

- Created a repo named “git-exam-repo”
- Added branch protection rules

Branch name pattern *

main

Applies to 1 branch

main

Protect matching branches

☒ **Require a pull request before merging**
When enabled, all commits must be made to a non-protected branch and submitted via a pull request before they can be merged into a branch that matches this rule.

☒ **Require approvals**
When enabled, pull requests targeting a matching branch require a number of approvals and no changes requested before they can be merged.
Required number of approvals before merging: 1

☒ **Dismiss stale pull request approvals when new commits are pushed**
New reviewable commits pushed to a matching branch will dismiss pull request review approvals.

☐ **Require review from Code Owners**
Require an approved review in pull requests including files with a designated code owner.

☐ **Require approval of the most recent reviewable push**
Whether the most recent reviewable push must be approved by someone other than the person who pushed it.

☐ **Require status checks to pass before merging**
Choose which **status checks** must pass before branches can be merged into a branch that matches this rule. When enabled, commits must first be pushed to another branch, then merged or pushed directly to a branch that matches this rule after status checks have passed.

☐ **Require conversation resolution before merging**
When enabled, all conversations on code must be resolved before a pull request can be merged into a branch that matches this rule.
[Learn more](#).

☐ **Require signed commits**
Commits pushed to matching branches must have verified signatures.






☐ **Require linear history**
Prevent merge commits from being pushed to matching branches.

☐ **Require deployments to succeed before merging**
Choose which environments must be successfully deployed to before branches can be merged into a branch that matches this rule.

☒ **Lock branch**
Branch is read-only. Users cannot push to the branch.

☐ **Do not allow bypassing the above settings**
The above settings will apply to administrators and custom roles with the "bypass branch protections" permission.

- Added contributors

<input type="text" value="Find a collaborator..."/>			
<input type="checkbox"/>	 Antu Acharjee antu-acharjee • Collaborator		Remove
<input type="checkbox"/>	 Md Shihab Uddin mdshihabuddin10 • Collaborator		Remove
<input type="checkbox"/>	 Ujjal K. Saha Awaiting uksaha77's response	Pending Invite 	Remove
<input type="checkbox"/>	 Umme Habiba umme-habiba328 • Collaborator		Remove

- Here's the command i used to clone, checkout and add task one to readme.md file and push to member1 branch

```
git clone git@github.com:nahtyan-DevOps/git-exam-repo.git
cd git-exam-repo/
ls
git status
git checkout -b member1
git status
git add .
git commit -m "readme file task one added" -m "as per instruction, readme file has been updated" -m "00-30083"
git push
git push --set-upstream origin member1
git status
git add .
git commit --amend --no-edit
git push
```

- And this is the command i used to merge each branches and push to the main branch

```
git status
git branch -r
git merge origin/member2
git add .
git commit -m "task 2 merged with main branch"
git push
git pull
git branch -r
git merge origin/member-3
git add .
git commit -m "task 3 added"
git push
git pull
git branch -r
git merge origin/member4-branch
git add .
git commit -m "task 4 added"
git push
```

Team format:

Nahiyen Mubashshir Group Lead, Member 1

Antu Acharjee Member 2

Md. Shihab Uddin Member 3

Umme Habiba Member 4