1. git config --global user.name "janunama"

git config --global user.email [jahnavi.durga@practo.com](mailto:jahnavi.durga@practo.com)

2. git branch feature-branch

git checkout feature-branch

3. git branch

4. git checkout main

5. git reset --soft HEAD~1

6. git checkout -b conflict-branch

7. git branch feature1

git checkout feature1

8. git checkout feature1

echo "Some new content" > newfile.txt

git add newfile.txt

git commit -m "Made changes in feature1 branch"

9. git checkout main

git merge feature1

git commit

git push origin main

10. git checkout conflict-branch

echo "Conflicting content" > newfile.txt

git add newfile.txt

git commit -m "Made conflicting changes in conflict-branch"

11. git merge main

git add <resolved-file>

git commit

12. [conflict-branch 1a2b3c4] Merge branch 'main' into conflict-branch

13. git remote add origin <https://github.com/your-janunama15/cricketscorecard-repository.git>

14.git clone https://github.com/your-username/your-forked-repo.git

cd your-forked-repo

15. git checkout -b new-feature-branch

echo "Some new feature" > feature.txt

git add feature.txt

git commit -m "Add new feature"

git push origin new-feature-branch

16. In `feature.txt`, the variable name `x` is not descriptive. Consider using a more meaningful name, such as `userInput`:

```diff

- let x = getInput();

+ let userInput = getInput();

17. git config --global –edit

gitlol = log --oneline

git gitlol

18. cd /path/to/your/repository

cd .git/hooks

cat << 'EOF' > pre-commit

exec 1>&2

echo "Error: You have trailing whitespace in your changes. Please fix it before committing."

exit 1

fi

EOF

chmod +x pre-commit

git add .

git commit -m "Test commit"

19. git stash

git checkout other-branch

git stash apply

git stash drop

20. git checkout HEAD --<file>

21. git add <file>

Git commit –amend

22.git reset --hard HEAD

23.git show <commit\_hash>

24.git commit –amend

25.git cherry-pick abcdefg

26.git push origin <your\_branch>

27.git reset HEAD <file>

git status

git reset HEAD <file>

git reset HEAD example.txt

28.git rm --cached \*.yml

git rm --cached -r config/

29.git diff-tree --no-commit-id --name-only -r HEAD

30.git fetch

31.git branch -a

git checkout -b my-branch abc1234

git checkout my-branch

git log

32.git clean -fd



