

# Git and GitHub Challenges

## - Resolve Merge Conflicts:

- Create a merge conflict intentionally (two users editing the same line).
- Resolve the conflict and push the changes.

```
$ git merge Polo
Auto-merging Sample
CONFLICT (content): Merge conflict in Sample
Automatic merge failed; fix conflicts and then commit the result.

Viqaas@LAPTOP-VG025G9U MINGW64 ~/Desktop/Testing/Testing (Marco|MERGI
G)
$ vi sample

Viqaas@LAPTOP-VG025G9U MINGW64 ~/Desktop/Testing/Testing (Marco|MERGI
G)
$ git add .

Viqaas@LAPTOP-VG025G9U MINGW64 ~/Desktop/Testing/Testing (Marco|MERGI
G)
$

Viqaas@LAPTOP-VG025G9U MINGW64 ~/Desktop/Testing/Testing (Marco|MERGI
G)
$ git commit -m "done"
[Marco ae77231] done
```

## - Recover Deleted Branch:

- Delete a local branch and then recover it using the reflog.

```
Viqaas@LAPTOP-VG025G9U MINGW64 ~/Desktop/Testing/Testing (main)
$ git branch --list
  Polo
* main

Viqaas@LAPTOP-VG025G9U MINGW64 ~/Desktop/Testing/Testing (main)
$ git branch Marco 103b0e2

Viqaas@LAPTOP-VG025G9U MINGW64 ~/Desktop/Testing/Testing (main)
$ git AMrco
git: 'AMrco' is not a git command. See 'git --help'.

Viqaas@LAPTOP-VG025G9U MINGW64 ~/Desktop/Testing/Testing (main)
$ git checkout Marco
Switched to branch 'Marco'

Viqaas@LAPTOP-VG025G9U MINGW64 ~/Desktop/Testing/Testing (Marco)
$ ls
DevOps  README.md  why  fighter  password
Line   Sample    a    killer   test.txt
```

## - Undo Wrong Push:

# Git and GitHub Challenges

- Push a wrong commit to GitHub, then undo it without losing history.

```
$ git log -1
commit 341130fdd0fd279199a4d3e16b9fc9eb03032f00 (HEAD -> Marco)
Author: ViqaasTaimoor <viqaastaimoor@gmail.com>
Date:   Fri Dec 26 15:28:18 2025 +0530

    Revert "wrong commit"

    This reverts commit c8ac173640d6979722d26fc4227a1089d1337b4d.
```

- **Amend a Commit:**
- Make a commit, then add a missing file to it using `git commit --amend`.

```
$ git commit --amend
[Marco 86bd8fe] Revert "wrong commit"
Date: Fri Dec 26 15:28:18 2025 +0530
2 files changed, 0 insertions(+), 0 deletions(-)
create mode 100644 a
create mode 100644 pat
```

# Git and GitHub Challenges

- Take a specific commit from one branch and apply it to another branch.

```
Viqaas@LAPTOP-VG025G9U MINGW64 ~/Desktop/Testing/Testing (Marco)
$ git checkout Polo
Switched to branch 'Polo'
Your branch and 'origin/Polo' have diverged,
and have 2 and 2 different commits each, respectively.
(use "git pull" to merge the remote branch into yours)

Viqaas@LAPTOP-VG025G9U MINGW64 ~/Desktop/Testing/Testing (Polo)
$ git log -1
commit 6b393ac4689fc01a0cf0d96b64c87a6e568b5936 (HEAD -> Polo)
Author: ViqaasTaimoor <viqaastaimoor@gmail.com>
Date:   Fri Dec 26 15:02:37 2025 +0530

    modified

Viqaas@LAPTOP-VG025G9U MINGW64 ~/Desktop/Testing/Testing (Polo)
$ git cherry-pick 86bd8
[Polo 584bc9c] Revert "wrong commit"
Date: Fri Dec 26 15:28:18 2025 +0530
2 files changed, 0 insertions(+), 0 deletions(-)
create mode 100644 a
create mode 100644 pat

Viqaas@LAPTOP-VG025G9U MINGW64 ~/Desktop/Testing/Testing (Polo)
$ git log -1
commit 584bc9c23756b3d1dc719702ab178bb56b742b33 (HEAD -> Polo)
Author: ViqaasTaimoor <viqaastaimoor@gmail.com>
Date:   Fri Dec 26 15:28:18 2025 +0530

    Revert "wrong commit"

    This reverts commit c8ac173640d6979722d26fc4227a1089d1337b4d.
```

- Reorder and squash multiple commits into a single clean commit.

```
Viqaas@LAPTOP-VG025G9U MINGW64 ~/Desktop/Testing/Testing (Polo)
$ git log -2
commit 5de59133d7b4659cb36302c158eacfb487ac34ed (HEAD -> Polo)
Author: ViqaasTaimoor <viqaastaimoor@gmail.com>
Date:   Fri Dec 26 15:02:37 2025 +0530

    modified

    Revert "wrong commit"

    This reverts commit c8ac173640d6979722d26fc4227a1089d1337b4d.

    added why

    added fighter

    I did not
```

# Git and GitHub Challenges

- Create a version tag (v1.0), push it to GitHub, then delete and restore it.

```
viqaas@LAPTOP-VG025G9U MINGW64 ~/Desktop/Testing/Testing (Polo)
$ git fetch --tags

viqaas@LAPTOP-VG025G9U MINGW64 ~/Desktop/Testing/Testing (Polo)
$ git push origin v1.0
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 12 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (5/5), 676 bytes | 676.00 KiB/s, done.
Total 5 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:ViqaasTaimoor/Testing.git
 * [new tag]          v1.0 -> v1.0
```

- Clone only a subdirectory of a repo using sparse checkout.

```
[root@ip-172-31-72-11 Testing]# ls
DevOps  Fight  Line  README.md  why  fighter  password  test.txt
[root@ip-172-31-72-11 Testing]# git checkout Polo
Already on 'Polo'
Your branch is up to date with 'origin/Polo'.
[root@ip-172-31-72-11 Testing]# git sparse-checkout set why
[root@ip-172-31-72-11 Testing]# git sparse-checkout reapply
[root@ip-172-31-72-11 Testing]# ls
DevOps  Line  README.md  why  fighter  password  test.txt
```

# Git and GitHub Challenges

- Demonstrate the difference between `git reset --hard` and `git revert` in a repo.

```
Viqaas@LAPTOP-VG025G9U MINGW64 ~/Desktop/Testing/Testing (Polo)
$ git log -3
commit efc9f265c0d4b2863039c48ea4b963689ead0ff9 (HEAD -> Polo)
Author: ViqaasTaimoor <viqaastaimoor@gmail.com>
Date:   Fri Dec 26 17:14:25 2025 +0530

    testings

commit e65e4cb7f8973d56d6dc3242c4bbcf4f1ba7c3c9
Author: ViqaasTaimoor <viqaastaimoor@gmail.com>
Date:   Fri Dec 26 17:14:18 2025 +0530

    testing

commit 077de34b9b74e2b6876432887033d2d70e42a4f2
Author: ViqaasTaimoor <viqaastaimoor@gmail.com>
Date:   Fri Dec 26 17:14:09 2025 +0530

    test

Viqaas@LAPTOP-VG025G9U MINGW64 ~/Desktop/Testing/Testing (Polo)
$ git reset --hard HEAD~2
HEAD is now at 077de34 test

Viqaas@LAPTOP-VG025G9U MINGW64 ~/Desktop/Testing/Testing (Polo)
$ git log -2
commit 077de34b9b74e2b6876432887033d2d70e42a4f2 (HEAD -> Polo)
Author: ViqaasTaimoor <viqaastaimoor@gmail.com>
Date:   Fri Dec 26 17:14:09 2025 +0530

    test

commit 3d2574133c6bcd53bb48817eac2046df796d5919
Author: ViqaasTaimoor <viqaastaimoor@gmail.com>
Date:   Fri Dec 26 17:12:32 2025 +0530

    Revert "testing"

    This reverts commit 1e6d21b40bf9d26b4dd7793bb0f7c6a26f950dd0.

Viqaas@LAPTOP-VG025G9U MINGW64 ~/Desktop/Testing/Testing (Polo)
$ git revert HEAD
[Polo 489fb06] Revert "test"
 1 file changed, 0 insertions(+), 0 deletions(-)
 delete mode 100644 b
```

- Detached HEAD Challenge:

# Git and GitHub Challenges

- Checkout a specific commit (detached HEAD state) and create a new branch from it.

```
Viqaas@LAPTOP-VG025G9U MINGW64 ~/Desktop/Testing/Testing (Polo)
$ git checkout 077de34
Note: switching to '077de34'.

You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
state without impacting any branches by switching back to a branch.

If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -c with the switch command. Example:

    git switch -c <new-branch-name>

Or undo this operation with:

    git switch -

Turn off this advice by setting config variable advice.detachedHead to
false

HEAD is now at 077de34 test

Viqaas@LAPTOP-VG025G9U MINGW64 ~/Desktop/Testing/Testing ((077de34...))
$ git checkotu -b fix
git: 'checkotu' is not a git command. See 'git --help'.

The most similar command is
    checkout

Viqaas@LAPTOP-VG025G9U MINGW64 ~/Desktop/Testing/Testing ((077de34...))
$ git checkout -b fix
Switched to a new branch 'fix'

Viqaas@LAPTOP-VG025G9U MINGW64 ~/Desktop/Testing/Testing (fix)
$ git branch --list
  Marco
  Polo
* fix
  main
```

# Git and GitHub Challenges

- Configure a `pre-commit` hook to reject commits without a message format (e.g., must start with `JIRA-XXX`).

```
Viqaas@LAPTOP-VG025G9U MINGW64 ~/Desktop/Testing/Testing (main)
$ git commit -m "fix login"
X Commit rejected!
Message must start with JIRA-XXX
Example: JIRA-123 Fix login issue

Viqaas@LAPTOP-VG025G9U MINGW64 ~/Desktop/Testing/Testing (main)
$ git commit -m "JIRA-123 fix login"
[main d4b3fe2] JIRA-123 fix login
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 man

Viqaas@LAPTOP-VG025G9U MINGW64 ~/Desktop/Testing/Testing (main)
$ git log -1
commit d4b3fe2537c54c1fe19c094a9a648cd5d82afdcc (HEAD -> main)
Author: ViqaasTaimoor <viqaastaimoor@gmail.com>
Date: Mon Dec 29 11:25:47 2025 +0530

    JIRA-123 fix login
```

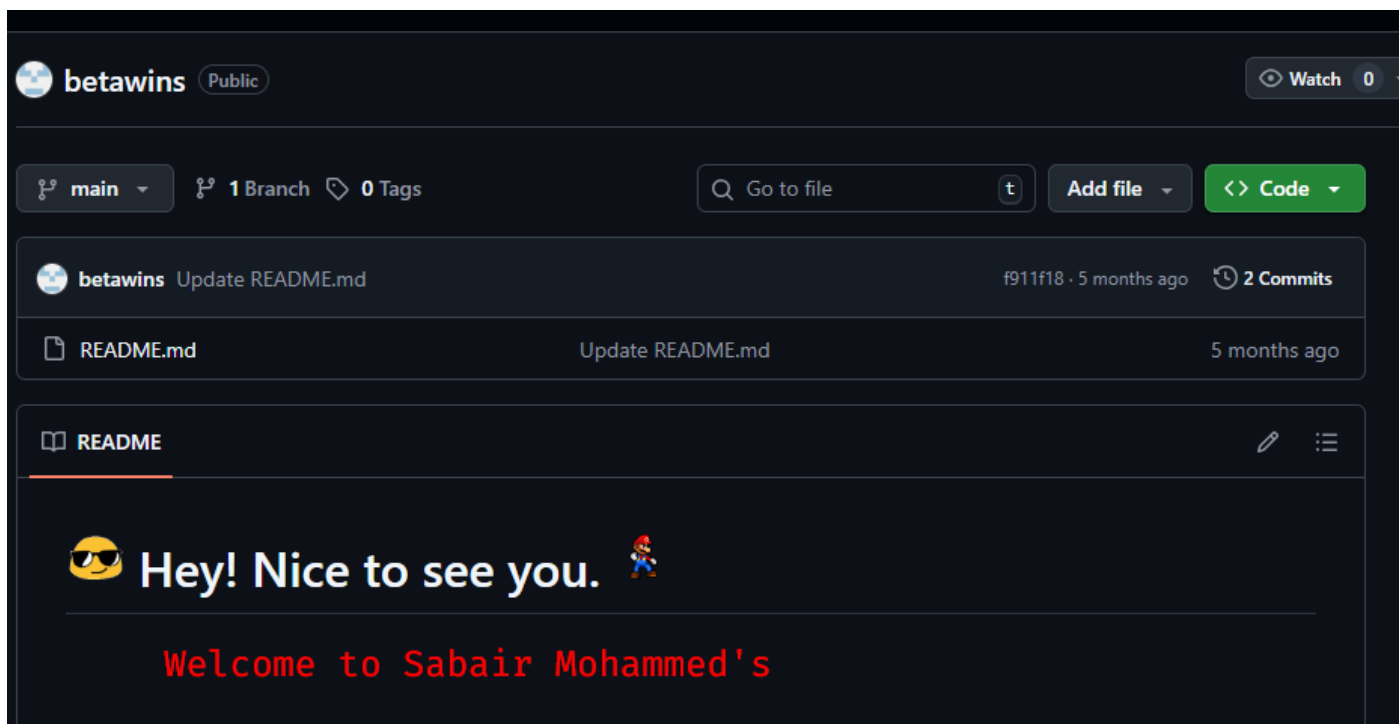
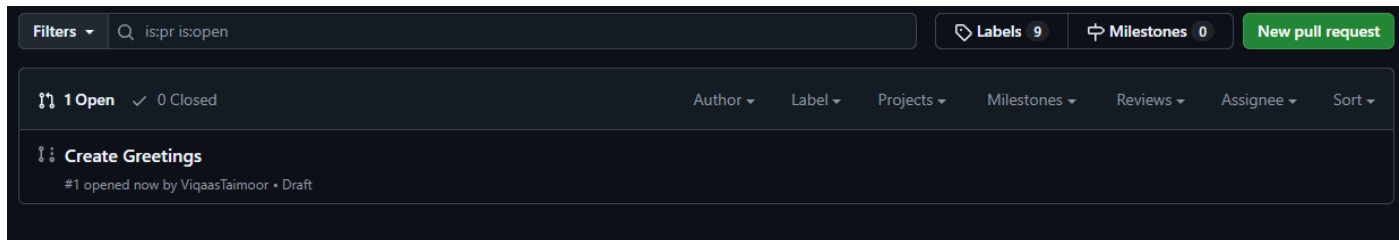
- **Squash Merge vs Rebase Merge:**
- Show the difference between squash merge and rebase merge with evidence.

```
Viqaas@LAPTOP-VG025G9U MINGW64 ~/Desktop/Testing/Testing (fix)
$ git rebase --continue
[detached HEAD 42fe048] modified
2 files changed, 0 insertions(+), 0 deletions(-)
create mode 100644 a
create mode 100644 pat
dropping 9929b1395a5ddd6acd8c69ce5e00f76e0bffd25e added fighter -- patch
s already upstream
dropping 358e03d6d07a60e6e789875c517306df8e74f376 added files to test --
contents already upstream
Successfully rebased and updated refs/heads/fix.

Viqaas@LAPTOP-VG025G9U MINGW64 ~/Desktop/Testing/Testing (fix)
$ git log --oneline --graph
* 5a65780 (HEAD -> fix) test
* 007e20f Revert "testing"
* f3140ca testing
* 42fe048 modified
* 5f72b3d (origin/main, origin/HEAD, main) Add gamma
* b152123 Add beta
* d1910f7 Add alpha
```

# Git and GitHub Challenges

- Fork a repo, make a change, and submit a pull request to the original repo.



- Commit something, reset hard, and then recover it using git reflog.

```
Viqaas@LAPTOP-VG025G9U MINGW64 ~/Desktop/Testing/Testing (fix)
$ git reset --hard 43d9967
HEAD is now at 43d9967 Modification

Viqaas@LAPTOP-VG025G9U MINGW64 ~/Desktop/Testing/Testing (fix)
$ git log -3
commit 43d9967fe2e2e2535e8ab9d99d40b5959e1d95ae (HEAD -> fix)
Author: VikaasTaimoor <viqaastaimoor@gmail.com>
Date: Mon Dec 29 14:28:56 2025 +0530

    Modification
```