Lab Exercise 1- Working with Git Revert

Lab Exercise: Git Revert

This exercise will guide you through reverting changes in Git. The git revert command is used to create a new commit that undoes the changes introduced by a previous commit without modifying the history.

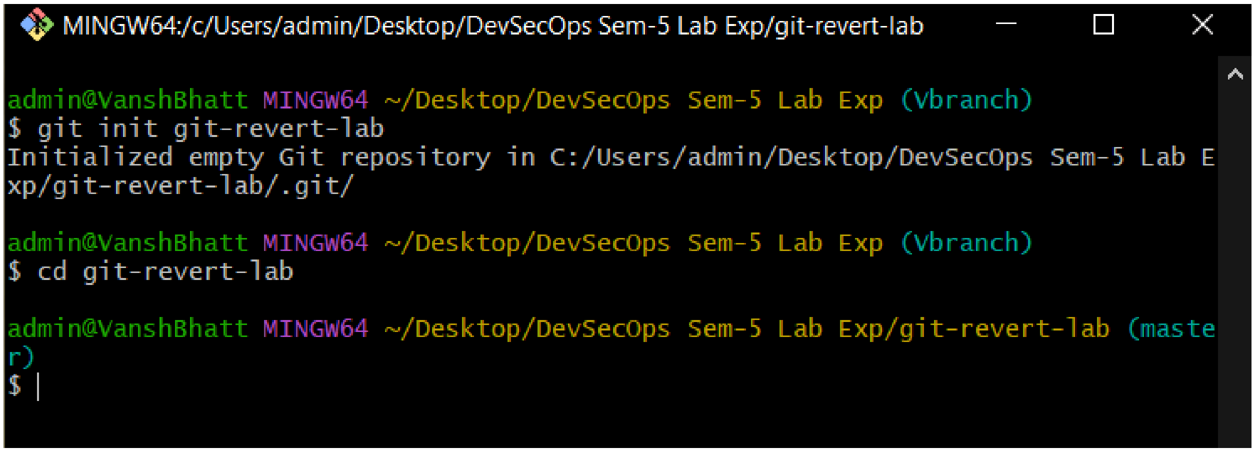
Objective

- Learn how to use git revert to undo changes from specific commits.
- Practice handling merge conflicts during a revert.

Prerequisites

1. Install Git on your system.
2. Set up a Git repository

```
git init git-revert-lab

cd git-revert-lab
```



MINGW64:/c/Users/admin/Desktop/DevSecOps Sem-5 Lab Exp/git-revert-lab

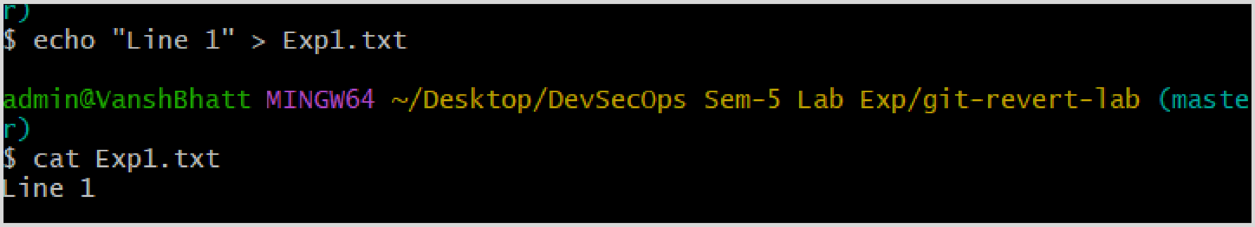
```
admin@VanshBhatt MINGW64 ~/Desktop/DevSecOps Sem-5 Lab Exp (Vbranch)
$ git init git-revert-lab
Initialized empty Git repository in C:/Users/admin/Desktop/DevSecOps Sem-5 Lab E
xp/git-revert-lab/.git/

admin@VanshBhatt MINGW64 ~/Desktop/DevSecOps Sem-5 Lab Exp (Vbranch)
$ cd git-revert-lab

admin@VanshBhatt MINGW64 ~/Desktop/DevSecOps Sem-5 Lab Exp/git-revert-lab (maste
r)
$ |
```

Steps

1. Set Up the Repository

1. Create a file:

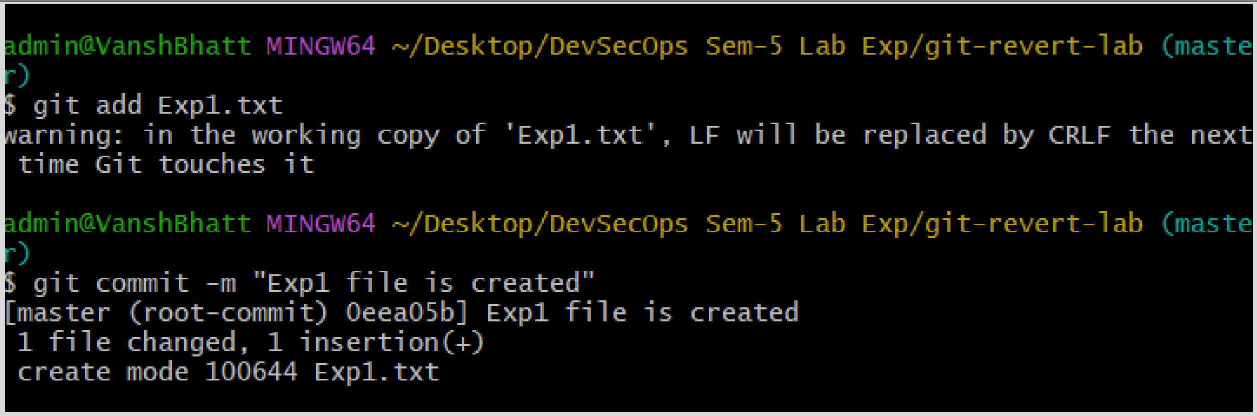
```
echo "Line 1" > file.txt
```



```
r)
$ echo "Line 1" > Exp1.txt

admin@VanshBhatt MINGW64 ~/Desktop/DevSecOps Sem-5 Lab Exp/git-revert-lab (maste
r)
$ cat Exp1.txt
Line 1
```

2. Stage and commit the file:

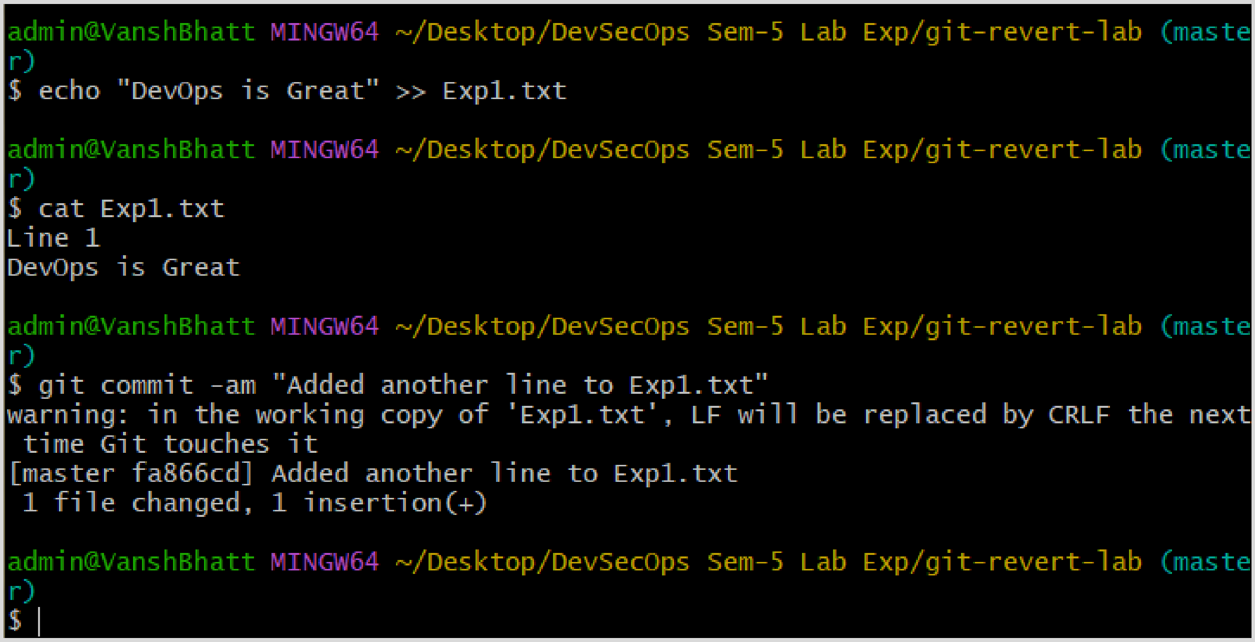
```
git add file.txt

git commit -m "Initial commit: Add file.txt with Line 1"
```



```
admin@VanshBhatt MINGW64 ~/Desktop/DevSecOps Sem-5 Lab Exp/git-revert-lab (maste
r)
$ git add Exp1.txt
warning: in the working copy of 'Exp1.txt', LF will be replaced by CRLF the next
 time Git touches it

admin@VanshBhatt MINGW64 ~/Desktop/DevSecOps Sem-5 Lab Exp/git-revert-lab (maste
r)
$ git commit -m "Exp1 file is created"
[master (root-commit) 0eea05b] Exp1 file is created
 1 file changed, 1 insertion(+)
 create mode 100644 Exp1.txt
```

3. Add more changes:

```
echo "Line 2" >> file.txt


git commit -am "Add Line 2 to file.txt"
```

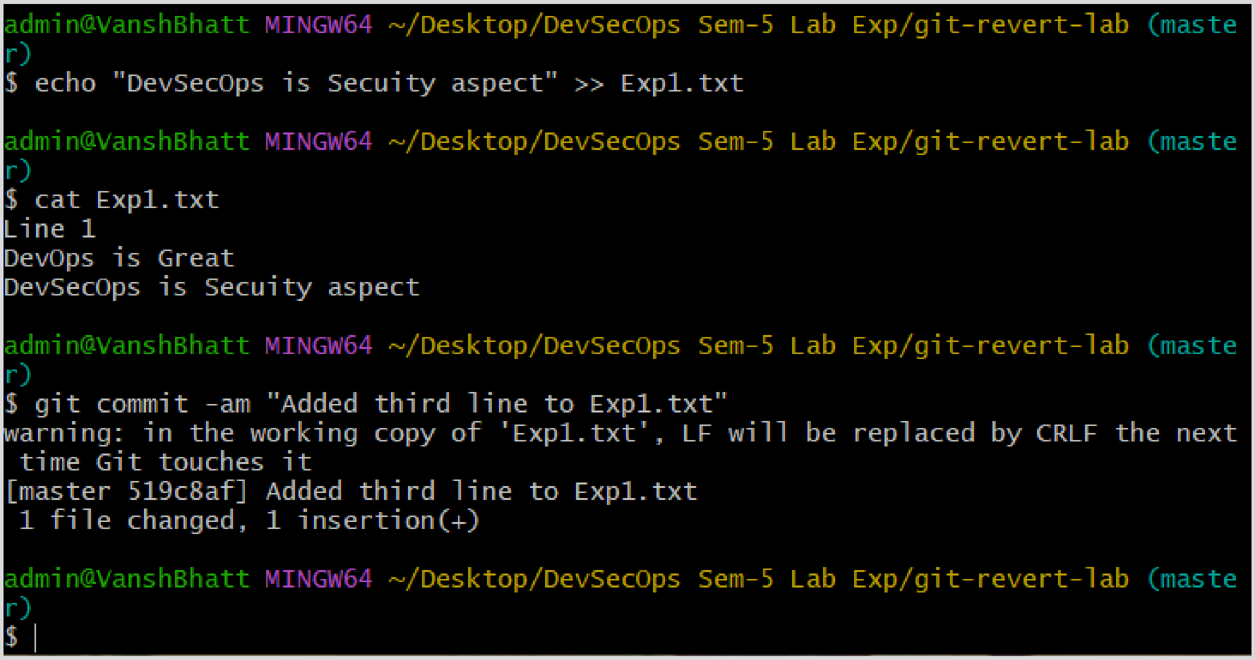


```
admin@VanshBhatt MINGW64 ~/Desktop/DevSecOps Sem-5 Lab Exp/git-revert-lab (maste
r)
$ echo "DevOps is Great" >> Exp1.txt

admin@VanshBhatt MINGW64 ~/Desktop/DevSecOps Sem-5 Lab Exp/git-revert-lab (maste
r)
$ cat Exp1.txt
Line 1
DevOps is Great

admin@VanshBhatt MINGW64 ~/Desktop/DevSecOps Sem-5 Lab Exp/git-revert-lab (maste
r)
$ git commit -am "Added another line to Exp1.txt"
warning: in the working copy of 'Exp1.txt', LF will be replaced by CRLF the next
 time Git touches it
[master fa866cd] Added another line to Exp1.txt
 1 file changed, 1 insertion(+)

admin@VanshBhatt MINGW64 ~/Desktop/DevSecOps Sem-5 Lab Exp/git-revert-lab (maste
r)
$ |
```

4. Add another change:

```
echo "Line 3" >> file.txt
```

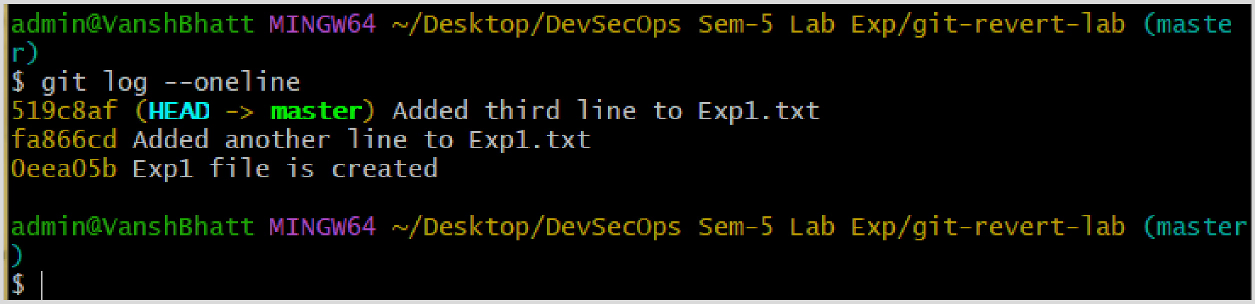
```
git commit -am "Add Line 3 to file.txt"
```



```
admin@VanshBhatt MINGW64 ~/Desktop/DevSecOps Sem-5 Lab Exp/git-revert-lab (maste
r)
$ echo "DevSecOps is Secuity aspect" >> Exp1.txt

admin@VanshBhatt MINGW64 ~/Desktop/DevSecOps Sem-5 Lab Exp/git-revert-lab (maste
r)
$ cat Exp1.txt
Line 1
DevOps is Great
DevSecOps is Secuity aspect

admin@VanshBhatt MINGW64 ~/Desktop/DevSecOps Sem-5 Lab Exp/git-revert-lab (maste
r)
$ git commit -am "Added third line to Exp1.txt"
warning: in the working copy of 'Exp1.txt', LF will be replaced by CRLF the next
 time Git touches it
[master 519c8af] Added third line to Exp1.txt
 1 file changed, 1 insertion(+)

admin@VanshBhatt MINGW64 ~/Desktop/DevSecOps Sem-5 Lab Exp/git-revert-lab (maste
r)
$ |
```

5. Verify the commit history:

```
git log –oneline
```



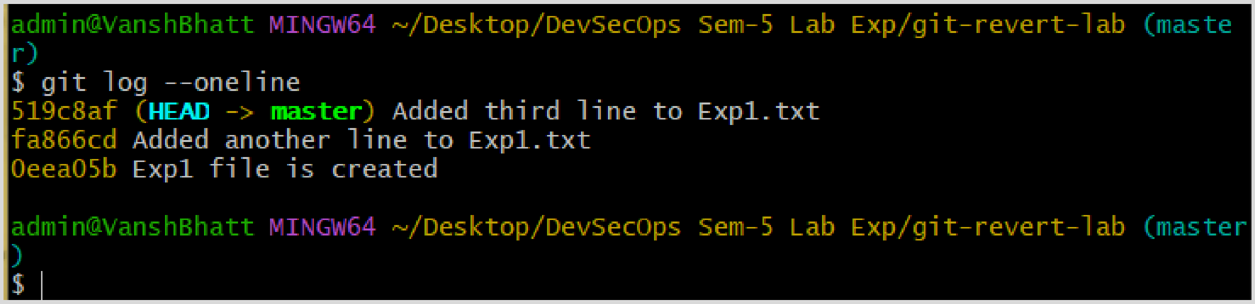
```
admin@VanshBhatt MINGW64 ~/Desktop/DevSecOps Sem-5 Lab Exp/git-revert-lab (maste
r)
$ git log --oneline
519c8af (HEAD -> master) Added third line to Exp1.txt
fa866cd Added another line to Exp1.txt
0eea05b Exp1 file is created

admin@VanshBhatt MINGW64 ~/Desktop/DevSecOps Sem-5 Lab Exp/git-revert-lab (master
)
$ |
```

Example output:

```
c3f6b6b (HEAD -> main) Add Line 3 to file.txt

8b2a1c1 Add Line 2 to file.txt
```

```
4d5f8e9 Initial commit: Add file.txt with Line 1
```

Output Received:-



```
admin@VanshBhatt MINGW64 ~/Desktop/DevSecOps Sem-5 Lab Exp/git-revert-lab (master)
$ git log --oneline
519c8af (HEAD -> master) Added third line to Exp1.txt
fa866cd Added another line to Exp1.txt
0eea05b Exp1 file is created

admin@VanshBhatt MINGW64 ~/Desktop/DevSecOps Sem-5 Lab Exp/git-revert-lab (master)
$ 
```

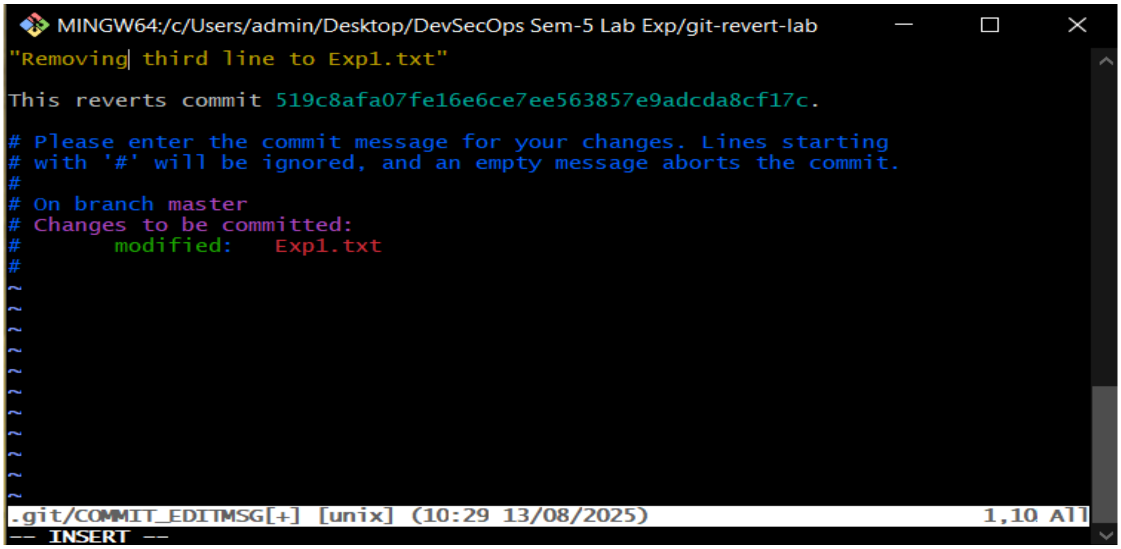
---

2. Revert the Last Commit

1. Revert the most recent commit:

```
git revert HEAD
```

2. Git will open a text editor for the commit message. Save and close the editor to complete the revert.



MINGW64:/c/Users/admin/Desktop/DevSecOps Sem-5 Lab Exp/git-revert-lab

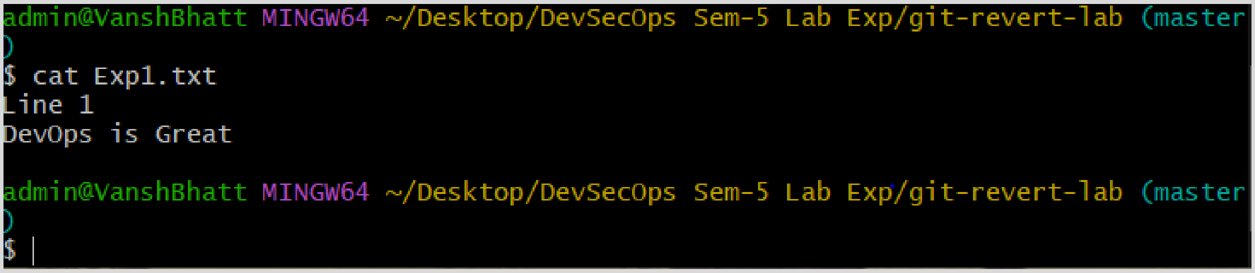
```
"Removing third line to Exp1.txt"

This reverts commit 519c8afa07fe16e6ce7ee563857e9adcda8cf17c.

# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the commit.
#
# On branch master
# Changes to be committed:
#       modified:   Exp1.txt
#
~
~
~
~
~
~
~
~
~
~
~
.git/COMMIT_EDITMSG[+] [unix] (10:29 13/08/2025)                1,10 All
-- INSERT --
```

3. Verify the contents of file.txt:

```
cat file.txt
```

Output:

```
Line 1

Line 2
```



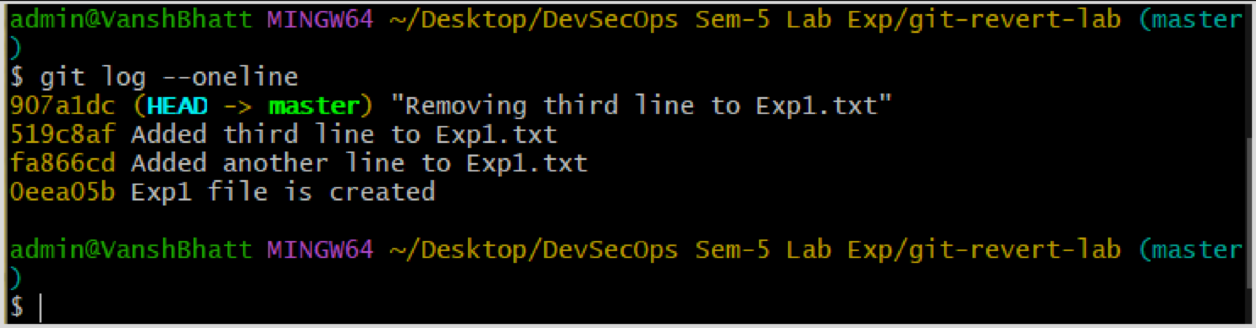
```
admin@VanshBhatt MINGW64 ~/Desktop/DevSecOps Sem-5 Lab Exp/git-revert-lab (master
)
$ cat Exp1.txt
Line 1
DevOps is Great

admin@VanshBhatt MINGW64 ~/Desktop/DevSecOps Sem-5 Lab Exp/git-revert-lab (master
)
$ |
```

4. Check the commit history:

```
git log --oneline
```

Example output:

```
e5d2f8a Revert "Add Line 3 to file.txt"

c3f6b6b Add Line 3 to file.txt

8b2a1c1 Add Line 2 to file.txt

4d5f8e9 Initial commit: Add file.txt with Line 1
```



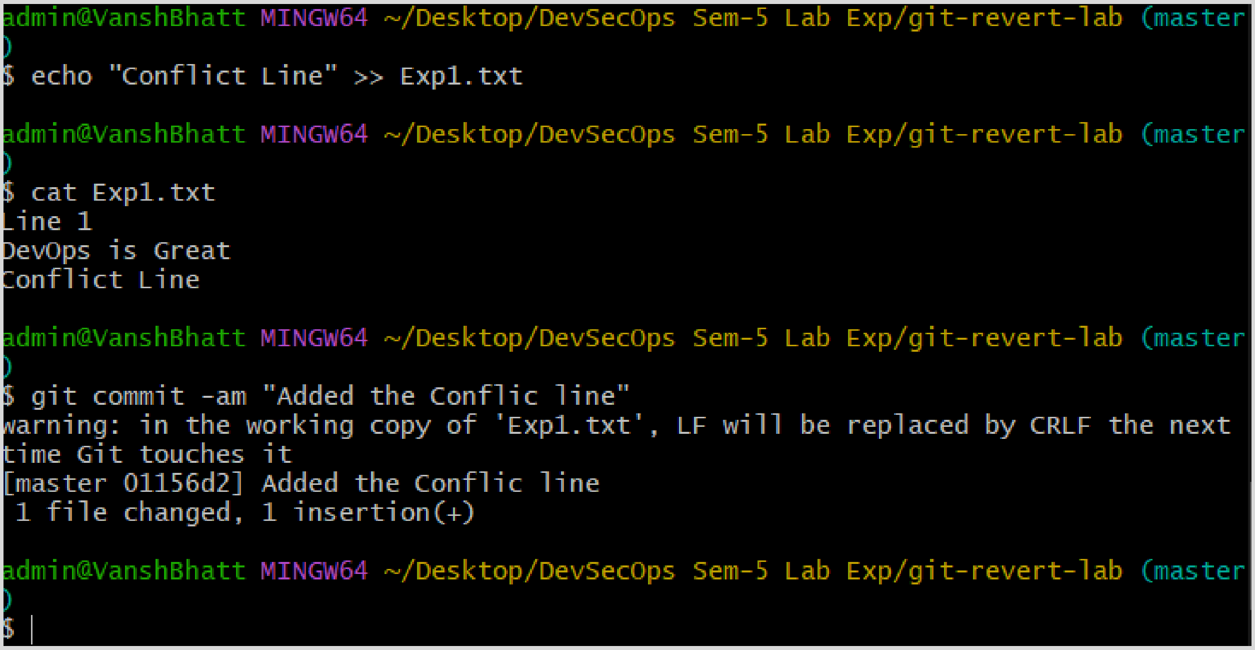
```
admin@VanshBhatt MINGW64 ~/Desktop/DevSecOps Sem-5 Lab Exp/git-revert-lab (master
)
$ git log --oneline
907a1dc (HEAD -> master) "Removing third line to Exp1.txt"
519c8af Added third line to Exp1.txt
fa866cd Added another line to Exp1.txt
0eea05b Exp1 file is created

admin@VanshBhatt MINGW64 ~/Desktop/DevSecOps Sem-5 Lab Exp/git-revert-lab (master
)
$ |
```

---

3. Handle Merge Conflicts During Revert

1. Modify file.txt:

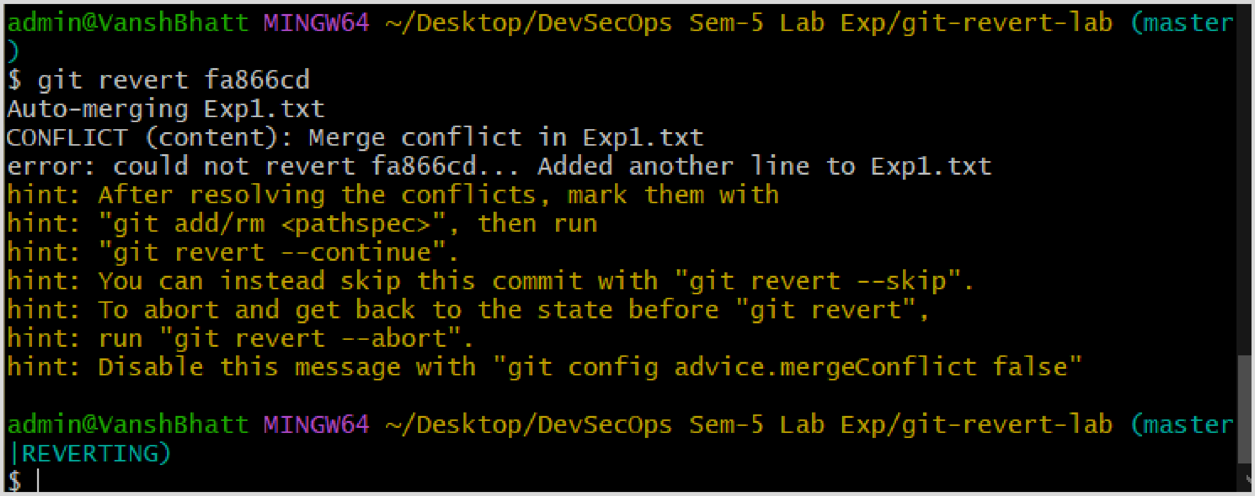
```
echo "Conflict Line" >> file.txt

git commit -am "Add Conflict Line"
```



```
admin@VanshBhatt MINGW64 ~/Desktop/DevSecOps Sem-5 Lab Exp/git-revert-lab (master
)
$ echo "Conflict Line" >> Exp1.txt

admin@VanshBhatt MINGW64 ~/Desktop/DevSecOps Sem-5 Lab Exp/git-revert-lab (master
)
$ cat Exp1.txt
Line 1
DevOps is Great
Conflict Line

admin@VanshBhatt MINGW64 ~/Desktop/DevSecOps Sem-5 Lab Exp/git-revert-lab (master
)
$ git commit -am "Added the Conflic line"
warning: in the working copy of 'Exp1.txt', LF will be replaced by CRLF the next
time Git touches it
[master 01156d2] Added the Conflic line
 1 file changed, 1 insertion(+)

admin@VanshBhatt MINGW64 ~/Desktop/DevSecOps Sem-5 Lab Exp/git-revert-lab (master
)
$ |
```

2. Revert the second commit (8b2a1c1 again) to trigger a conflict:

```
git revert 8b2a1c1
```



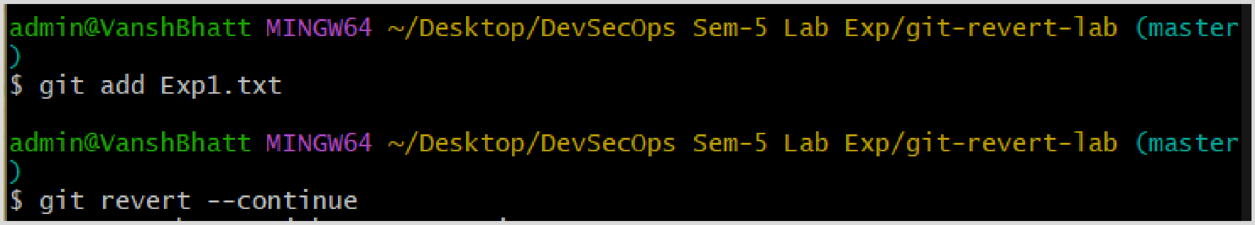
```
admin@VanshBhatt MINGW64 ~/Desktop/DevSecOps Sem-5 Lab Exp/git-revert-lab (master
)
$ git revert fa866cd
Auto-merging Exp1.txt
CONFLICT (content): Merge conflict in Exp1.txt
error: could not revert fa866cd... Added another line to Exp1.txt
hint: After resolving the conflicts, mark them with
hint: "git add/rm <pathspec>", then run
hint: "git revert --continue".
hint: You can instead skip this commit with "git revert --skip".
hint: To abort and get back to the state before "git revert",
hint: run "git revert --abort".
hint: Disable this message with "git config advice.mergeConflict false"

admin@VanshBhatt MINGW64 ~/Desktop/DevSecOps Sem-5 Lab Exp/git-revert-lab (master
|REVERTING)
$ |
```

3. Git will indicate a conflict. Resolve it:
   - Open file.txt and remove conflict markers.
   - Keep the desired lines.
4. Stage the resolved file:

```
git add file.txt
```

5. Complete the revert:

```
git revert –continue
```



```
admin@VanshBhatt MINGW64 ~/Desktop/DevSecOps Sem-5 Lab Exp/git-revert-lab (master
)
$ git add Exp1.txt

admin@VanshBhatt MINGW64 ~/Desktop/DevSecOps Sem-5 Lab Exp/git-revert-lab (master
)
$ git revert --continue
```