

GIT

Assignment – 1

Task to be performed

1. Based on what you have learnt in the class, do the following steps:
 - a) Create a new folder
 - b) Put the following files in the folder
 - Code.txt
 - Log.txt
 - Output.txt
 - c) Stage the code.txt and output.txt files
 - d) Commit them
 - e) And finally push them to GitHub
2. Please share the commands for the above points

Steps and Commands

- Create a directory in EC2 instance by running the command **mkdir (directory name)**
- Get into the directory by running the command **cd (directory name)**
- To Initializing the git by running the command **git init**
- To create the files by running the command **touch code.txt, touch log.txt and touch output.txt**
- To stage code.txt and log txt files by running the command **git add code.txt and git add output.txt**
- And to check the files which are in status by running the command **git status**
- To push the files to commit stage by running command **git commit -m "necessary notes"**
- Go Inside the **GITHUB** account
 - a) Click the New in the top repositories
 - b) Provide the name for repository
 - c) Finally click the create repository
- Next to link the repository with local machine by running the command **git remote add origin (repository URL link)**
- To push the files into the repository by running the command **git push origin master**
- Provide the username and password token which was assigned by github
- Now the files which was created in EC2 local machine will be reflected in Github repository

```

ubuntu@ip-172-31-85-101:~$ sudo apt-get update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease [109 kB]
Get:4 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 Packages [14.1 MB]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe Translation-en [5652 kB]
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 c-n-f Metadata [286 kB]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse amd64 Packages [217 kB]
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse Translation-en [112 kB]
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse amd64 c-n-f Metadata [8372 B]
Get:11 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [1458 kB]
Get:12 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main Translation-en [283 kB]
Get:13 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [1557 kB]
Get:14 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted Translation-en [259 kB]
Get:15 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [1054 kB]
Get:16 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe Translation-en [238 kB]
Get:17 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 c-n-f Metadata [22.1 kB]
Get:18 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 Packages [42.1 kB]
Get:19 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/multiverse Translation-en [10.1 kB]
Get:20 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 c-n-f Metadata [472 B]
Get:21 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/main amd64 Packages [41.7 kB]
Get:22 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/main Translation-en [10.5 kB]
Get:23 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/main amd64 c-n-f Metadata [388 B]
Get:24 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/restricted amd64 c-n-f Metadata [116 B]

```

```

ubuntu@ip-172-31-85-101:~$ mkdir task
ubuntu@ip-172-31-85-101:~$ cd task
ubuntu@ip-172-31-85-101:~/task$ 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/ubuntu/task/.git/
ubuntu@ip-172-31-85-101:~/task$

```

```

ubuntu@ip-172-31-85-101:~/task$ nano code.txt
ubuntu@ip-172-31-85-101:~/task$ nano log.txt
ubuntu@ip-172-31-85-101:~/task$ nano output.txt
ubuntu@ip-172-31-85-101:~/task$ git add code.txt
ubuntu@ip-172-31-85-101:~/task$ git add output.txt
ubuntu@ip-172-31-85-101:~/task$ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   code.txt
        new file:   output.txt

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        log.txt

ubuntu@ip-172-31-85-101:~/task$ git commit -m "this is my first commit"

```

```

        new file:   output.txt

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        log.txt

ubuntu@ip-172-31-85-101:~/task$ git commit -m "this is my first commit"
[master (root-commit) b64d783] this is my first commit
  Committer: Ubuntu <ubuntu@ip-172-31-85-101.ec2.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

2 files changed, 2 insertions(+)
create mode 100644 code.txt
create mode 100644 output.txt
ubuntu@ip-172-31-85-101:~/task$

```

```
ubuntu@ip-172-31-85-101:~/task$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        log.txt

nothing added to commit but untracked files present (use "git add" to track)
ubuntu@ip-172-31-85-101:~/task$
```

New repository

Q Type / to search

>-

+

🔄

🔍

📧

👤

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository](#).

Required fields are marked with an asterisk (*).

Owner *

task007007

/

Repository name *

task

task is available.

Great repository names are short and memorable. Need inspiration? How about [didactic-goggles](#) ?

Description (optional)

☒ Public

Anyone on the internet can see this repository. You choose who can commit.

☐ Private

You choose who can see and commit to this repository.

task

Public

Pin

Unwatch 1

Fork 0

Star 0

Set up GitHub Copilot

Use GitHub's AI pair programmer to autocomplete suggestions as you code.

Get started with GitHub Copilot

Add collaborators to this repository

Search for people using their GitHub username or email address.

Invite collaborators

Quick setup — if you've done this kind of thing before



📄 Set up in Desktop or HTTPS SSH https://github.com/task007007/task.git







Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).










...or create a new repository on the command line






echo "# task" >> README.md
git init
git add README.md
git commit -m "first commit"




```
ubuntu@ip-172-31-85-101:~/task$ ls
code.txt  log.txt  output.txt
ubuntu@ip-172-31-85-101:~/task$ git remote add origin https://github.com/task007007/task.git
ubuntu@ip-172-31-85-101:~/task$ git push origin master
Username for 'https://github.com': task007007
Password for 'https://task007007@github.com':
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (4/4), 304 bytes | 304.00 KiB/s, done.
Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/task007007/task.git
 * [new branch]      master -> master
ubuntu@ip-172-31-85-101:~/task$
```


 task007007 / task



Q Type  to search     


 Code  Issues  Pull requests  Actions  Projects  Wiki  Security  Insights  Settings



 **task** Public  Pin  Unwatch 1  Fork 0  Star 0


 master  1 Branch  0 Tags




 Add file  Code

 **Ubuntu** this is my first commit b64d783 · 20 minutes ago 1 Commits

 code.txt	this is my first commit	20 minutes ago
 output.txt	this is my first commit	20 minutes ago


 README





Add a README


About

No description, website, or topics provided.

 Activity

 0 stars

 1 watching

 0 forks

Releases

No releases published

[Create a new release](#)

Packages

Assignment – 2

Task to be performed

1. Create a Git working directory with feature1.txt and feature2.txt in the master branch
2. Create 3 branches develop, feature1 and feature 2
3. In develop branch create develop.txt do not stage or commit it
4. Stash this file and check out to feature1 branch
5. Create new.txt file in feature1 branch, stage and commit this file
6. Checkout to develop, unstash this file and commit
7. Please submit all the git commands used to do the above steps

Steps and Commands

- Create a directory in EC2 instance by running the command **mkdir (directory name)**
- Get into the directory by running the command **cd (directory name)**
- To Initializing the git by running the command **git init**
- To create the files by running the command **touch feature1.txt and touch feature2.txt**
- To stage feature1.txt and feature2.txt files by running the command **git add feature1.txt and add feature2.txt**
- And to check the files which are in which stage by running the command **git status**
- To push the files to commit stage by running command **git commit –m “necessary notes”**
- To create new branches by running command **git branch develop, git branch feature1 and git branch feature2**
- To check the list of branch by running command **git branch**
- Next to switch from master branch to develop branch by running the command **git checkout develop**
- If we run the command **ls** in develop branch the files which was created in master branch will be reflected
- To create the files by running the command **touch develop.txt**
- If we run the **git status** command the created develop.txt file will in the untracked stage
- To Stash the file by running the command **git stash -u**
- If we run the **ls** command to see list files the new.txt file will not be reflected in the list since it was stashed
- Next to switch from develop branch to feature1 branch by running the command **git checkout feature1**
- To create the files by running the command **touch new.txt**

- Next to stage and commit the file by running combined command **git add . && git commit -m "Add necessary notes"**
- Next to switch from feature1 branch to develop branch by running the command **git checkout develop**
- To unstash the new.txt file by running the command **git stash pop**
- Next to stage and commit the file by running single command **git add . && git commit -m "Add necessary notes"**

```
ubuntu@ip-172-31-85-101:~$ mkdir assign2
ubuntu@ip-172-31-85-101:~$ cd assign2
ubuntu@ip-172-31-85-101:~/assign2$ 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/ubuntu/assign2/.git/
ubuntu@ip-172-31-85-101:~/assign2$
```

```
ubuntu@ip-172-31-85-101:~/assign2$ touch master.txt feature1.txt feature2.txt
ubuntu@ip-172-31-85-101:~/assign2$ git add .
ubuntu@ip-172-31-85-101:~/assign2$ git commit -m "this is my commit"
```

```
ubuntu@ip-172-31-85-101:~/assign2$ git commit -m "this is my commit"
[master (root-commit) 3c1ba32] this is my commit
Committer: Ubuntu <ubuntu@ip-172-31-85-101.ec2.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

3 files changed, 0 insertions(+), 0 deletions(-)
create mode 100644 feature1.txt
create mode 100644 feature2.txt
create mode 100644 master.txt
ubuntu@ip-172-31-85-101:~/assign2$ git branch
* master
ubuntu@ip-172-31-85-101:~/assign2$ git status
On branch master
nothing to commit, working tree clean
ubuntu@ip-172-31-85-101:~/assign2$
```

```
ubuntu@ip-172-31-85-101:~/assign2$ git branch feature1
ubuntu@ip-172-31-85-101:~/assign2$ git branch feature2
ubuntu@ip-172-31-85-101:~/assign2$ git branch develop
```

```
ubuntu@ip-172-31-85-101:~/assign2$ git branch feature1
ubuntu@ip-172-31-85-101:~/assign2$ git branch feature2
ubuntu@ip-172-31-85-101:~/assign2$ git branch
  develop
  feature1
  feature2
* master
ubuntu@ip-172-31-85-101:~/assign2$
```

```
ubuntu@ip-172-31-85-101:~/assign2$ git branch feature1
ubuntu@ip-172-31-85-101:~/assign2$ git branch feature2
ubuntu@ip-172-31-85-101:~/assign2$ git branch
  develop
  feature1
  feature2
* master
ubuntu@ip-172-31-85-101:~/assign2$ git checkout develop
Switched to branch 'develop'
ubuntu@ip-172-31-85-101:~/assign2$ ls
feature1.txt feature2.txt master.txt
ubuntu@ip-172-31-85-101:~/assign2$
```

```
ubuntu@ip-172-31-85-101:~/assign2$ touch develop.txt
ubuntu@ip-172-31-85-101:~/assign2$ ls
develop.txt feature1.txt feature2.txt master.txt
ubuntu@ip-172-31-85-101:~/assign2$ git status
On branch develop
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    develop.txt

nothing added to commit but untracked files present (use "git add" to track)
ubuntu@ip-172-31-85-101:~/assign2$ git stash -u
Saved working directory and index state WIP on develop: 3c1ba32 this is my commit
ubuntu@ip-172-31-85-101:~/assign2$ ls
feature1.txt feature2.txt master.txt
ubuntu@ip-172-31-85-101:~/assign2$
```

```

ubuntu@ip-172-31-85-101:~/assign2$ git checkout feature1
Switched to branch 'feature1'
ubuntu@ip-172-31-85-101:~/assign2$ touch new.txt
ubuntu@ip-172-31-85-101:~/assign2$ git add new.txt
ubuntu@ip-172-31-85-101:~/assign2$ git commit -m "feature1 cmooit"
[feature1 de6c0ff2] feature1 cmooit
  Committer: Ubuntu <ubuntu@ip-172-31-85-101.ec2.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, 0 insertions(+), 0 deletions(-)
create mode 100644 new.txt
ubuntu@ip-172-31-85-101:~/assign2$

```

```

ubuntu@ip-172-31-85-101:~/assign2$ ls
feature1.txt  feature2.txt  master.txt  new.txt
ubuntu@ip-172-31-85-101:~/assign2$ git branch
  develop
* feature1
  feature2
  master
ubuntu@ip-172-31-85-101:~/assign2$ git checkout develop
Switched to branch 'develop'
ubuntu@ip-172-31-85-101:~/assign2$ ls
feature1.txt  feature2.txt  master.txt
ubuntu@ip-172-31-85-101:~/assign2$ git stash pop
Already up to date.
On branch develop
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    develop.txt

nothing added to commit but untracked files present (use "git add" to track)
Dropped refs/stash@{0} (6a5b36d592431b92d310647333ffe227f42e8a81)
ubuntu@ip-172-31-85-101:~/assign2$

```

```

nothing added to commit but untracked files present (use "git add" to track)
Dropped refs/stash@{0} (6a5b36d592431b92d310647333ffe227f42e8a81)
ubuntu@ip-172-31-85-101:~/assign2$ git add develop.txt
ubuntu@ip-172-31-85-101:~/assign2$ git commit -m "finalised changes"
[develop 5b3c8d1] finalised changes
  Committer: Ubuntu <ubuntu@ip-172-31-85-101.ec2.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, 0 insertions(+), 0 deletions(-)
create mode 100644 develop.txt
ubuntu@ip-172-31-85-101:~/assign2$ git status
On branch develop
nothing to commit, working tree clean
ubuntu@ip-172-31-85-101:~/assign2$

```


Assignment – 3

Task to be performed

1. Create a Git working directory, with following branches
Develop
f1
f2
2. In the master branch, commit main.txt file
3. Put develop.txt in develop branch, f1.txt and f2.txt in f1 and f2 respectively
4. Push all these branches to Github
5. On local delete f2 branch
6. Delete the same branch on Github as well

Steps and Commands

- Create a directory in EC2 instance by running the command **mkdir (directory name)**
- Get into the directory by running the command **cd (directory name)**
- To Initializing the git by running the command **git init**
- To create the files by running the command **touch main.txt**
- Next to stage and commit the file by running combined command **git add . && git commit -m "Add necessary notes"**
- To create new branches by running command **git branch develop, git branch f1 and git branch f2**
- Next to switch from master branch to develop branch by running the command **git checkout develop**
- To create the files by running the command **touch develop.txt**
- Next to stage and commit the file by running combined command **git add . && git commit -m "Add necessary notes"**
- Next to switch from develop branch to f1 branch by running the command **git checkout f1**
- To create the files by running the command **touch f1.txt**
- Next to stage and commit the file by running combined command **git add . && git commit -m "Add necessary notes"**
- Next to switch from f1 branch to f2 branch by running the command **git checkout f2**
- To create the files by running the command **touch f2.txt**
- Next to stage and commit the file by running combined command **git add . && git commit -m "Add necessary notes"**

- Go Inside the **GITHUB** account
 - d) Click the New in the top repositories
 - e) Provide the name for repository
 - f) Finally click the create repository
- Next to link the repository with local machine by running the command **git remote add origin (repository URL link)**
- To push all branch into the repository by running the command **git push --all**
- Provide the username and password token which was assigned by github
- Now the files which was created in all branch will be reflected in Github repository
- Next to switch from f2 branch to master branch by running the command **git checkout master**
- To delete the branch from local machine by running the command **git branch -D f2**
- To delete the branch from the Github repository by running command **git push origin --delete f2**
- Provide the username and password token which was assigned by github
- Now the f2 branch is deleted from both local machine as well as form github repository

```
ubuntu@ip-172-31-85-101:~$ mkdir assingment
ubuntu@ip-172-31-85-101:~$ cd assingment
ubuntu@ip-172-31-85-101:~/assingment$ 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/ubuntu/assingment/.git/
ubuntu@ip-172-31-85-101:~/assingment$
```

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

```
ubuntu@ip-172-31-85-101:~/assingment$ touch main.txt
ubuntu@ip-172-31-85-101:~/assingment$ git add main.txt
ubuntu@ip-172-31-85-101:~/assingment$ git commit -m "new assignment"
[master (root-commit) 391c7e7] new assignment
  Committer: Ubuntu <ubuntu@ip-172-31-85-101.ec2.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, 0 insertions(+), 0 deletions(-)
create mode 100644 main.txt
ubuntu@ip-172-31-85-101:~/assingment$
```

```
ubuntu@ip-172-31-85-101:~/assingment$ ls
main.txt
ubuntu@ip-172-31-85-101:~/assingment$ git branch
* master
ubuntu@ip-172-31-85-101:~/assingment$ git branch develop
ubuntu@ip-172-31-85-101:~/assingment$ git branch f1
ubuntu@ip-172-31-85-101:~/assingment$ git branch f2
ubuntu@ip-172-31-85-101:~/assingment$ git checkout develop
Switched to branch 'develop'
ubuntu@ip-172-31-85-101:~/assingment$ touch develop.txt
ubuntu@ip-172-31-85-101:~/assingment$ git add . && git commit -m "develop"
[develop ce98618] develop
Committer: Ubuntu <ubuntu@ip-172-31-85-101.ec2.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
```

```
ubuntu@ip-172-31-85-101:~/assingment$ git checkout f1
Switched to branch 'f1'
ubuntu@ip-172-31-85-101:~/assingment$ touch f1.txt
ubuntu@ip-172-31-85-101:~/assingment$ git add . && git commit -m "f1"
[f1 96a3770] f1
Committer: Ubuntu <ubuntu@ip-172-31-85-101.ec2.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, 0 insertions(+), 0 deletions(-)
create mode 100644 f1.txt
ubuntu@ip-172-31-85-101:~/assingment$
```



```
ubuntu@ip-172-31-85-101:~/assingment$ git checkout f2
Switched to branch 'f2'
ubuntu@ip-172-31-85-101:~/assingment$ touch f2.txt
ubuntu@ip-172-31-85-101:~/assingment$ git add . && git commit -m "f2"
[f2 9234fde] f2
Committer: Ubuntu <ubuntu@ip-172-31-85-101.ec2.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, 0 insertions(+), 0 deletions(-)
create mode 100644 f2.txt
ubuntu@ip-172-31-85-101:~/assingment$
```

 New repository


Q Type  to search      

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository](#).

Required fields are marked with an asterisk (*).


Owner *

 task007007

/


Repository name *

assignment3


 assignment3 is available.

Great repository names are short and memorable. Need inspiration? How about [solid-octo-giggle](#) ?

Description (optional)

☒  Public

Anyone on the internet can see this repository. You choose who can commit.

☐  Private

You choose who can see and commit to this repository.

```
ubuntu@ip-172-31-85-101:~/assingment$ git remote add origin https://github.com/task007007/assignment3.git
```

```
ubuntu@ip-172-31-85-101:~/assingment$ git push --all
Username for 'https://github.com': task007007
Password for 'https://task007007@github.com':
Enumerating objects: 9, done.
Counting objects: 100% (9/9), done.
Compressing objects: 100% (7/7), done.
Writing objects: 100% (9/9), 717 bytes | 358.00 KiB/s, done.
Total 9 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), done.
To https://github.com/task007007/assignment3.git
 * [new branch]      develop -> develop
 * [new branch]      f1 -> f1
 * [new branch]      f2 -> f2
 * [new branch]      master -> master
ubuntu@ip-172-31-85-101:~/assingment$
```

assignment3Public

Pin

Unwatch1

Fork0

Star0

f2 had recent pushes 3 minutes ago

Compare & pull request

f1 had recent pushes 3 minutes ago

Compare & pull request

develop4 Branches0 Tags

Go to file

Add file

Code

Switch branches/tags

Find or create a branch...

BranchesTags

developdefault

f1

f2

master

ce98618 · 22 minutes ago2 Commits

develop	22 minutes ago
new assignment	26 minutes ago

About

No description, website, or topics provided.

Activity

0 stars

1 watching

0 forks

Releases

No releases published

Create a new release

Packages

No packages published

Publish your first package

develop4 Branches0 Tags

Go to file

Add file

Code

Ubuntu develop

ce98618 · 22 minutes ago2 Commits

develop.txt	develop	22 minutes ago
main.txt	new assignment	26 minutes ago

README

Add a README

Help people interested in this repository understand your project by adding a README.

Add a README

Releases

No releases published

Create a new release

Packages

No packages published


Publish your first package

```
ubuntu@ip-172-31-85-101:~/assignment$ git checkout master
Switched to branch 'master'
ubuntu@ip-172-31-85-101:~/assignment$ git branch
  develop
   f1
   f2
* master
ubuntu@ip-172-31-85-101:~/assignment$ git branch -D f2
Deleted branch f2 (was 9234fde).
ubuntu@ip-172-31-85-101:~/assignment$ git branch
  develop
   f1
* master
ubuntu@ip-172-31-85-101:~/assignment$
```


```

ubuntu@ip-172-31-85-101:~/assingment$ git checkout master
Switched to branch 'master'
ubuntu@ip-172-31-85-101:~/assingment$ git branch
  develop
   f1
   f2
* master
ubuntu@ip-172-31-85-101:~/assingment$ git branch -D f2
Deleted branch f2 (was 9234fde).
ubuntu@ip-172-31-85-101:~/assingment$ git branch
  develop
   f1
* master
ubuntu@ip-172-31-85-101:~/assingment$ git push origin --delete f2
Username for 'https://github.com': task007007
Password for 'https://task007007@github.com':
To https://github.com/task007007/assignment3.git
- [deleted]          f2
ubuntu@ip-172-31-85-101:~/assingment$

```


assignment3
Public

Pin
Unwatch 1
Fork 0
Star 0


f1 had recent pushes 14 minutes ago
 Compare & pull request

develop
3 Branches
0 Tags

Branches

Tags

✓ develop default

f1

master

View all branches


⌵

Add file
<> Code

ce98618 · 32 minutes ago

2 Commits

develop	32 minutes ago
new assignment	36 minutes ago



Add a README

Help people interested in this repository understand your project by adding a README.

About

No description, website, or topics provided.

Activity
0 stars
1 watching
0 forks

Releases

No releases published

[Create a new release](#)

Packages

No packages published

[Publish your first package](#)

Assignment – 4

Task to be performed

1. Put master.txt on master branch, stage and commit
2. Create 3 branches public 1, public 2 and private
3. Put public1.txt on public 1 branch, stage and commit
4. Merge public 1 on master branch
5. Merge public 2 on master branch
6. Edit master.txt on private branch, stage and commit
7. Now update branch public 1 and public 2 with new master code in private
8. Also update new master code on master
9. Finally update new master code on private branch

Steps and Commands

- Create a directory in EC2 instance by running the command **mkdir (directory name)**
- Get into the directory by running the command **cd (directory name)**
- To Initializing the git by running the command **git init**
- To create the files by running the command **touch master.txt**
- Next to stage and commit the file by running combined command **git add . && git commit -m "Add necessary notes"**
- To create new branches by running command **git branch public1, git branch public2 and git private**
- Next to switch from master branch to public1 branch by running the command **git checkout public1**
- To create the files by running the command **touch public1.txt**
- Next to stage and commit the file by running combined command **git add . && git commit -m "Add necessary notes"**
- Next to switch from public1 branch to public2 branch by running the command **git checkout public2**
- To create the files by running the command **touch public2.txt**
- Next to stage and commit the file by running combined command **git add . && git commit -m "Add necessary notes"**
- Next to switch from public2 branch to master branch by running the command **git checkout master**
- To merge the master and public1 two branches by running the command **git merge public1**
- To merge the master and public2 two branches by running the command **git merge public2**
- Next to switch from master branch to private branch by running the command **git checkout private**

- To edit the master.txt in private branch by running command **nano master.txt**
- Next to stage and commit the file by running combined command **git add . && git commit -m "Add necessary notes"**
- Next to switch from private branch to public1 branch by running the command **git checkout public1**
- To merge the public1 and private two branches by running the command **git merge private**
- Next to switch from public1 branch to public2 branch by running the command **git checkout public2**
- To merge the public2 and private two branches by running the command **git merge private**
- Next to switch from public2 branch to master branch by running the command **git checkout master**
- To merge the master and private two branches by running the command **git merge private**
- Now the edited master.txt file which was done in private branch will be seen in master branch master.txt file by running the command **cat master.txt**

```
ubuntu@ip-172-31-85-101:~$ mkdir assignment4
ubuntu@ip-172-31-85-101:~$ cd assignment4
ubuntu@ip-172-31-85-101:~/assignment4$ 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/ubuntu/assignment4/.git/
ubuntu@ip-172-31-85-101:~/assignment4$
```

```
ubuntu@ip-172-31-85-101:~/assignment4$ touch master.txt
```



```
ubuntu@ip-172-31-85-101:~/assignment4$ git add master.txt
ubuntu@ip-172-31-85-101:~/assignment4$ git commit -m "master"
[master (root-commit) a4b60f7] master
  Committer: Ubuntu <ubuntu@ip-172-31-85-101.ec2.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, 0 insertions(+), 0 deletions(-)
create mode 100644 master.txt
ubuntu@ip-172-31-85-101:~/assignment4$
```

```
ubuntu@ip-172-31-85-101:~/assignment4$ git branch public1
ubuntu@ip-172-31-85-101:~/assignment4$ git branch public2
ubuntu@ip-172-31-85-101:~/assignment4$ git branch private
ubuntu@ip-172-31-85-101:~/assignment4$ git branch
* master
  private
  public1
  public2
ubuntu@ip-172-31-85-101:~/assignment4$ git checkout public1
Switched to branch 'public1'
ubuntu@ip-172-31-85-101:~/assignment4$ touch public1.txt
ubuntu@ip-172-31-85-101:~/assignment4$ git add . && git commit -m "public1.txt"
[public1 1f9e687] public1.txt
  Committer: Ubuntu <ubuntu@ip-172-31-85-101.ec2.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
```

```
ubuntu@ip-172-31-85-101:~/assignment4$ ls
master.txt  public1.txt
ubuntu@ip-172-31-85-101:~/assignment4$ git branch
  master
  private
* public1
  public2
ubuntu@ip-172-31-85-101:~/assignment4$ git checkout public2
Switched to branch 'public2'
ubuntu@ip-172-31-85-101:~/assignment4$ touch public2.txt
ubuntu@ip-172-31-85-101:~/assignment4$ git add . && git commit -m "p2"
[public2 2376ed8] p2
  Committer: Ubuntu <ubuntu@ip-172-31-85-101.ec2.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
```

```
ubuntu@ip-172-31-85-101:~/assignment4$ git checkout master
Switched to branch 'master'
ubuntu@ip-172-31-85-101:~/assignment4$ ls
master.txt
ubuntu@ip-172-31-85-101:~/assignment4$ git merge public1
Updating a4b60f7..1f9e687
Fast-forward
 public1.txt | 0
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 public1.txt
ubuntu@ip-172-31-85-101:~/assignment4$
```

```

ubuntu@ip-172-31-85-101:~/assignment4$ git checkout master
Switched to branch 'master'
ubuntu@ip-172-31-85-101:~/assignment4$ ls
master.txt
ubuntu@ip-172-31-85-101:~/assignment4$ git merge public1
Updating a4b60f7..1f9e687
Fast-forward
 public1.txt | 0
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 public1.txt
ubuntu@ip-172-31-85-101:~/assignment4$ git merge public2
Merge made by the 'ort' strategy.
 public2.txt | 0
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 public2.txt
ubuntu@ip-172-31-85-101:~/assignment4$ █

```

```

GNU nano 6.2 master.txt *
this is an edit made in private branch █

^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location   ^U Undo       ^M Set Mark   ^J To Bracket
^X Exit      ^R Read File  ^\ Replace    ^U Paste      ^J Justify    ^_/ Go To Line ^B Redo       ^G Copy       ^C Where Was

```

```

ubuntu@ip-172-31-85-101:~/assignment4$ git checkout private
Switched to branch 'private'
ubuntu@ip-172-31-85-101:~/assignment4$ ls
master.txt
ubuntu@ip-172-31-85-101:~/assignment4$ nano master.txt
ubuntu@ip-172-31-85-101:~/assignment4$ git add . && git commit -m "edited master file"
[private 2a4f54b] edited master file
Committer: Ubuntu <ubuntu@ip-172-31-85-101.ec2.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, 1 insertion(+)
ubuntu@ip-172-31-85-101:~/assignment4$ █

```

```

ubuntu@ip-172-31-85-101:~/assignment4$ git checkout master
Switched to branch 'master'
ubuntu@ip-172-31-85-101:~/assignment4$ git merge private
Merge made by the 'ort' strategy.
 master.txt | 1 +
 1 file changed, 1 insertion(+)
ubuntu@ip-172-31-85-101:~/assignment4$ █

```

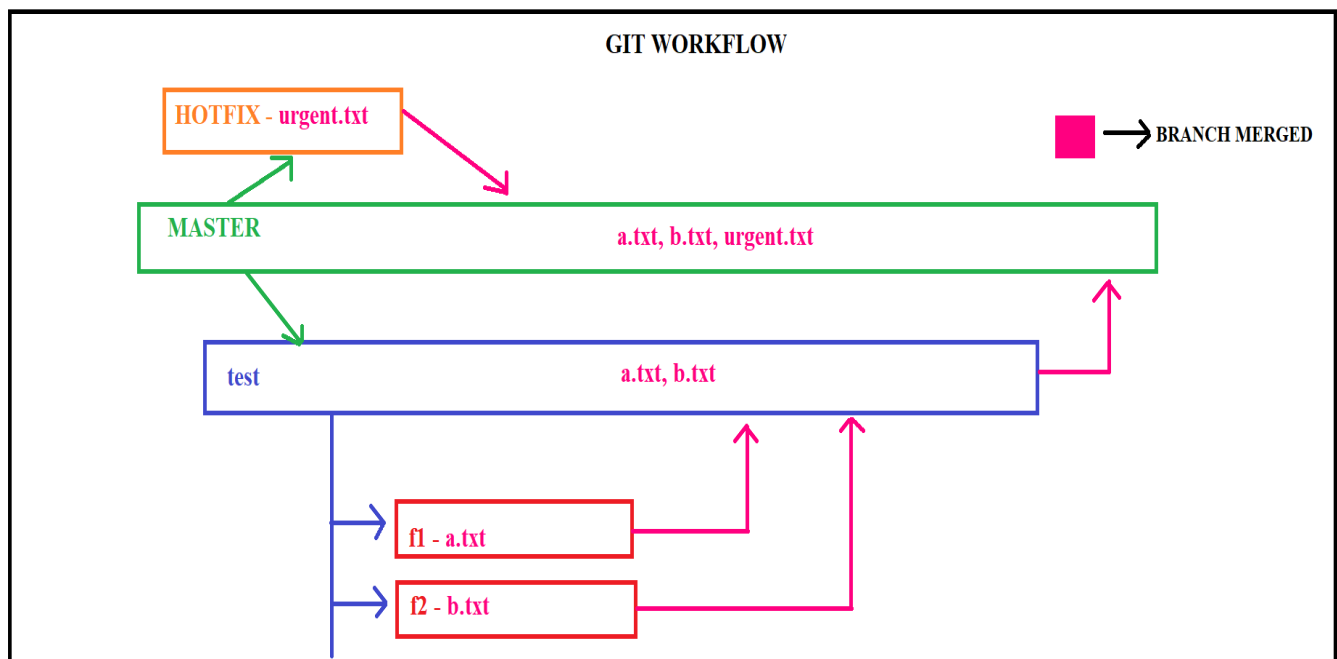
```
ubuntu@ip-172-31-85-101:~/assignment4$ ls
master.txt  public1.txt  public2.txt
ubuntu@ip-172-31-85-101:~/assignment4$ git branch
* master
  private
  public1
  public2
ubuntu@ip-172-31-85-101:~/assignment4$ cat master.txt
this is an edit made in private branch
ubuntu@ip-172-31-85-101:~/assignment4$
```

Assignment – 5

Task to be performed

1. Create a Git Flow workflow architecture on Git
2. Create all the required branches
3. Starting from the feature branch, push the breach to the master, following the architecture
4. Push a urgent.txt on master using hotfix

GIT WORK FLOW ARCHITECTURE



Steps and Commands

- Create a directory in EC2 instance by running the command **mkdir (directory name)**
- Get into the directory by running the command **cd (directory name)**
- To Initializing the git by running the command **git init**
- To create the files by running the command **touch master.txt**
- Next to stage and commit the file by running combined command **git add . && git commit -m "Add necessary notes"**
- To create new branches by running command **git branch test** and **git branch hotfix**
- Next to switch from master branch to test branch by running the command **git checkout test**

- To create new branches in test branch by running command **git branch f1** and **git branch f2**
- Next to switch from test branch to f1 branch by running the command **git checkout f1**
- To create the files by running the command **touch a.txt**
- Next to stage and commit the file by running combined command **git add . && git commit -m "Add necessary notes"**
- Next to switch from f1 branch to f2 branch by running the command **git checkout f2**
- To create the files by running the command **touch b.txt**
- Next to stage and commit the file by running combined command **git add . && git commit -m "Add necessary notes"**
- Next to switch from f2 branch to test branch by running the command **git checkout test**
- To merge the test and f1 two branches by running the command **git merge f1**
- To merge the test and f2 two branches by running the command **git merge f2**
- And Now the files which are created in f1 and f2 can be seen in the test branch

- Next to switch from test branch to master branch by running the command **git checkout master**
- To merge the master and test two branches by running the command **git merge test**
- Now the files which are created in f1 and f2 can be seen in the master branch
- Next to switch from master branch to hotfix branch by running the command **git checkout hotfix**
- To create the files by running the command **touch urgent.txt**
- Next to stage and commit the file by running combined command **git add . && git commit -m "Add necessary notes"**
- Next to switch from hotfix branch to master branch by running the command **git checkout master**
- To merge the master and hotfix two branches by running the command **git merge hotfix**
- Now the files which are created in f1, f2 and hotfix can be seen in the master branch

```
ubuntu@ip-172-31-85-101:~$ mkdir assignmentfive
ubuntu@ip-172-31-85-101:~$ cd assignmentfive
ubuntu@ip-172-31-85-101:~/assignmentfive$ 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/ubuntu/assignmentfive/.git/
ubuntu@ip-172-31-85-101:~/assignmentfive$
```

```
ubuntu@ip-172-31-85-101:~/assignmentfive$ touch master.txt
ubuntu@ip-172-31-85-101:~/assignmentfive$ git add . && git commit -m "master"
[master (root-commit) 89c4216] master
Committer: Ubuntu <ubuntu@ip-172-31-85-101.ec2.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, 0 insertions(+), 0 deletions(-)
create mode 100644 master.txt
ubuntu@ip-172-31-85-101:~/assignmentfive$
```

```
ubuntu@ip-172-31-85-101:~/assignmentfive$ git branch test
ubuntu@ip-172-31-85-101:~/assignmentfive$ git branch hotfix
ubuntu@ip-172-31-85-101:~/assignmentfive$ git branch
  hotfix
* master
  test
ubuntu@ip-172-31-85-101:~/assignmentfive$
```

```
ubuntu@ip-172-31-85-101:~/assignmentfive$ git checkout test
Switched to branch 'test'
ubuntu@ip-172-31-85-101:~/assignmentfive$ git branch f1
ubuntu@ip-172-31-85-101:~/assignmentfive$ git branch f2
ubuntu@ip-172-31-85-101:~/assignmentfive$ git branch
  f1
  f2
  hotfix
  master
* test
ubuntu@ip-172-31-85-101:~/assignmentfive$
```

```
ubuntu@ip-172-31-85-101:~/assignmentfive$ git checkout f1
Switched to branch 'f1'
ubuntu@ip-172-31-85-101:~/assignmentfive$ touch a.txt
ubuntu@ip-172-31-85-101:~/assignmentfive$ git add . && git commit -m "a file"
[f1 cfeb4aa] a file
Committer: Ubuntu <ubuntu@ip-172-31-85-101.ec2.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, 0 insertions(+), 0 deletions(-)
create mode 100644 a.txt
ubuntu@ip-172-31-85-101:~/assignmentfive$
```

```
ubuntu@ip-172-31-85-101:~/assignmentfive$ git checkout f2
Switched to branch 'f2'
ubuntu@ip-172-31-85-101:~/assignmentfive$ touch b.txt
ubuntu@ip-172-31-85-101:~/assignmentfive$ git add . && git commit -m "b file"
[f2 49a7d9b] b file
Committer: Ubuntu <ubuntu@ip-172-31-85-101.ec2.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, 0 insertions(+), 0 deletions(-)
create mode 100644 b.txt
ubuntu@ip-172-31-85-101:~/assignmentfive$
```

```
ubuntu@ip-172-31-85-101:~/assignmentfive$ git checkout test
Switched to branch 'test'
ubuntu@ip-172-31-85-101:~/assignmentfive$ git merge f1
Updating 89c4216..cfeb4aa
Fast-forward
 a.txt | 0
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 a.txt
ubuntu@ip-172-31-85-101:~/assignmentfive$ ls
a.txt master.txt
ubuntu@ip-172-31-85-101:~/assignmentfive$
```

```
ubuntu@ip-172-31-85-101:~/assignmentfive$ git merge f2
Merge made by the 'ort' strategy.
 b.txt | 0
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 b.txt
ubuntu@ip-172-31-85-101:~/assignmentfive$ ls
a.txt b.txt master.txt
ubuntu@ip-172-31-85-101:~/assignmentfive$
```

```
ubuntu@ip-172-31-85-101:~/assignmentfive$ git checkout master
Switched to branch 'master'
ubuntu@ip-172-31-85-101:~/assignmentfive$ git merge test
Updating 89c4216..140061f
Fast-forward
 a.txt | 0
 b.txt | 0
 2 files changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 a.txt
 create mode 100644 b.txt
ubuntu@ip-172-31-85-101:~/assignmentfive$ ls
a.txt b.txt master.txt
ubuntu@ip-172-31-85-101:~/assignmentfive$
```

```
ubuntu@ip-172-31-85-101:~/assignmentfive$ git checkout hotfix
Switched to branch 'hotfix'
ubuntu@ip-172-31-85-101:~/assignmentfive$ touch urgent.txt
ubuntu@ip-172-31-85-101:~/assignmentfive$ git add . && git commit -m "urgent file"
[hotfix d2c2ca7] urgent file
Committer: Ubuntu <ubuntu@ip-172-31-85-101.ec2.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, 0 insertions(+), 0 deletions(-)
 create mode 100644 urgent.txt
ubuntu@ip-172-31-85-101:~/assignmentfive$
```

```
ubuntu@ip-172-31-85-101:~/assignmentfive$ git checkout master
Switched to branch 'master'
ubuntu@ip-172-31-85-101:~/assignmentfive$ git merge hotfix
Merge made by the 'ort' strategy.
 urgent.txt | 0
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 urgent.txt
ubuntu@ip-172-31-85-101:~/assignmentfive$ ls
a.txt b.txt master.txt urgent.txt
ubuntu@ip-172-31-85-101:~/assignmentfive$
```