

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.

STEP 1 - Establish a new directory

```

, #
~ \ ###_ Amazon Linux 2023
~~ \ #####\
~~ \ ###|
~~ \ #/ https://aws.amazon.com/linux/amazon-linux-2023
~~ V~' '->
~~~
~~ . .
~~ / /
~~ /m/ ' '->
Last login: Fri Sep 6 08:52:38 2024 from 13.239.158.5
[ec2-user@ip-172-31-39-229 ~]$ sudo su
[root@ip-172-31-39-229 ec2-user]# mkdir task5
[root@ip-172-31-39-229 ec2-user]# cd task5
[root@ip-172-31-39-229 task5]# pwd
/home/ec2-user/task5

```

- **Sudo su** command for root user
- **Mkdir** command for folder create
- **Cd** command for open folder
- **Pwd** command for present work directory

```
[root@ip-172-31-39-229 task5]# yum install git
Last metadata expiration check: 0:02:33 ago on Mon Sep  9 18:43:37 2024.
Dependencies resolved.
```

Package	Architecture	Version
Installing:		
git	x86_64	2.40.1-1.amzn2023.0.3
Installing dependencies:		
git-core	x86_64	2.40.1-1.amzn2023.0.3
git-core-doc	noarch	2.40.1-1.amzn2023.0.3
perl-Error	noarch	1:0.17029-5.amzn2023.0.2
perl-File-Find	noarch	1.37-477.amzn2023.0.6
perl-Git	noarch	2.40.1-1.amzn2023.0.3
perl-TermReadKey	x86_64	2.38-9.amzn2023.0.2
perl-lib	x86_64	0.65-477.amzn2023.0.6

- **Git -v** command for check git installed or not
- **Yum** command for install commands

STEP 2 - convert the local directory into a Git repository & populate it with script files

```
[root@ip-172-31-39-229 task5]# git init
hint: Using 'master' as the name for the initial branch. This default branch name
hint: is subject to change. To configure the initial branch name to use in all
hint: of your new repositories, which will suppress this warning, call:
hint:
hint:   git config --global init.defaultBranch <name>
hint:
hint: Names commonly chosen instead of 'master' are 'main', 'trunk' and
hint: 'development'. The just-created branch can be renamed via this command:
hint:
hint:   git branch -m <name>
Initialized empty Git repository in /home/ec2-user/task5/.git/
[root@ip-172-31-39-229 task5]# ls -la
total 0
drwxr-xr-x. 3 root    root      18 Sep  9 18:49 .
drwx----- 7 ec2-user ec2-user 156 Sep  9 18:44 ..
drwxr-xr-x. 7 root    root      119 Sep  9 18:49 .git
[root@ip-172-31-39-229 task5]# touch github.py
[root@ip-172-31-39-229 task5]# vim github.py
```

- **Git init** command for create local repo
- **Ls -la** command for check git repo created
- **Touch** command for create a file
- **Vim** command for write content inside it

```
[root@ip-172-31-39-229 task5]# git add github.py
[root@ip-172-31-39-229 task5]# git commit -m "the first code written"
[master (root-commit) bd8c192] the first code written
Committer: root <root@ip-172-31-39-229.ap-southeast-2.compute.internal>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:

    git config --global --edit

After doing this, you may fix the identity used for this commit with:

    git commit --amend --reset-author

1 file changed, 9 insertions(+)
create mode 100755 github.py
```

- **Git add** command for add a file to staging area
- **Git commit -m "some text"** command for move the file to local repo

```

[root@ip-172-31-39-229 task5]# git status
On branch master
nothing to commit, working tree clean
[root@ip-172-31-39-229 task5]# git branch
* master
[root@ip-172-31-39-229 task5]# ls -l
total 4
-rwxr-xr-x. 1 root root 314 Sep  9 18:51 github.py
[root@ip-172-31-39-229 task5]# git branch guvi
[root@ip-172-31-39-229 task5]# git checkout guvi
Switched to branch 'guvi'
[root@ip-172-31-39-229 task5]# git branch
* guvi
  master

```

- **Git status** command to check the commit or uncommit changes
- **Git branch** command for create a branch
- **Git checkout** command for switch btw branches

```

-rwxr-xr-x. 1 root root 314 Sep  9 18:51 github.py
[root@ip-172-31-39-229 task5]# vim github.py
[root@ip-172-31-39-229 task5]# git add github.py
[root@ip-172-31-39-229 task5]# git commit -m "hello world to team"
[guvi 4b129db] hello world to team
Committer: root <root@ip-172-31-39-229.ap-southeast-2.compute.internal>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:

    git config --global --edit

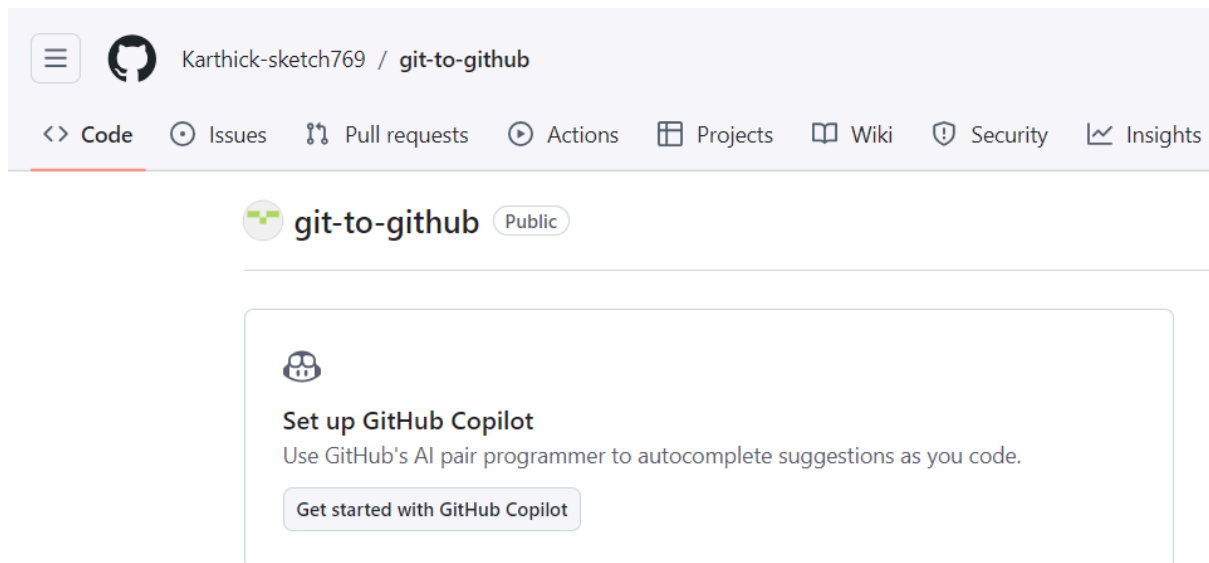
After doing this, you may fix the identity used for this commit with:

    git commit --amend --reset-author

1 file changed, 2 insertions(+), 2 deletions(-)
[root@ip-172-31-39-229 task5]# git status
On branch guvi
nothing to commit, working tree clean

```

STEP 3 - initiate an empty repository on GitHub



How to create github repo/remote repository?

Install github > create an account > create public repository > settings > develop.settings > token > generate classic token
now ,

STEP 4 - link it to GitHub for pushing the code into the repository



Git remote set-url origin

[http://token github.com/Karthick-sketch769/chennai.git](http://token.github.com/Karthick-sketch769/chennai.git)

Git branch -m main create a branch in github

Git push -u origin main push git main file to git repo

```
[root@ip-172-31-39-229 task5]# git remote set-url origin https://ghp_FYUIcnt0B6UYscY129W2.git
[root@ip-172-31-39-229 task5]# git push origin master
Enumerating objects: 15, done.
Counting objects: 100% (15/15), done.
Compressing objects: 100% (10/10), done.
Writing objects: 100% (15/15), 1.41 KiB | 722.00 KiB/s, done.
Total 15 (delta 6), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (6/6), done.
To https://github.com/Karthick-sketch769/git-to-github.git
* [new branch]      master -> master
```

STEP 5 - Perform merge, rebase, stash commands in following github repo.

Merge & rebase both are same

Before merged - master branch

```
// This is a simple Java program that prints "Hello, World!" to the console

public class HelloWorld {

    // The main method is the entry point of any Java application
    public static void main(String[] args) {
        // Print "Hello, World!" to the console
        System.out.println("Hello, World!");
    }
}
```

Guvi branch

```
// This is a simple Java program that prints "Hello, team!" to the console

public class Hello team {

    // The main method is the entry point of any Java application
    public static void main(String[] args) {
        // Print "Hello, team!" to the console
        System.out.println("Hello, team!");
    }
}
```

- **Example** if there is 2 branch(guvi,main/master) if you want merge from one to another then you sit in main then use git merge guvi command to merge guvi files in main folder

Merge/rebase conflicts

```
<<<<<< HEAD
// This is a simple Java program that prints "Hello,world!" to the console
=====
// This is a simple Java program that prints "Hello, team!" to the console
>>>>>> guvi

public class Hello team {

    // The main method is the entry point of any Java application
    public static void main(String[] args) {
        // Print "Hello, world!" to the console
        System.out.println("Hello,world!");
    }
}
```

- If 2 persons making changes in same line of a same file from different branches then it may cause conflict when merging
- So the conflict error represent in <<<<<< **HEAD** =====>>>>> **GUVI** so erase this conflict and try again to merge then this conflict solved

Merged - master to guvi

```
// This is a simple Java program that prints "Hello,team!" to the console

public class Hello team {

    // The main method is the entry point of any Java application
    public static void main(String[] args) {
        // Print "Hello, team!" to the console
        System.out.println("Hello,team!");
    }
}
```

stash commands

```
[root@ip-172-31-39-229 pen]# git stash
Saved working directory and index state WIP on main: d633b67 karthick
```

Git stash command is used for keep a file as temporary without commit move another branch then to commit this file then u need to come back to existing branch and use **git unstash <stash id>** then do commit process

```
switched to branch 'master'
[root@ip-172-31-39-229 movie]# git stash apply
error: Your local changes to the following files would be overwritten by merge:
      thala.txt
Please commit your changes or stash them before you merge.
Aborting
On branch master
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:   thala.txt

no changes added to commit (use "git add" and/or "git commit -a")
```

Git stash apply <stash id > to unstash and do commit process