

1 -> 1

```
bash

# a) Create 10 files under your home directory
cd ~
touch cse ise aiml ai_ds ece eee eie iem civil mech

# b) Create 3 directories under your home directory
mkdir itbranch circuitbranch others

# c) Create a new file 'students' and write text to it
echo "Students of SIT" > students

# Create a soft (symbolic) link in /tmp
ln -s ~/students /tmp/students_softlink

# Create a hard link in /tmp
ln ~/students /tmp/students_hardlink

# Check inodes of both links and original file
ls -li ~/students /tmp/students_softlink /tmp/students_hardlink
```

1 -> 2

bash

Copy code

```
day=$(date +%A)

if [ "$day" = "Saturday" ] || [ "$day" = "Sunday" ]; then
    echo "It's a weekend."
else
    echo "It's a weekday."
fi
```

2 -> 1

```
bash

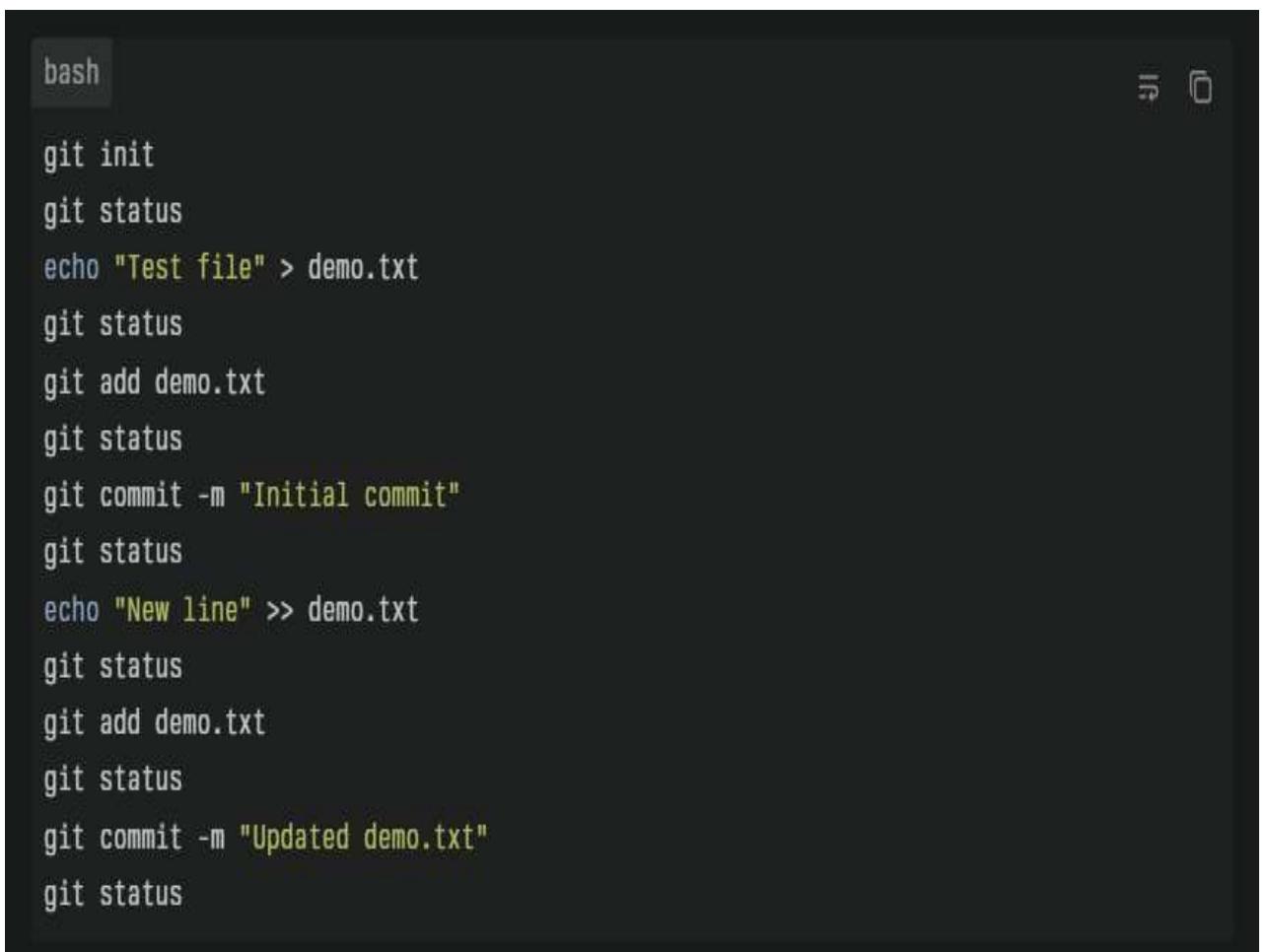
git init
echo "Hello, Git" > example.txt
git hash-object -w example.txt
git add example.txt
git write-tree
git cat-file -p $(git write-tree)
```

2 -> 2

```
bash

git init
echo "Hello, Git" > file1.txt
echo "Second file in Git repository" > file2.txt
git add file1.txt file2.txt
git write-tree
git commit-tree <tree_hash> -m "Initial commit with two files"
git cat-file -p <commit_hash>
git cat-file -p <tree_hash>
git hash-object file1.txt
git hash-object file2.txt
git cat-file -p <blob_hash>
```

3 -> 1



The screenshot shows a terminal window with a dark background and light-colored text. The title bar says "bash". The window contains the following command history:

```
git init
git status
echo "Test file" > demo.txt
git status
git add demo.txt
git status
git commit -m "Initial commit"
git status
echo "New line" >> demo.txt
git status
git add demo.txt
git status
git commit -m "Updated demo.txt"
git status
```

3 -> 2

```
bash
git init
echo "This is file 1" > file1.txt
echo "This is file 2" > file2.txt
git status
git add file1.txt file2.txt
git status
git commit -m "Add file1.txt and file2.txt"
git status
```

3 -> 3

```
bash
echo "This is file 3" > file3.txt
git status
git add file3.txt
git status
git commit -m "Add file3.txt"
git status
```

4 -> 1

```
bash

git init
echo "Hello, Git" > file1.txt
echo "Second file in Git repository" > file2.txt
git add file1.txt file2.txt
git commit -m "Our very first commit in the project"

echo "This is third file in our Git repository" > file3.txt
git add file3.txt
git commit -m "Second commit"

git log --oneline
```

4 -> 2

```
bash

git init
echo "Hello, Git" > file1.txt
echo "Second file in Git repository" > file2.txt
git add file1.txt file2.txt
git commit -m "Our very first commit in the project"

echo "This is third file in our Git repository" > file3.txt
git add file3.txt
git commit -m "Second commit"

rm file1.txt file2.txt file3.txt
git add -u
git commit -m "Third commit"

git log --oneline
```

5 -> 1

```
bash
git init
echo "Hello, Git" > file1.txt
echo "Second file in Git repository" > file2.txt
git add file1.txt file2.txt
git commit -m "Our very first commit in the project"

echo "This is third file in our Git repository" > file3.txt
git add file3.txt
git commit -m "Second commit"

rm file1.txt file2.txt file3.txt
git add -u
git commit -m "Third commit"

git branch
git log --oneline --decorate --graph --all
```

5 -> 2

```
git init
echo "root" > file.txt
git add file.txt
git commit -m "Root commit"

# Commit 1 on master
echo "master 1" >> file.txt
git commit -am "Master commit 1"

# Branch BR-102 from commit 1
git checkout -b BR-102
echo "BR-102 1" >> file.txt
git commit -am "BR-102 commit 1"
echo "BR-102 2" >> file.txt
git commit -am "BR-102 commit 2"

# Branch temp from second commit of BR-102
git checkout -b temp
echo "temp 1" >> file.txt
git commit -am "Temp commit 1"
echo "temp 2" >> file.txt
git commit -am "Temp commit 2"

# Go back to BR-102 and put one commit
git checkout BR-102
echo "BR-102 3" >> file.txt
git commit -am "BR-102 commit 3"

# Go back to master and commit once
git checkout master
echo "master 2" >> file.txt
git commit -am "Master commit 2"

# New branch exp from that commit, 3 commits on exp
git checkout -b exp
echo "exp 1" >> file.txt
git commit -am "Exp commit 1"
echo "exp 2" >> file.txt
git commit -am "Exp commit 2"
echo "exp 3" >> file.txt
git commit -am "Exp commit 3"

# Back to master and commit once
git checkout master
echo "master 3" >> file.txt
git commit -am "Master commit 3"

# Set HEAD at exp
git checkout exp

# Merge temp into exp
git merge temp

# View the branch graph
git log --oneline --decorate --graph --all
```