

VCS Task

Task Description:

Establish a new directory, populate it with script files, initiate an empty repository on GitHub, convert the local directory into a Git repository, and link it to GitHub for pushing the code into the repository. Perform merge, rebase, stash commands in following github repo.

New directory creation with script files:

```
vineeth@vineeth-pc:~$ cd Dev/
vineeth@vineeth-pc:~/Dev$ ls
aws-ci-cd-demo  devops-playground  spiky-gold-practice
vineeth@vineeth-pc:~/Dev$ mkdir vcs-task
vineeth@vineeth-pc:~/Dev$ cd vcs-task/
vineeth@vineeth-pc:~/Dev/vcs-task$ vim script1.sh
vineeth@vineeth-pc:~/Dev/vcs-task$ vim script2.sh
vineeth@vineeth-pc:~/Dev/vcs-task$ ls -la
total 16
drwxrwxr-x 2 vineeth vineeth 4096 Jul  2 10:51 .
drwxrwxr-x 6 vineeth vineeth 4096 Jul  2 10:50 ..
-rw-rw-r-- 1 vineeth vineeth 166 Jul  2 10:50 script1.sh
-rw-rw-r-- 1 vineeth vineeth  83 Jul  2 10:51 script2.sh
```

Git Hub repo creation:

The screenshot shows a GitHub repository named 'vcs-task' by user 'rtvin0618'. The repository is private and has a 'main' branch. The commit history shows two commits, both adding 'script1.sh' and 'script2.sh'. The repository has no description, website, or topics provided. The 'About' section is empty. The 'Releases' section shows no releases published. The 'Packages' section shows no packages published. The 'README' section is empty, with a prompt to 'Add a README'.

Initialize Local Git Repo & Connect to GitHub:

```
vineeth@vineeth-pc:~/Dev/vcs-task$ git init
Initialized empty Git repository in /home/vineeth/Dev/vcs-task/.git/
vineeth@vineeth-pc:~/Dev/vcs-task$ ls -la
total 20
drwxrwxr-x 3 vineeth vineeth 4096 Jul  2 10:57 .
drwxrwxr-x 6 vineeth vineeth 4096 Jul  2 10:50 ..
drwxrwxr-x 7 vineeth vineeth 4096 Jul  2 10:57 .git
-rwxrwxr-x 1 vineeth vineeth 166 Jul  2 10:50 script1.sh
-rwxrwxr-x 1 vineeth vineeth  83 Jul  2 10:51 script2.sh
vineeth@vineeth-pc:~/Dev/vcs-task$ git add .
vineeth@vineeth-pc:~/Dev/vcs-task$ git status
On branch main

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   script1.sh
    new file:   script2.sh

vineeth@vineeth-pc:~/Dev/vcs-task$ git commit -m "Add two file script1.sh and script2.sh"
[main (root-commit) 6e8baf9] Add two file script1.sh and script2.sh
 2 files changed, 16 insertions(+)
 create mode 100755 script1.sh
 create mode 100755 script2.sh
vineeth@vineeth-pc:~/Dev/vcs-task$ git branch -M main
vineeth@vineeth-pc:~/Dev/vcs-task$ git remote add origin https://github.com/ritvin0618/vcs-task
vineeth@vineeth-pc:~/Dev/vcs-task$ git push -u origin main
Username for 'https://github.com': ritvin0618
Password for 'https://ritvin0618@github.com':
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 8 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 441 bytes | 441.00 KiB/s, done.
Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/ritvin0618/vcs-task
 * [new branch]      main -> main
Branch 'main' set up to track remote branch 'main' from 'origin'.
vineeth@vineeth-pc:~/Dev/vcs-task$
```

Create a New Branch and merge to main branch:

```
vineeth@vineeth-pc:~/Dev/vcs-task$ git status
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean
vineeth@vineeth-pc:~/Dev/vcs-task$ git checkout -b feature-branch
Switched to a new branch 'feature-branch'
vineeth@vineeth-pc:~/Dev/vcs-task$ git status
On branch feature-branch
nothing to commit, working tree clean
vineeth@vineeth-pc:~/Dev/vcs-task$ git add script1.sh
vineeth@vineeth-pc:~/Dev/vcs-task$ git commit -m "Modified script1 in feature branch"
On branch feature-branch
nothing to commit, working tree clean
vineeth@vineeth-pc:~/Dev/vcs-task$ vim script1.sh
vineeth@vineeth-pc:~/Dev/vcs-task$ git status
On branch feature-branch
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    modified:   script1.sh

no changes added to commit (use "git add" and/or "git commit -a")
vineeth@vineeth-pc:~/Dev/vcs-task$ git add script1.sh
vineeth@vineeth-pc:~/Dev/vcs-task$ git commit -m "Modified script1 in feature branch"
[feature-branch 69c3e0] Modified script1 in feature branch
 1 file changed, 1 insertion(+)
vineeth@vineeth-pc:~/Dev/vcs-task$ git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.
vineeth@vineeth-pc:~/Dev/vcs-task$ git merge feature-branch
Updating 6e8baf9..69c3e0
Fast-forward
 script1.sh | 1 +
 1 file changed, 1 insertion(+)
vineeth@vineeth-pc:~/Dev/vcs-task$
```

Create New Branch for Rebase Switch back and make a new commit on main:

```
vineeth@vineeth-pc:~/Dev/vcs-task$ git status
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean
vineeth@vineeth-pc:~/Dev/vcs-task$ git checkout -b rebase-branch
Switched to a new branch 'rebase-branch'
vineeth@vineeth-pc:~/Dev/vcs-task$ vim script2.sh
vineeth@vineeth-pc:~/Dev/vcs-task$ git status
On branch rebase-branch
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   script2.sh

no changes added to commit (use "git add" and/or "git commit -a")
vineeth@vineeth-pc:~/Dev/vcs-task$ git add script2.sh
vineeth@vineeth-pc:~/Dev/vcs-task$ git commit -m "Rebase branch changes"
[rebase-branch 9474ba7] Rebase branch changes
 1 file changed, 1 insertion(+)
vineeth@vineeth-pc:~/Dev/vcs-task$ git checkout main
Switched to branch 'main'
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)
vineeth@vineeth-pc:~/Dev/vcs-task$ vim script2.sh
vineeth@vineeth-pc:~/Dev/vcs-task$ git add script2.sh
vineeth@vineeth-pc:~/Dev/vcs-task$ git commit -m "Main branch updated"
[main bdc6ea1] Main branch updated
 1 file changed, 1 insertion(+)
vineeth@vineeth-pc:~/Dev/vcs-task$ git checkout rebase-branch
Switched to branch 'rebase-branch'
vineeth@vineeth-pc:~/Dev/vcs-task$ git rebase main
Auto-merging script2.sh
CONFLICT (content): Merge conflict in script2.sh
error: could not apply 9474ba7... Rebase branch changes
hint: Resolve all conflicts manually, mark them as resolved with
hint: "git add/rm <conflicted_files>", then run "git rebase --continue".
hint: You can instead skip this commit: run "git rebase --skip".
hint: To abort and get back to the state before "git rebase", run "git rebase --abort".
Could not apply 9474ba7... Rebase branch changes
vineeth@vineeth-pc:~/Dev/vcs-task$ vim script2.sh
vineeth@vineeth-pc:~/Dev/vcs-task$ git add script2.sh
vineeth@vineeth-pc:~/Dev/vcs-task$ git rebase --continue
[detached HEAD ff0e366] Rebase branch changes
 1 file changed, 4 insertions(+)
Successfully rebased and updated refs/heads/rebase-branch.
vineeth@vineeth-pc:~/Dev/vcs-task$
```

Use Git Stash:

```
vineeth@vineeth-pc:~/Dev/vcs-task$ git checkout mian
error: pathspec 'mian' did not match any file(s) known to git
vineeth@vineeth-pc:~/Dev/vcs-task$ git checkout main
Switched to branch 'main'
Your branch is ahead of 'origin/main' by 2 commits.
  (use "git push" to publish your local commits)
vineeth@vineeth-pc:~/Dev/vcs-task$ vim script1.sh
vineeth@vineeth-pc:~/Dev/vcs-task$ git add script1.sh
vineeth@vineeth-pc:~/Dev/vcs-task$ git stash
Saved working directory and index state WIP on main: bdc6ea1 Main branch updated
vineeth@vineeth-pc:~/Dev/vcs-task$ git stash list
stash@{0}: WIP on main: bdc6ea1 Main branch updated
vineeth@vineeth-pc:~/Dev/vcs-task$ git stash pop
On branch main
Your branch is ahead of 'origin/main' by 2 commits.
  (use "git push" to publish your local commits)

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   script1.sh

no changes added to commit (use "git add" and/or "git commit -a")
Dropped refs/stash@{0} (c44b09462d03d300e807dcf25716ba86b740a611)
vineeth@vineeth-pc:~/Dev/vcs-task$
```