

# Lab Exercise 1- Working with Git Revert

Name- Misha (B2 DevOps)

SAP ID- 500119679

---

## 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
```

```
PS C:\Users\Misha> git init git-revert-lab
Initialized empty Git repository in C:/Users/Misha/git-revert-lab/.git/
PS C:\Users\Misha> cd git-revert-lab
```

## Steps

### 1. Set Up the Repository

1. Create a file:

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

2. Stage and commit the file:

```
git add file.txt
```

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

```
PS C:\Users\Misha\git-revert-lab> echo "Line1">file.txt
PS C:\Users\Misha\git-revert-lab> git add file/txt
fatal: pathspec 'file/txt' did not match any files
PS C:\Users\Misha\git-revert-lab> git add file.txt
PS C:\Users\Misha\git-revert-lab> git commit -m "initial commit: add file.txt with line1"
[master (root-commit) ebbbaa3] initial commit: add file.txt with line1
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 file.txt
```

3. Add more changes:

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

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

```

PS C:\Users\Misha\git-revert-lab> echo "line2">>file.txt
PS C:\Users\Misha\git-revert-lab> git commit -m "add line2 to file.txt"
On branch master
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:   file.txt

no changes added to commit (use "git add" and/or "git commit -a")
PS C:\Users\Misha\git-revert-lab> git add .
PS C:\Users\Misha\git-revert-lab> git commit -m "add line2 to file.txt"
[master b444084] add line2 to file.txt
1 file changed, 0 insertions(+), 0 deletions(-)

```

3. Add another change:

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

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

```

PS C:\Users\Misha\git-revert-lab> echo "line3" >> file.txt
PS C:\Users\Misha\git-revert-lab> git add .
PS C:\Users\Misha\git-revert-lab> git commit -m "add line3 to file.txt"
[master 61e1d16] add line3 to file.txt
1 file changed, 0 insertions(+), 0 deletions(-)

```

4. Verify the commit history:

```
git log --oneline
```

```

PS C:\Users\Misha\git-revert-lab> git log --oneline
61e1d16 (HEAD -> master) add line3 to file.txt
b444084 add line2 to file.txt
ebbbaa3 initial commit: add file.txt with line1
PS C:\Users\Misha\git-revert-lab> git revert HEAD
[master a88f6b0] Revert "add line3 to file.txt"
1 file changed, 0 insertions(+), 0 deletions(-)

```

## 2. Revert the Last Commit

1. Revert the most recent commit:

git revert HEAD

```
PS C:\Users\Misha\git-revert-lab> git revert HEAD
[master a88f6b0] Revert "add line3 to file.txt"
1 file changed, 0 insertions(+), 0 deletions(-)
```

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

```
Revert "add line3 to file.txt"

This reverts commit 61e1d16599989777266e3a3122fed20c0ce6f51e.

# 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:   file.txt
#
```

3. Verify the contents of file.txt:

cat file.txt

```
PS C:\Users\Misha\git-revert-lab> cat file.txt
Line1
line2
```

#### 4. Check the commit history:

```
git log --oneline
```

```
PS C:\Users\Misha\git-revert-lab> git log --oneline
a88f6b0 (HEAD -> master) Revert "add line3 to file.txt"
61e1d16 add line3 to file.txt
b444084 add line2 to file.txt
ebbbaa3 initial commit: add file.txt with line1
```

---

### 3. Handle Merge Conflicts During Revert

#### 1. Modify file.txt:

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

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

```
PS C:\Users\Misha\git-revert-lab> echo "conflict line" >> file.txt
PS C:\Users\Misha\git-revert-lab> git add .
PS C:\Users\Misha\git-revert-lab> git commit -m "add conflict line"
[master 65d2d5f] add conflict line
1 file changed, 0 insertions(+), 0 deletions(-)
```

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

```
git revert 8b2a1c1
```

```
1 file changed, 0 insertions(+), 0 deletions(-)
PS C:\Users\Misha\git-revert-lab> git revert b444084
warning: Cannot merge binary files: file.txt (HEAD vs. parent of b444084 (ad
dd line2 to file.txt))
Auto-merging file.txt
CONFLICT (content): Merge conflict in file.txt
error: could not revert b444084... addd line2 to file.txt
hint: After resolving the conflicts, mark them with
hint: "git add/rm <paths>", 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"
PS C:\Users\Misha\git-revert-lab> |
```

2. Git will indicate a conflict. Resolve it:

- Open file.txt and remove conflict markers.
- Keep the desired lines.

```
PS C:\Users\Misha\git-revert-lab> get-content -path "C:\Users\Misha\git-reve
rt-lab\file.txt"
Line1
line2
conflict line
PS C:\Users\Misha\git-revert-lab> start-process -filepath "C:\Users\Misha\gi
t-revert-lab\file.txt"
PS C:\Users\Misha\git-revert-lab> git add file.txt
```

2. Stage the resolved file:

git add file.txt

```
Revert "add line2 to file.txt"

This reverts commit b444084e1d4332b7bc604176e4896dced0581e37.

# Conflicts:
#   file.txt

# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the commit.
#
# On branch master
# You are currently reverting commit b444084.
#
# Changes to be committed:
#   modified:   file.txt
#
```

3. Complete the revert:

git revert --continue

```
PS C:\Users\Misha\git-revert-lab> git revert --continue
[master 604d907] Revert "add line2 to file.txt"
 1 file changed, 0 insertions(+), 0 deletions(-)
PS C:\Users\Misha\git-revert-lab>
```