****

**Git Configuration**

git config --global user.name "My Name"

- set user name

git config --global user.email "example@gmail.com"

- set user email

git config --list

- list of all configuration

**Set remote**

git remote add origin https://github.com/username/repository.git

- Adds a remote repository.

- origin = change name

git remote set-url origin https://github.com/user/repo2.git

- Change the 'origin' remote's URL

git remote -v

- to verify remote

**Connect local directory to git repositorie**

git init

- new folder add to Initializes a new Git repository.

git add file.txt

- Stages changes in a file for commit.

git add .

- Stages all modified files in the current directory and subdirectories for commit.

git commit -m "message"

- Commits the staged changes with a message.

git commit -a -m "message"

- direct commit without traked file add in staging area

- skipe staging area

git push origin main

- Pushes local commits to the remote repository.

git push -u origin main

- if add -u one time to futcher not again type origin main direct type "git push"

The error you're seeing occurs because the remote main branch on GitHub contains commits that you don't have locally. Git is rejecting your push to avoid overwriting remote changes.

To resolve this, you need to pull the remote changes and merge them with your local repository:

1. Pull the changes from the remote main branch:

**git pull origin main**

If there are any merge conflicts, Git will notify you, and you'll need to resolve them manually.

1. Once the pull is successful (and conflicts are resolved), push your local changes to the remote main branch:

**git push origin main**

If you're not concerned about the changes on the remote and want to overwrite the remote with your local changes, you can force push (be cautious with this):

**git push origin main --force**

Let me know how it goes or if you need further assistance!

**Git Branch**

git branch

- to check branch list

git branch -M main

- to rename branch

git branch new-branch-name

- create a new branch.

git checkout -b other-branch-name

- Switches to a different branch.

git checkout -b other-branch-name

- if avalible to swithc if not avalible to create new branch and then switch

git branch -d branch-name

- delete branch