SAHYADRI
COLLEGE OF ENGINEERING & MANAGEMENT
An Autonomous Institution
MANGALURU

GitHub: AI – Powered Developer Platform

Prepared by

Vignesh D Jogi

4SF22CI117

Under the Guidance of

Dr. Duddela Sai Prashanth

Assistant Professor

Department of CSE (AI&ML)

SCEM

Maximum Marks	Marks Secured

....

Signature of the Student **Signature of the Faculty**

Problem Statement:

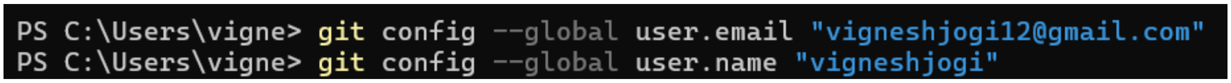
You and your team have embarked on an exciting new project called "GitMaster," designed to create a comprehensive guide and toolset for mastering Git. As a key member, you take on the responsibility of setting up the initial project structure and ensuring proper version control. First, you install Git on your system, configure your global username and email, and create a new directory named my-git-project. Navigating into this directory, you initialize a new Git repository. You create a README.md file with the content: "This is my first repository." Checking the status of your repository, you see the untracked file. You add README.md to the staging area and commit the change with the message: "Add initial README.md file." As the project progresses, you update README.md by adding: "This is an additional line." Viewing the changes, you add the modified file to the staging area and commit with the message: "Update README.md with an additional line." You then review the commit history to see all your changes. To further demonstrate your Git skills, you: Display details of a specific commit using its ID. Show changes between the staging area and the last commit. Rename or move a file .Remove a file from the working tree and the index. For more complex tasks, you create a new directory named advanced-git-project, initialize a Git repository, and create a script.py file with the content: print('Hello, World!'). Checking the status, you add script.py to the staging area and commit with the message: "Add initial script.py file." To test your skills, you modify script.py by adding print('Another line'). Deciding to discard these changes, you restore the file to its last committed state. Then, you modify script.py again by adding print('A new line'). After adding the changes to the staging area, you unstage them to ensure they are no longer staged. Next, you add a comment to script.py: # This is a comment. You stage the changes and amend the last commit to include this change, modifying the commit message to: "Add initial script.py file with a comment." Exploring further, you revert the last commit, view the commit history, and display details of a specific commit using both the full commit ID and the first five characters. Finally, you identify a specific commit to revert and execute the command. As the project gains momentum, your team decides to implement a new feature. You create a new branch named feature-branch to work on this feature without affecting the main codebase. In the feature-branch, you add a new file called feature.txt with some initial content. You commit the new file with the message: "Add feature.txt with initial content." After discussing with your team, you make further changes to feature.txt and commit with the message: "Update feature.txt with additional content." Once the feature is complete, you merge the feature-branch back into the main branch. During the merge, you encounter a conflict, resolve it, commit the changes, and complete the merge. To finalize, you clean up by deleting the feature-branch.

1.Git config command:

The Git config command is used to set the values to identify who made changes to Git repositories. To set the values of user.email and user.name to your email and name, type:

: git config - -global user.email me@example.com

: git config - -global user.name “My name”



```
PS C:\Users\vigne> git config --global user.email "vigneshjogi12@gmail.com"
PS C:\Users\vigne> git config --global user.name "vigneshjogi"
```

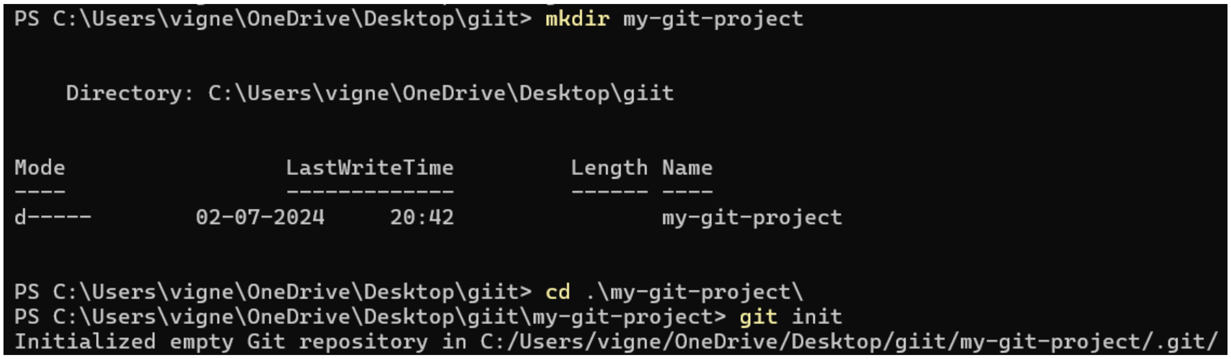
Figure 1

2.Create and Initialize Git Repository:

: mkdir my-git-project : cd my-git-project

: git init

The Git init command can create a new empty repository in a current directory or re-initialize an existing one. The mkdir and cd comand is used to navigate to a suitable directory and create a new directory for your project:

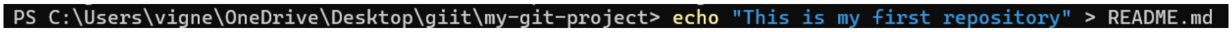


```
PS C:\Users\vigne\OneDrive\Desktop\giit> mkdir my-git-project


    Directory: C:\Users\vigne\OneDrive\Desktop\giit


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----        02-07-2024     20:42                my-git-project


PS C:\Users\vigne\OneDrive\Desktop\giit> cd .\my-git-project\
PS C:\Users\vigne\OneDrive\Desktop\giit\my-git-project> git init
Initialized empty Git repository in C:/Users/vigne/OneDrive/Desktop/giit/my-git-project/.git/
```

Figure 2

3. Create and Commit README.md:

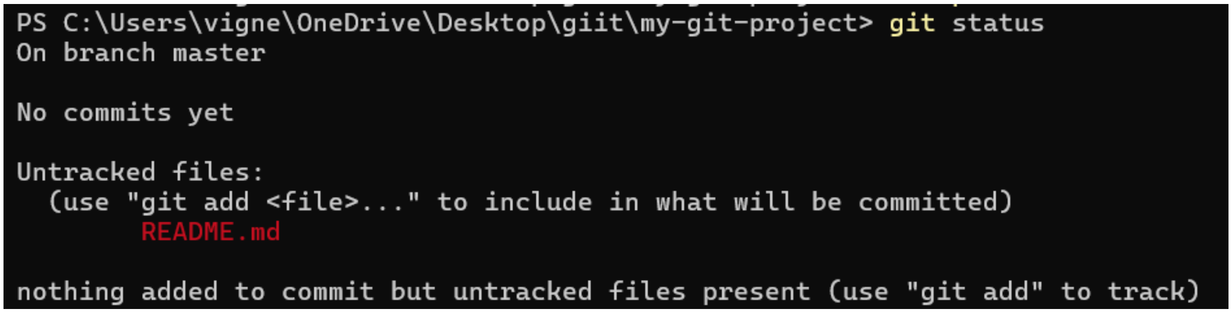
- Create a README.md file with initial content



```
PS C:\Users\vigne\OneDrive\Desktop\giit\my-git-project> echo "This is my first repository" > README.md
```

Figure 3

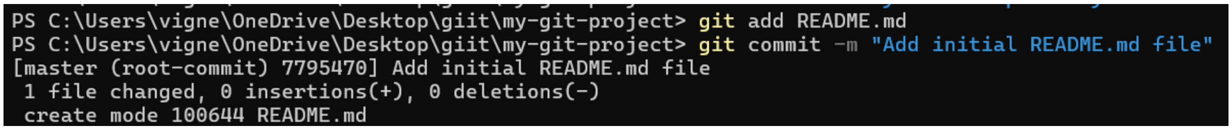
- Check the status of your repository to see untracked files:



```
PS C:\Users\vigne\OneDrive\Desktop\giit\my-git-project> git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        README.md

nothing added to commit but untracked files present (use "git add" to track)
```

Figure 4

- Add README.md to the staging area and commit:

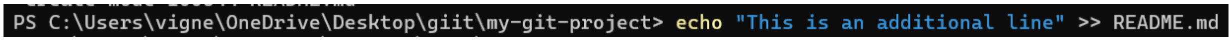


```
PS C:\Users\vigne\OneDrive\Desktop\giit\my-git-project> git add README.md
PS C:\Users\vigne\OneDrive\Desktop\giit\my-git-project> git commit -m "Add initial README.md file"
[master (root-commit) 7795470] Add initial README.md file
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 README.md
```

Figure 5

4.Update and Commit README.md:

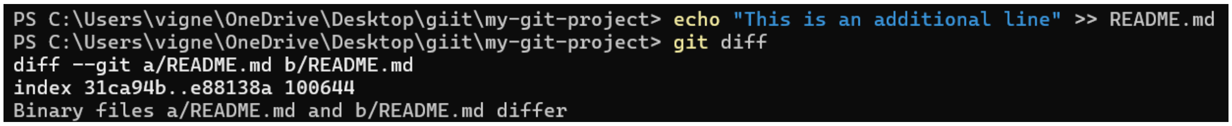
- Update README.md by adding an additional line



```
PS C:\Users\vigne\OneDrive\Desktop\giit\my-git-project> echo "This is an additional line" >> README.md
```

Figure 6

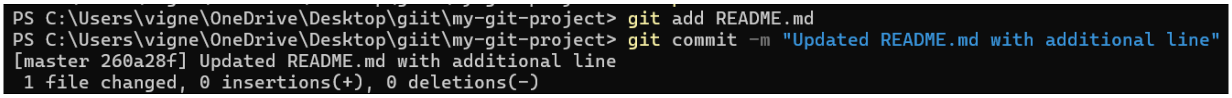
- View the changes made to README.md:



```
PS C:\Users\vigne\OneDrive\Desktop\giit\my-git-project> echo "This is an additional line" >> README.md
PS C:\Users\vigne\OneDrive\Desktop\giit\my-git-project> git diff
diff --git a/README.md b/README.md
index 31ca94b..e88138a 100644
Binary files a/README.md and b/README.md differ
```

Figure 7

- Add the modified file to the staging area and commit:

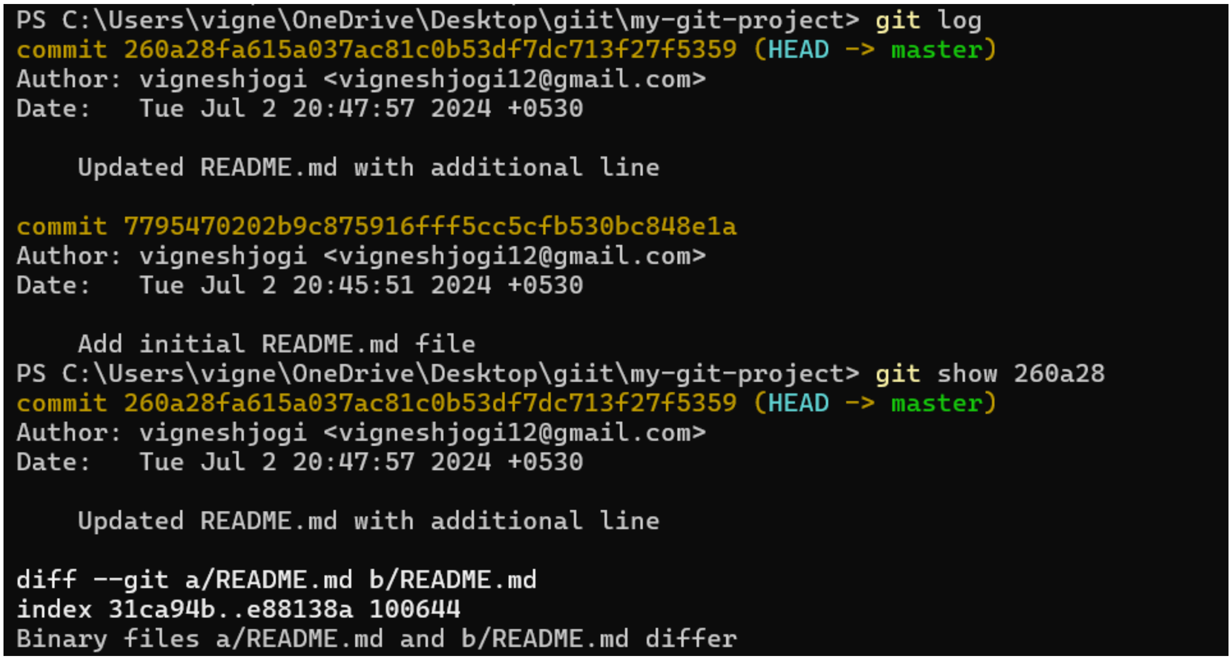


```
PS C:\Users\vigne\OneDrive\Desktop\giit\my-git-project> git add README.md
PS C:\Users\vigne\OneDrive\Desktop\giit\my-git-project> git commit -m "Updated README.md with additional line"
[master 260a28f] Updated README.md with additional line
 1 file changed, 0 insertions(+), 0 deletions(-)
```

Figure 8

5.Demonstrating Git Skills:

- Display Details of a Specific Commit:

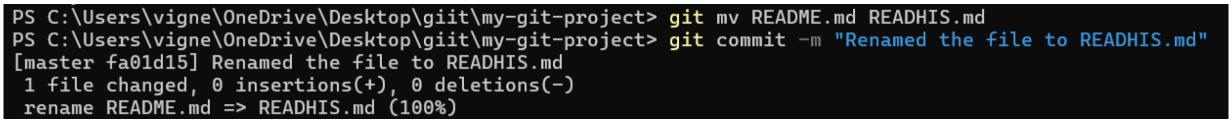


```
PS C:\Users\vigne\OneDrive\Desktop\giit\my-git-project> git log
commit 260a28fa615a037ac81c0b53df7dc713f27f5359 (HEAD -> master)
Author: vigneshjogi <vigneshjogi12@gmail.com>
Date:   Tue Jul 2 20:47:57 2024 +0530

    Updated README.md with additional line

commit 7795470202b9c875916fff5cc5cfb530bc848e1a
Author: vigneshjogi <vigneshjogi12@gmail.com>
Date:   Tue Jul 2 20:45:51 2024 +0530

    Add initial README.md file
PS C:\Users\vigne\OneDrive\Desktop\giit\my-git-project> git show 260a28
commit 260a28fa615a037ac81c0b53df7dc713f27f5359 (HEAD -> master)
Author: vigneshjogi <vigneshjogi12@gmail.com>
Date:   Tue Jul 2 20:47:57 2024 +0530

    Updated README.md with additional line

diff --git a/README.md b/README.md
index 31ca94b..e88138a 100644
Binary files a/README.md and b/README.md differ
```

Figure 9

6.Rename or Move a File:

- Rename or move a file using git mv:



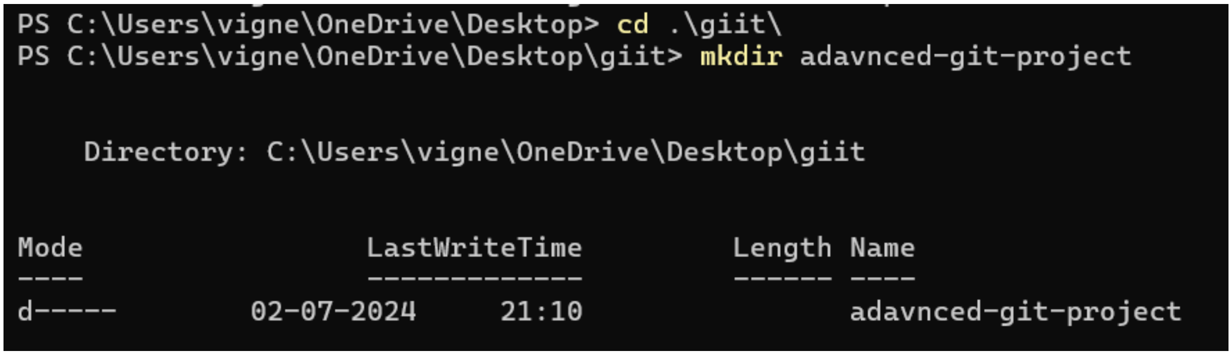
```
PS C:\Users\vigne\OneDrive\Desktop\giit\my-git-project> git mv README.md READHIS.md
PS C:\Users\vigne\OneDrive\Desktop\giit\my-git-project> git commit -m "Renamed the file to READHIS.md"
[master fa01d15] Renamed the file to READHIS.md
 1 file changed, 0 insertions(+), 0 deletions(-)
 rename README.md => READHIS.md (100%)
```

Figure 10

7.Advanced Git Tasks:

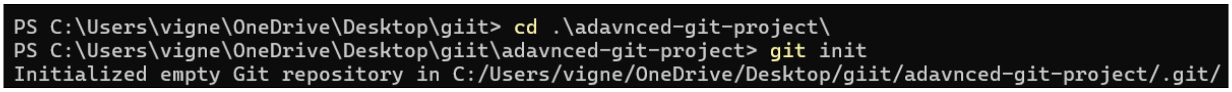
a) Create Advanced Git project:

- Create a new directory and initialize a Git repository



```
PS C:\Users\vigne\OneDrive\Desktop> cd .\giit\
PS C:\Users\vigne\OneDrive\Desktop\giit> mkdir adavnced-git-project


    Directory: C:\Users\vigne\OneDrive\Desktop\giit


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----        02-07-2024     21:10                adavnced-git-project
```

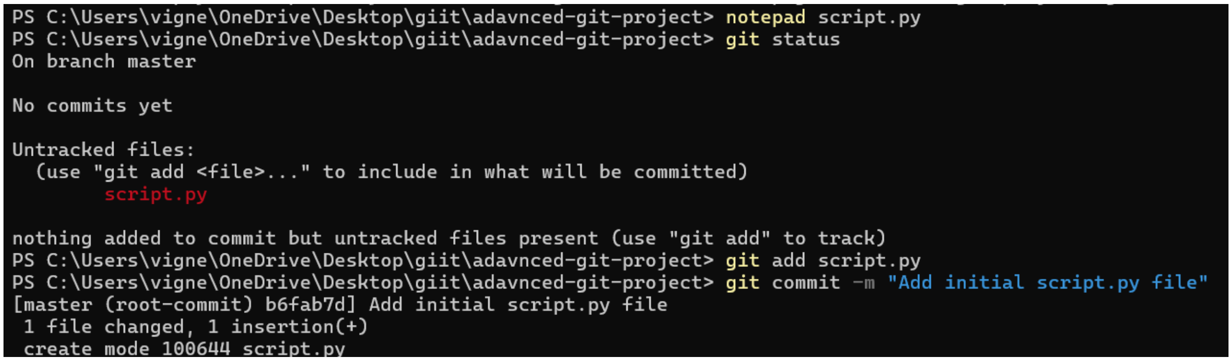
Figure 11



```
PS C:\Users\vigne\OneDrive\Desktop\giit> cd .\adavnced-git-project\
PS C:\Users\vigne\OneDrive\Desktop\giit\adavnced-git-project> git init
Initialized empty Git repository in C:/Users/vigne/OneDrive/Desktop/giit/adavnced-git-project/.git/
```

Figure 12

- Create a script.py file with initial content and commit it:

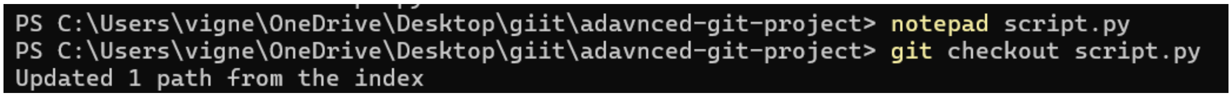


```
PS C:\Users\vigne\OneDrive\Desktop\giit\adavnced-git-project> notepad script.py
PS C:\Users\vigne\OneDrive\Desktop\giit\adavnced-git-project> git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        script.py

nothing added to commit but untracked files present (use "git add" to track)
PS C:\Users\vigne\OneDrive\Desktop\giit\adavnced-git-project> git add script.py
PS C:\Users\vigne\OneDrive\Desktop\giit\adavnced-git-project> git commit -m "Add initial script.py file"
[master (root-commit) b6fab7d] Add initial script.py file
 1 file changed, 1 insertion(+)
 create mode 100644 script.py
```

Figure 13

b) Modify and Discard Changes:

- Modify script.py and then discard changes to restore to the last committed state:

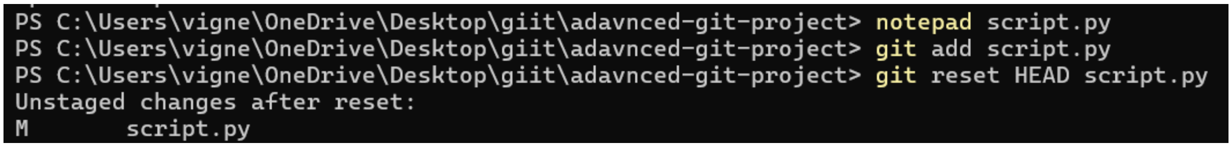


```
PS C:\Users\vigne\OneDrive\Desktop\giit\adavnced-git-project> notepad script.py
PS C:\Users\vigne\OneDrive\Desktop\giit\adavnced-git-project> git checkout script.py
Updated 1 path from the index
```

Figure 14

c) Unstage Changes:

- Modify script.py again and add changes to the staging area, then unstage them:

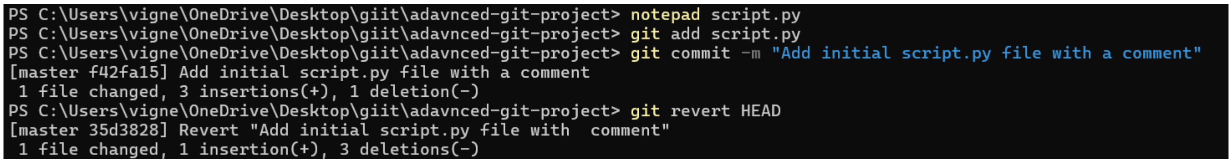


```
PS C:\Users\vigne\OneDrive\Desktop\giit\adavnced-git-project> notepad script.py
PS C:\Users\vigne\OneDrive\Desktop\giit\adavnced-git-project> git add script.py
PS C:\Users\vigne\OneDrive\Desktop\giit\adavnced-git-project> git reset HEAD script.py
Unstaged changes after reset:
M       script.py
```

Figure 15

d) Amend Last Commit:

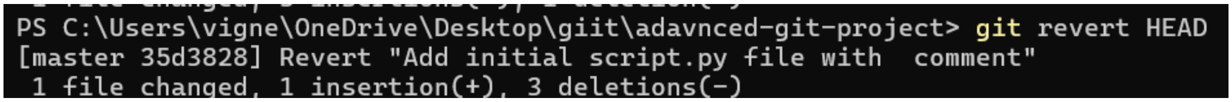
- Add a comment to script.py, stage the changes, and amend the last commit:



```
PS C:\Users\vigne\OneDrive\Desktop\giit\adavnced-git-project> notepad script.py
PS C:\Users\vigne\OneDrive\Desktop\giit\adavnced-git-project> git add script.py
PS C:\Users\vigne\OneDrive\Desktop\giit\adavnced-git-project> git commit -m "Add initial script.py file with a comment"
[master f42fa15] Add initial script.py file with a comment
 1 file changed, 3 insertions(+), 1 deletion(-)
PS C:\Users\vigne\OneDrive\Desktop\giit\adavnced-git-project> git revert HEAD
[master 35d3828] Revert "Add initial script.py file with  comment"
 1 file changed, 1 insertion(+), 3 deletions(-)
```

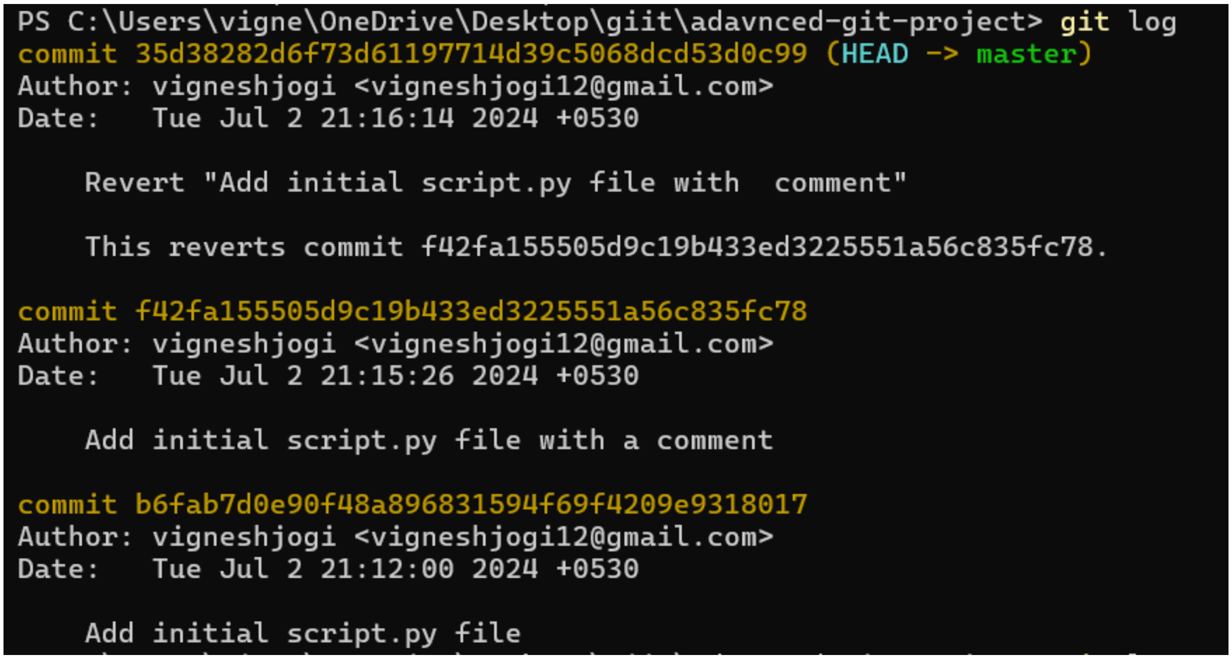
Figure 16

e) Revert Last Commit:



```
PS C:\Users\vigne\OneDrive\Desktop\giit\adavnced-git-project> git revert HEAD
[master 35d3828] Revert "Add initial script.py file with  comment"
 1 file changed, 1 insertion(+), 3 deletions(-)
```

Figure 17

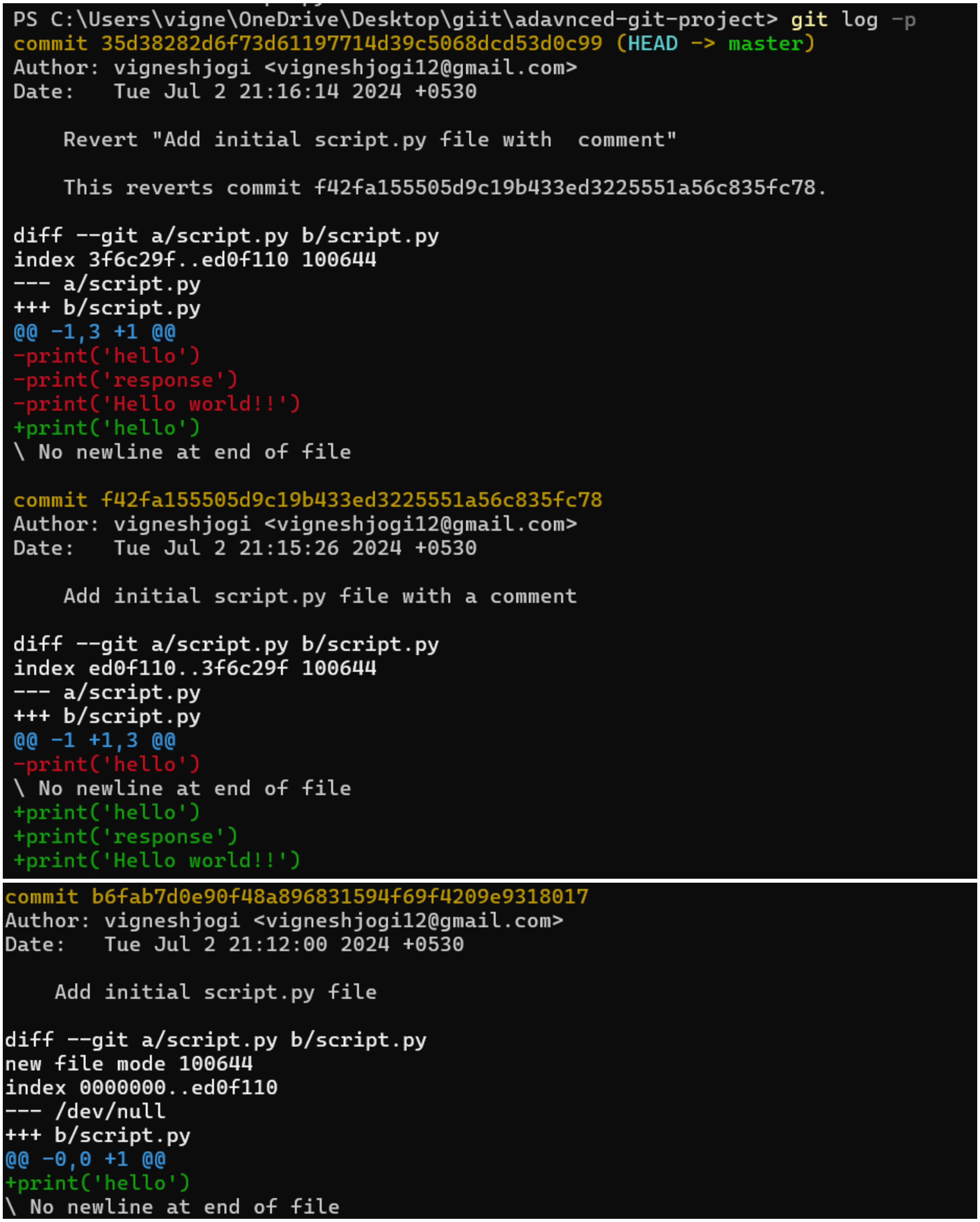


```
PS C:\Users\vigne\OneDrive\Desktop\giit\adavnced-git-project> git log
commit 35d38282d6f73d61197714d39c5068dcd53d0c99 (HEAD -> master)
Author: vigneshjogi <vigneshjogi12@gmail.com>
Date:   Tue Jul 2 21:16:14 2024 +0530

    Revert "Add initial script.py file with  comment"

    This reverts commit f42fa155505d9c19b433ed3225551a56c835fc78.

commit f42fa155505d9c19b433ed3225551a56c835fc78
Author: vigneshjogi <vigneshjogi12@gmail.com>
Date:   Tue Jul 2 21:15:26 2024 +0530

    Add initial script.py file with a comment

commit b6fab7d0e90f48a896831594f69f4209e9318017
Author: vigneshjogi <vigneshjogi12@gmail.com>
Date:   Tue Jul 2 21:12:00 2024 +0530

    Add initial script.py file
```

Figure 18

f) View Commit History and Details:

- View commit history and details using commit IDs (full and abbreviated):



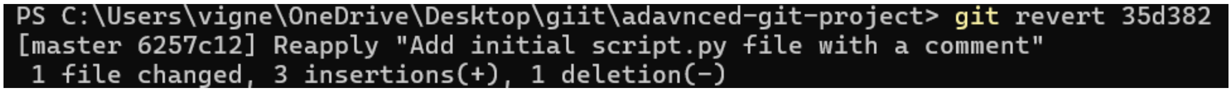
```
PS C:\Users\vigne\OneDrive\Desktop\giit\adavnced-git-project> git log -p
commit 35d38282d6f73d61197714d39c5068dcd53d0c99 (HEAD -> master)
Author: vigneshjogi <vigneshjogi12@gmail.com>
Date:   Tue Jul 2 21:16:14 2024 +0530

    Revert "Add initial script.py file with  comment"

    This reverts commit f42fa155505d9c19b433ed3225551a56c835fc78.

diff --git a/script.py b/script.py
index 3f6c29f..ed0f110 100644
--- a/script.py
+++ b/script.py
@@ -1,3 +1 @@
-print('hello')
-print('response')
-print('Hello world!!')
+print('hello')
\ No newline at end of file

commit f42fa155505d9c19b433ed3225551a56c835fc78
Author: vigneshjogi <vigneshjogi12@gmail.com>
Date:   Tue Jul 2 21:15:26 2024 +0530

    Add initial script.py file with a comment

diff --git a/script.py b/script.py
index ed0f110..3f6c29f 100644
--- a/script.py
+++ b/script.py
@@ -1 +1,3 @@
-print('hello')
\ No newline at end of file
+print('hello')
+print('response')
+print('Hello world!!')
```

```
commit b6fab7d0e90f48a896831594f69f4209e9318017
Author: vigneshjogi <vigneshjogi12@gmail.com>
Date:   Tue Jul 2 21:12:00 2024 +0530

    Add initial script.py file

diff --git a/script.py b/script.py
new file mode 100644
index 0000000..ed0f110
--- /dev/null
+++ b/script.py
@@ -0,0 +1 @@
+print('hello')
\ No newline at end of file
```

Figure 19

g) Identify and Revert a Specific Commit:

- Identify a specific commit to revert and execute the revert command:



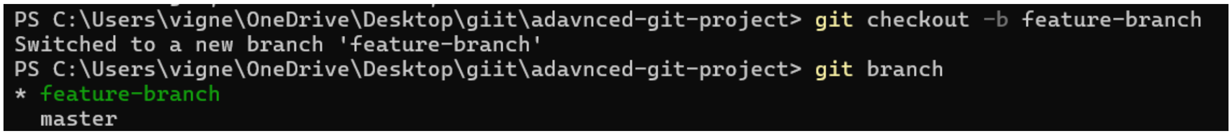
```
PS C:\Users\vigne\OneDrive\Desktop\giit\adavnced-git-project> git revert 35d382
[master 6257c12] Reapply "Add initial script.py file with a comment"
 1 file changed, 3 insertions(+), 1 deletion(-)
```

Figure 20

8. Implementing a New Feature Branch:

a) Create and Work on Feature Branch:

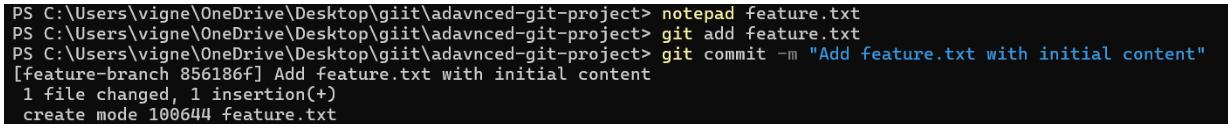
- Create a new branch for a new feature



```
PS C:\Users\vigne\OneDrive\Desktop\giit\adavnced-git-project> git checkout -b feature-branch
Switched to a new branch 'feature-branch'
PS C:\Users\vigne\OneDrive\Desktop\giit\adavnced-git-project> git branch
* feature-branch
  master
```

Figure 21

- Add a new file feature.txt, commit initial content:

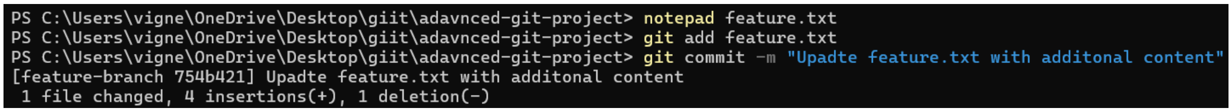


```
PS C:\Users\vigne\OneDrive\Desktop\giit\adavnced-git-project> notepad feature.txt
PS C:\Users\vigne\OneDrive\Desktop\giit\adavnced-git-project> git add feature.txt
PS C:\Users\vigne\OneDrive\Desktop\giit\adavnced-git-project> git commit -m "Add feature.txt with initial content"
[feature-branch 856186f] Add feature.txt with initial content
 1 file changed, 1 insertion(+)
 create mode 100644 feature.txt
```

Figure 22

b) Make Further Changes and Commit:

- Make additional changes to feature.txt, commit them

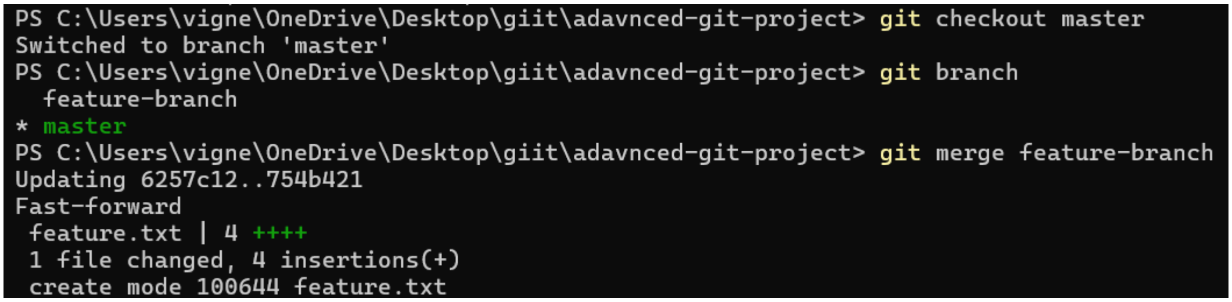


```
PS C:\Users\vigne\OneDrive\Desktop\giit\adavnced-git-project> notepad feature.txt
PS C:\Users\vigne\OneDrive\Desktop\giit\adavnced-git-project> git add feature.txt
PS C:\Users\vigne\OneDrive\Desktop\giit\adavnced-git-project> git commit -m "Upadte feature.txt with additonal content"
[feature-branch 754b421] Upadte feature.txt with additonal content
 1 file changed, 4 insertions(+), 1 deletion(-)
```

Figure 23

c) Merge Feature Branch into Main:

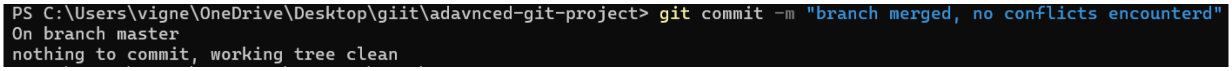
- Switch to main branch and merge feature-branch:



```
PS C:\Users\vigne\OneDrive\Desktop\giit\adavnced-git-project> git checkout master
Switched to branch 'master'
PS C:\Users\vigne\OneDrive\Desktop\giit\adavnced-git-project> git branch
  feature-branch
* master
PS C:\Users\vigne\OneDrive\Desktop\giit\adavnced-git-project> git merge feature-branch
Updating 6257c12..754b421
Fast-forward
 feature.txt | 4 ++++
 1 file changed, 4 insertions(+)
 create mode 100644 feature.txt
```

Figure 24

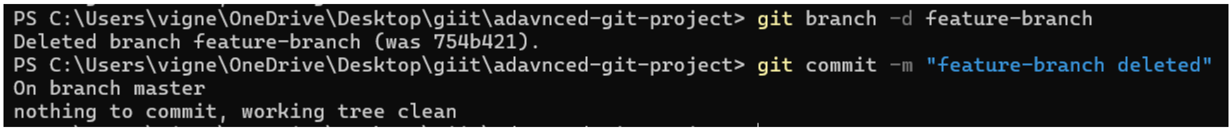
- Resolve any conflicts if encountered, commit the merge:



```
PS C:\Users\vigne\OneDrive\Desktop\giit\adavnced-git-project> git commit -m "branch merged, no conflicts encounterd"
On branch master
nothing to commit, working tree clean
```

Figure 25

d) Delete Feature Branch:



```
PS C:\Users\vigne\OneDrive\Desktop\giit\adavnced-git-project> git branch -d feature-branch
Deleted branch feature-branch (was 754b421).
PS C:\Users\vigne\OneDrive\Desktop\giit\adavnced-git-project> git commit -m "feature-branch deleted"
On branch master
nothing to commit, working tree clean
```

Figure 26