

1. Verify if master is in clean state.
2. Create a branch "GitWork". Add a file "hello.xml".
3. Update the content of "hello.xml" and observe the status
4. Commit the changes to reflect in the branch
5. Switch to master.
6. Add a file "hello.xml" to the master and add some different content than previous.
7. Commit the changes to the master
8. Observe the log by executing "git log --oneline --graph --decorate --all"
9. Check the differences with Git diff tool
10. For better visualization, use P4Merge tool to list out all the differences between master and branch
11. Merge the bran to the master
12. Observe the git mark up.
13. Use 3-way merge tool to resolve the conflict
14. Commit the changes to the master, once done with conflict
15. Observe the git status and add backup file to the .gitignore file.
16. Commit the changes to the .gitignore
17. List out all the available branches
18. Delete the branch, which merge to master.
19. Observe the log by executing "git log --oneline --graph --decorate"

```

MINGW64/c/Users/Admin/GitDemo
zy-fetch]
[--exec-path=<path>] [--html-path] [--man-path] [--info-path]
[-p | --paginate | -P | --no-pager] [--no-replace-objects] [--no-la
[--no-optional-locks] [--no-advice] [--bare] [--git-dir=<path>]
[--work-tree=<path>] [--namespace=<name>] [--config-env=<name>=<env
var>]
<command> [<args>]

Admin@DESKTOP-N2DRSBN MINGW64 ~/GitDemo (Gitwork)
$ git commit -m "Resolve merge conflict ad fix hello.xml content"
[gitwork 36ba214] Resolve merge conflict ad fix hello.xml content
1 file changed, 3 insertions(+)

Admin@DESKTOP-N2DRSBN MINGW64 ~/GitDemo (Gitwork)
$ git add hello.xml

Admin@DESKTOP-N2DRSBN MINGW64 ~/GitDemo (Gitwork)
$ git commit -m "Resolve merger conflict"
On branch Gitwork
nothing to commit, working tree clean

Admin@DESKTOP-N2DRSBN MINGW64 ~/GitDemo (Gitwork)
$ git staud
git: 'staud' is not a git command. See 'git --help'.

The most similar command is
    status

Admin@DESKTOP-N2DRSBN MINGW64 ~/GitDemo (Gitwork)
$ git status
On branch Gitwork
nothing to commit, working tree clean

Admin@DESKTOP-N2DRSBN MINGW64 ~/GitDemo (Gitwork)
$ echo "*.orig" >> .gitignore

Admin@DESKTOP-N2DRSBN MINGW64 ~/GitDemo (Gitwork)
$ git add .gitignore
warning: in the working copy of '.gitignore', LF will be replaced by CRLF the
next time Git touches it
go
Admin@DESKTOP-N2DRSBN MINGW64 ~/GitDemo (Gitwork)
$ git commit -m :Ignore merge backup files
> ^C

```

```
MINGW64/c:/Users/Admin/GitDemo
$ git commit -m "Ignore merge backup files"
> ^C

Admin@DESKTOP-N2DRSBN MINGW64 ~/GitDemo (Gitwork)
$ git commit -m "Ignore merge backup files"
[Gitwork 1ec53a4] Ignore merge backup files
1 file changed, 1 insertion(+)

Admin@DESKTOP-N2DRSBN MINGW64 ~/GitDemo (Gitwork)
$ git branch
* Gitwork
  main

Admin@DESKTOP-N2DRSBN MINGW64 ~/GitDemo (Gitwork)
$ git branch -d Gitwork
error: cannot delete branch 'Gitwork' used by worktree at 'C:/Users/Admin/GitDemo'

Admin@DESKTOP-N2DRSBN MINGW64 ~/GitDemo (Gitwork)
$ ^C

Admin@DESKTOP-N2DRSBN MINGW64 ~/GitDemo (Gitwork)
$ git checkout main
Switched to branch 'main'
Your branch is ahead of 'origin/main' by 5 commits.
(use 'git push' to publish your local commits)

Admin@DESKTOP-N2DRSBN MINGW64 ~/GitDemo (main)
$ git branch -d Gitwork
error: the branch 'Gitwork' is not fully merged
hint: If you are sure you want to delete it, run 'git branch -D Gitwork'
hint: Disable this message with 'git config set advice.forceDeleteBranch false'

Admin@DESKTOP-N2DRSBN MINGW64 ~/GitDemo (main)
$ ^C

Admin@DESKTOP-N2DRSBN MINGW64 ~/GitDemo (main)
$ git log main..Gitwork
commit 1ec53a47e4a9546ca63530a0707a295c1fa37fb6 (Gitwork)
Author: Siddharth <whenpongal@gmail.com>
Date: Tue Aug 12 11:06:50 2025 +0530

    Ignore merge backup files
```