC@RYMEC-PC MINGW32 ~/3VC15IS067 (feature-branch)
$ git checkout master
Switched to branch 'master'

RYMEC@RYMEC-PC MINGW32 ~/3VC15IS067 (master)
$ git merge --no-ff feature-branch -m "Your custom merge commit message"
Merge made by the 'ort' strategy.
 pgm6b.c | 6 ++++++
 1 file changed, 6 insertions(+)
 create mode 100644 pgm6b.c
```

Note: Here's a breakdown of the command:

➤ **git checkout master:** This switches the current working directory to the master branch.

➤ **git merge --no-ff feature-branch:** This merges feature-branch into the current branch (master). The --no-ff flag prevents git from performing a fast-forward merge. Fast-forward merges can happen when the merging branch (in this case, master) has no commits that the merged branch (feature-branch) does not have. By using --no-ff, it ensures that a new commit is created for the merge, preserving the history of the feature branch as a distinct branch.

➤ **-m "Your custom merge commit message":** Allows you to provide a custom message for the merge commit. Replace Your custom merge commit message with the actual message you want to use.

➤ Remember to replace "Your custom merge commit message" with the actual message you wish to use for documenting the merge. This approach ensures the merge is clearly documented and identifiable in the project's history.

Sl. No	Experiments
7	Write the command to create a lightweight Git tag named "v1.0" for a commit in your local repository.

git checkout main

git tag v1.0

git tag

```
RYMEC@RYMEC-PC MINGW32 ~/3VC15IS067 (master)
$ git tag v1.0

RYMEC@RYMEC-PC MINGW32 ~/3VC15IS067 (master)
$ git tag
v1.0
```

git tag -a v1.1 -m "tag for release ver 1.1"

git tag

git show v1.0

```
RYMEC@RYMEC-PC MINGW32 ~/3VC15IS067 (master)
$ git tag -a v1.1 -m "tag for release ver 1.1"

RYMEC@RYMEC-PC MINGW32 ~/3VC15IS067 (master)
$ git tag
v1.0
v1.1

RYMEC@RYMEC-PC MINGW32 ~/3VC15IS067 (master)
$ git show v1.0
commit 74eec160e7054ccb8f7ff67a4e12442130cc3971 (HEAD -> master, tag: v1.1, tag: v1.0)
Merge: 5f31aff 1d9cf5c
Author: lingarajpower <lingarajpower@gmail.com>
Date:   Mon Feb 5 11:17:50 2024 +0530

    Your custom merge commit message
```

Git push origin v1.0

Sl. No	Experiments
11	Write the command to display the last five commits in the repository's history.

$ echo "I am in 11a" >> lab11a.txt

$ echo " I am in 11b " >> lab11b.txt

$ echo " I am in 11c " >> lab11c.txt

$ echo " I am in 11d " >> lab11d.txt

$ echo " I am in 11e " >> lab11e.txt

git add .

$ git commit -m "create indes file"

$ git log –n 5

Sl. No	Experiments
10	Write the command to list all commits made by the author "user" between "2024-01-01" and "2024-02-07."

$ echo "I am in 10a" >> lab10a.txt

git add .

$ git commit -m " I am in 10a "

$ echo "I am in 10b" >> lab10b.txt

git add .

$ git commit -m " I am in 10b "

$ echo "I am in 10c" >> lab10c.txt

git add .

$ git commit -m " I am in 10c "

$ git log --author="lingarajpower" --since="2024-01-01" --until="2024-02-07"

```
RYMEC@RYMEC-PC MINGW32 ~/3VC15IS067 (master)
$ git log --author="lingarajpower" --since="2024-01-01" --until="2024-02-07"
commit dafe0944f420d5ea9d1d9c6f11988bdc35b8ed13 (HEAD -> master)
Author: lingarajpower <lingarajpower@gmail.com>
Date:   Mon Feb 5 13:58:03 2024 +0530

    i am in 11a

commit 74eec160e7054ccb8f7ff67a4e12442130cc3971 (tag: v1.1, tag: v1.0)
Merge: 5f31aff 1d9cf5c
Author: lingarajpower <lingarajpower@gmail.com>
Date:   Mon Feb 5 11:17:50 2024 +0530

    Your custom merge commit message

commit 1d9cf5c027b09a43242cbe20a4196d928b8acb10 (feature-branch)
Author: lingarajpower <lingarajpower@gmail.com>
Date:   Mon Feb 5 11:13:06 2024 +0530

    pgm 6b

commit 5f31aff812d5ff0c6b01e316c183e542215dd8e6
Author: lingarajpower <lingarajpower@gmail.com>
Date:   Mon Feb 5 11:06:30 2024 +0530

:
commit dafe0944f420d5ea9d1d9c6f11988bdc35b8ed13 (HEAD -> master)
Author: lingarajpower <lingarajpower@gmail.com>
Date:   Mon Feb 5 13:58:03 2024 +0530

    i am in 11a

commit 74eec160e7054ccb8f7ff67a4e12442130cc3971 (tag: v1.1, tag: v1.0)
Merge: 5f31aff 1d9cf5c
Author: lingarajpower <lingarajpower@gmail.com>
Date:   Mon Feb 5 11:17:50 2024 +0530
```

Sl. No	Experiments
9	Write the command to cherry-pick a range of commits from "source-branch" to the current branch.

$ git checkout master

$ echo "I am in 9a" >> lab9a.txt
git add .
$ git commit -m " I am in 9a branch master "
$ git branch cherry
$ git log

```
RYMEC@RYMEC-PC MINGW32 ~/3VC15IS067 (master)
$ git log
commit a0ea8dcdcc2e1942ec7f65c2c4c945c8686a1e71 (HEAD -> master, cheery)
Author: lingarajpower <lingarajpower@gmail.com>
Date:   Mon Feb 19 09:37:34 2024 +0530

    im in 9a master
```

$ git checkout cheery

$ echo "I am in 9b" >> lab9b.txt
git add .
$ git commit -m " I am in 9b branch cheery "
$ echo "I am in 9c" >> lab9c.txt
git add .
$ git commit -m " I am in 9c branch cheery "

```
RYMEC@RYMEC-PC MINGW32 ~/3VC15IS067 (cheery)
$ echo "I am in 9b" >> lab9b.txt

RYMEC@RYMEC-PC MINGW32 ~/3VC15IS067 (cheery)
$ git add .
warning: in the working copy of 'lab9b.txt', LF will be replaced by CRLF the next time Git touches it

RYMEC@RYMEC-PC MINGW32 ~/3VC15IS067 (cheery)
$ git commit -m " I am in 9b branch cheery "
[cheery 68ce021]  I am in 9b branch cheery
 1 file changed, 1 insertion(+)
 create mode 100644 lab9b.txt

RYMEC@RYMEC-PC MINGW32 ~/3VC15IS067 (cheery)
$ echo "I am in 9c" >> lab9c.txt

RYMEC@RYMEC-PC MINGW32 ~/3VC15IS067 (cheery)
$ git add .
warning: in the working copy of 'lab9c.txt', LF will be replaced by CRLF the next time Git touches it

RYMEC@RYMEC-PC MINGW32 ~/3VC15IS067 (cheery)
$ git commit -m " I am in 9c branch cheery "
[cheery f7b4649]  I am in 9c branch cheery
 1 file changed, 1 insertion(+)
 create mode 100644 lab9c.txt

RYMEC@RYMEC-PC MINGW32 ~/3VC15IS067 (cheery)
$ git log
commit f7b46493e18b1dd58703830637ae9d3e615adbde (HEAD -> cheery)
Author: lingarajpower <lingarajpower@gmail.com>
Date:   Mon Feb 19 09:44:10 2024 +0530

    I am in 9c branch cheery

commit 68ce02170f07ee2d50b99d885f3d48886fd4db9c
Author: lingarajpower <lingarajpower@gmail.com>
Date:   Mon Feb 19 09:43:12 2024 +0530

    I am in 9b branch cheery
```

$ git checkout master

$ git log

$ git cherry-pick f7b46493e18b1dd58703830637ae9d3e615adbde

$ git log

```
RYMEC@RYMEC-PC MINGW32 ~/3VC15IS067 (master)
$ git cherry-pick f7b46493e18b1dd58703830637ae9d3e615adbde
[master 95997fc]  I am in 9c branch cheery
 Date: Mon Feb 19 09:44:10 2024 +0530
 1 file changed, 1 insertion(+)
 create mode 100644 lab9c.txt

RYMEC@RYMEC-PC MINGW32 ~/3VC15IS067 (master)
$ git log
commit 95997fc7e3d70732cb5d3d78e917777b2c29572e (HEAD -> master)
Author: lingarajpower <lingarajpower@gmail.com>
Date:   Mon Feb 19 09:44:10 2024 +0530

    I am in 9c branch cheery
```

Sl. No	Experiments
8	Given a commit ID, how would you use Git to view the details of that specific commit, including the author, date, and commit message?

$ git checkout master

$ echo "I am in 8a" >> lab8a.txt
git add .
$ git commit -m " I am in 8a branch master "

$ git log

$ git show f7b46493e18b1dd58703830637ae9d3e615adbde

```
RYMEC@RYMEC-PC MINGW32 ~/3VC15IS067 (master)
$ git show f7b46493e18b1dd58703830637ae9d3e615adbde
commit f7b46493e18b1dd58703830637ae9d3e615adbde (cheery)
Author: lingarajpower <lingarajpower@gmail.com>
Date:   Mon Feb 19 09:44:10 2024 +0530

    I am in 9c branch cheery

diff --git a/lab9c.txt b/lab9c.txt
new file mode 100644
index 0000000..c9d7539
--- /dev/null
+++ b/lab9c.txt
@@ -0,0 +1 @@
+I am in 9c
```

Sl. No	Experiments
12	Write the command to undo the changes introduced by the commit with the ID "abc123".

$ git checkout master

$ vi 12a.txt

```
Hi i am in 12 a master
~
~
~
~
~
~
```

git add .
$ git commit -m " I am in 12a branch master "
$ vi 12b.txt

```
i am in 12 b master branch
~
~
~
~
```

git add .
$ git commit -m " I am in 12b branch master "

$ vi 12a.txt

```
Hi i am in 12 a master
I haved made necessary changes
```

git add .
$ git commit -m " I am in 12a branch master changes 1"

$ vi 12b.txt

```
i am in 12 b master branch
changes made |
~
~
```

git add .

$ git commit -m " I am in 12b branch master changes 1 "

$ git log --oneline

```
RYMEC@RYMEC-PC MINGW32 ~/3VC15IS067 (master)
$ git log --oneline
ec9d05e (HEAD -> master)  i am in 12a branch master changes 1
f34ebc1  i am in 12a branch master changes 1
11e264b  i am in 12a branch master
5fe8ac7  i am in 12b branch master
6de9cfd  i am in 12a branch master
95997fc  I am in 9c branch cheery
a0ea8dc im in 9a master
dafe094 i am in 11a
74eec16 (tag: v1.1, tag: v1.0) Your custom merge commit message
1d9cf5c (feature-branch) pgm 6b
5f31aff pgm6.c
```

$ git revert ec9d05e

```
RYMEC@RYMEC-PC MINGW32 ~/3VC15IS067 (master|REVERTING)
$ git revert ec9d05e
[master 4cfac75] Revert " i am in 12a branch master changes 1"
 1 file changed, 1 deletion(-)
```