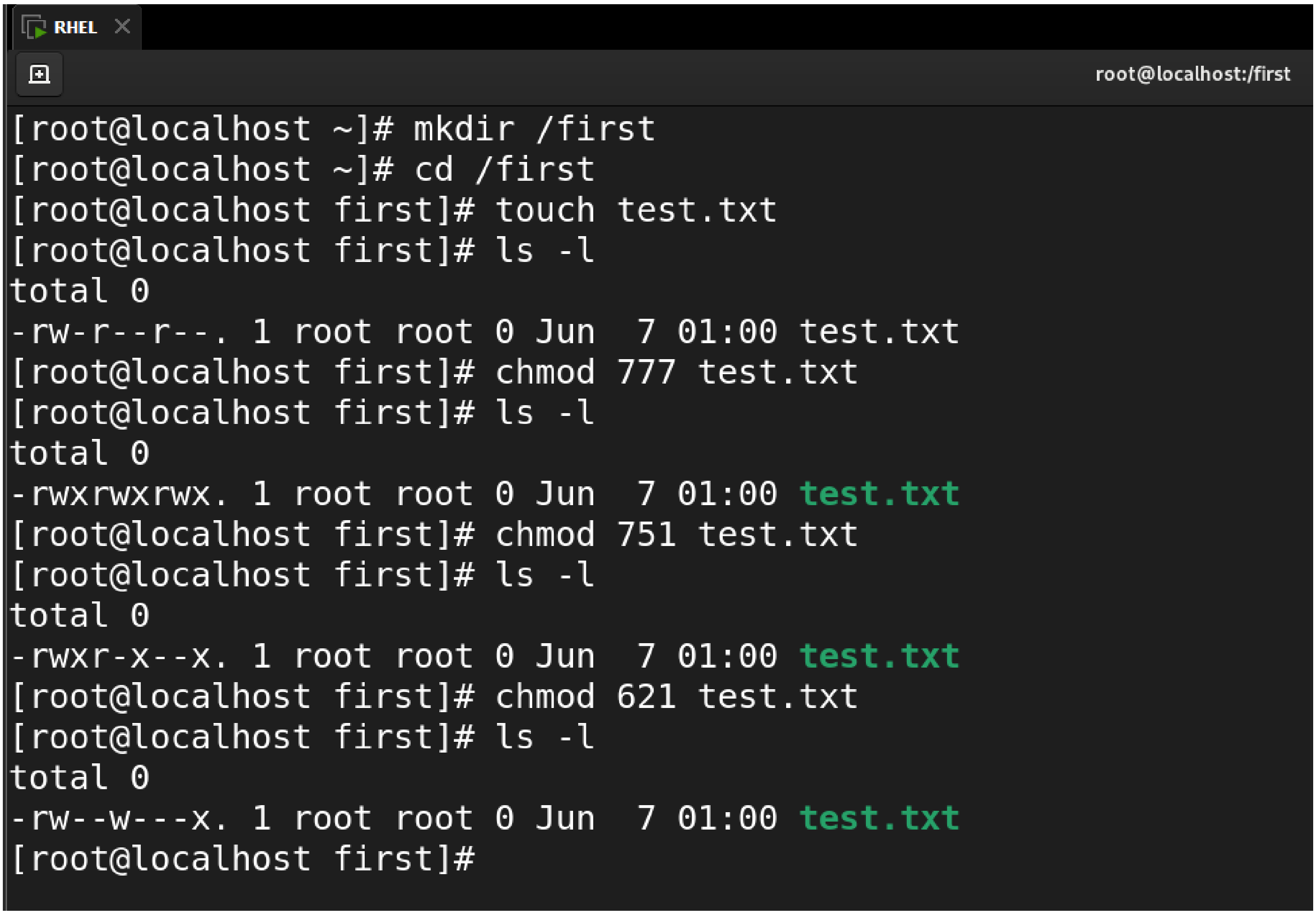
CELEBAL SUMMER INTERNSHIP

Assignment- Linux/Git

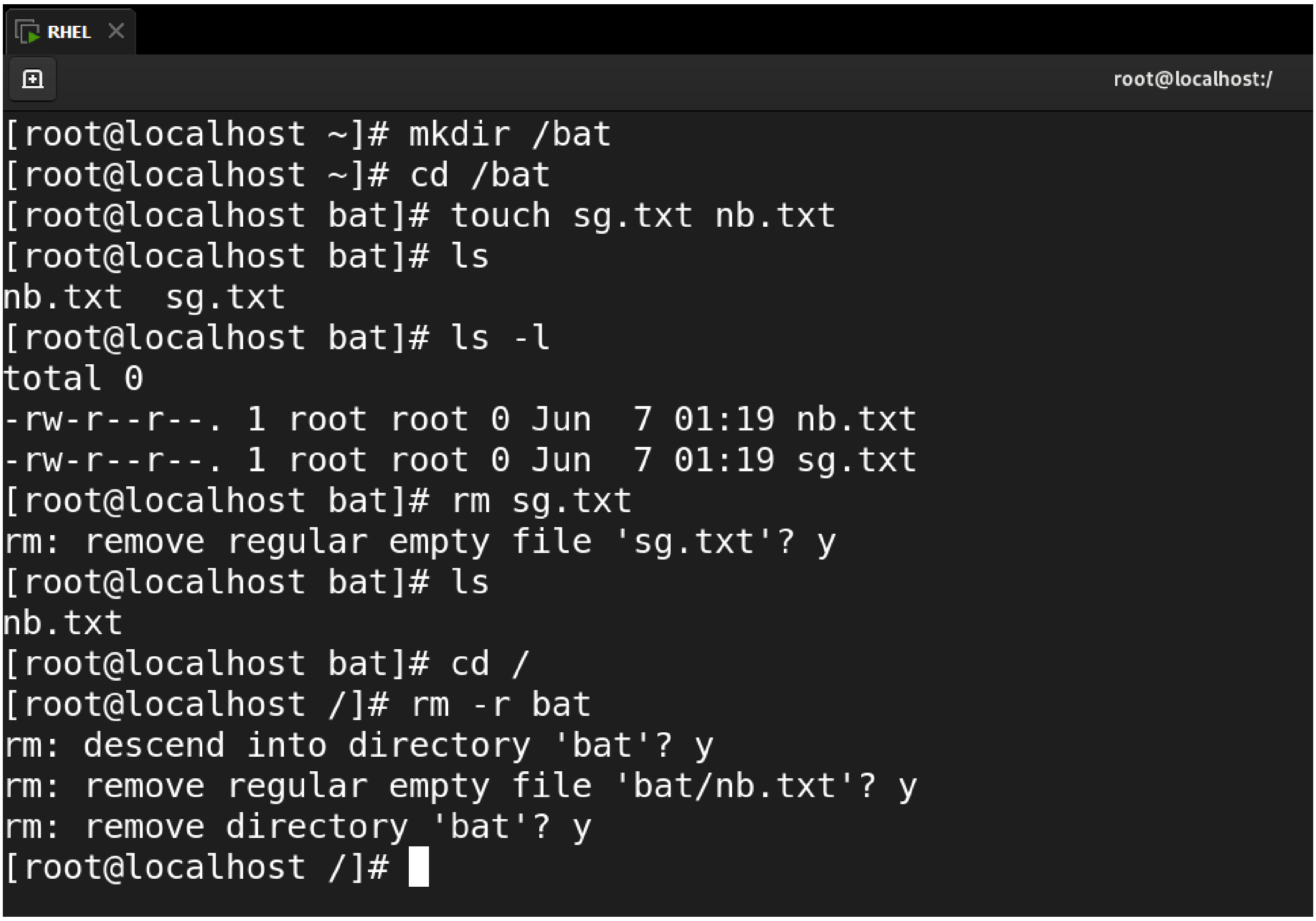
Submitted by: Hemansh Sankhla

- Create a file, assign permissions (read, write, execute) to different user categories (owner, group, others), and practice changing permissions using chmod.



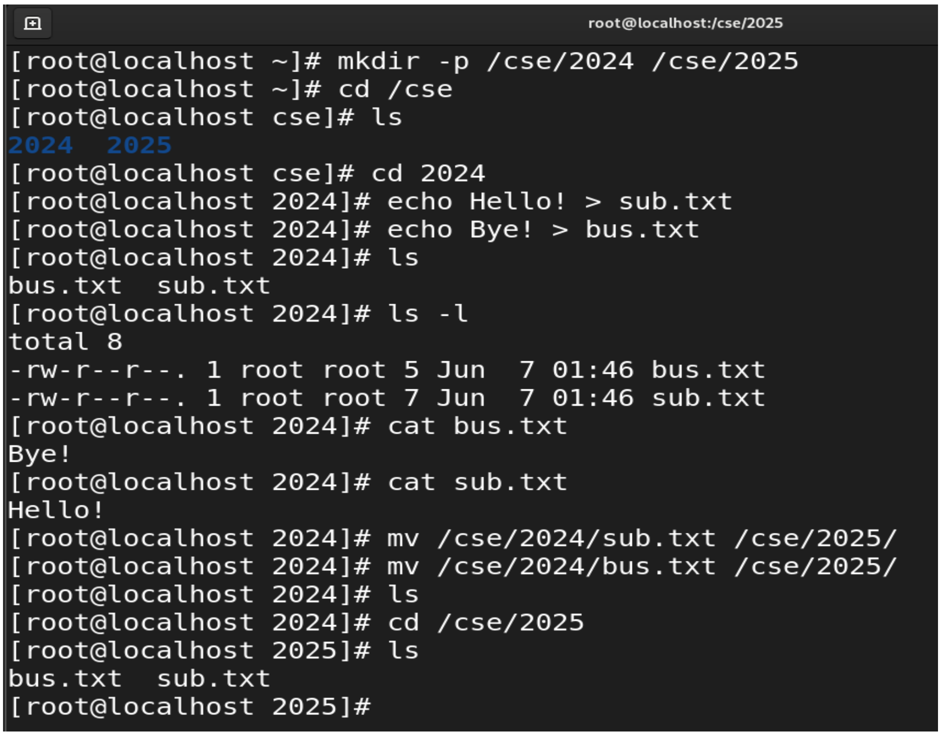
```
[root@localhost ~]# mkdir /first
[root@localhost ~]# cd /first
[root@localhost first]# touch test.txt
[root@localhost first]# ls -l
total 0
-rw-r--r--. 1 root root 0 Jun  7 01:00 test.txt
[root@localhost first]# chmod 777 test.txt
[root@localhost first]# ls -l
total 0
-rwxrwxrwx. 1 root root 0 Jun  7 01:00 test.txt
[root@localhost first]# chmod 751 test.txt
[root@localhost first]# ls -l
total 0
-rwxr-x--x. 1 root root 0 Jun  7 01:00 test.txt
[root@localhost first]# chmod 621 test.txt
[root@localhost first]# ls -l
total 0
-rw--w---x. 1 root root 0 Jun  7 01:00 test.txt
[root@localhost first]#
```

- Execute basic Linux commands (e.g., ls, cd, mkdir, rm, touch) to manipulate files and directories, with an emphasis on understanding their usage.



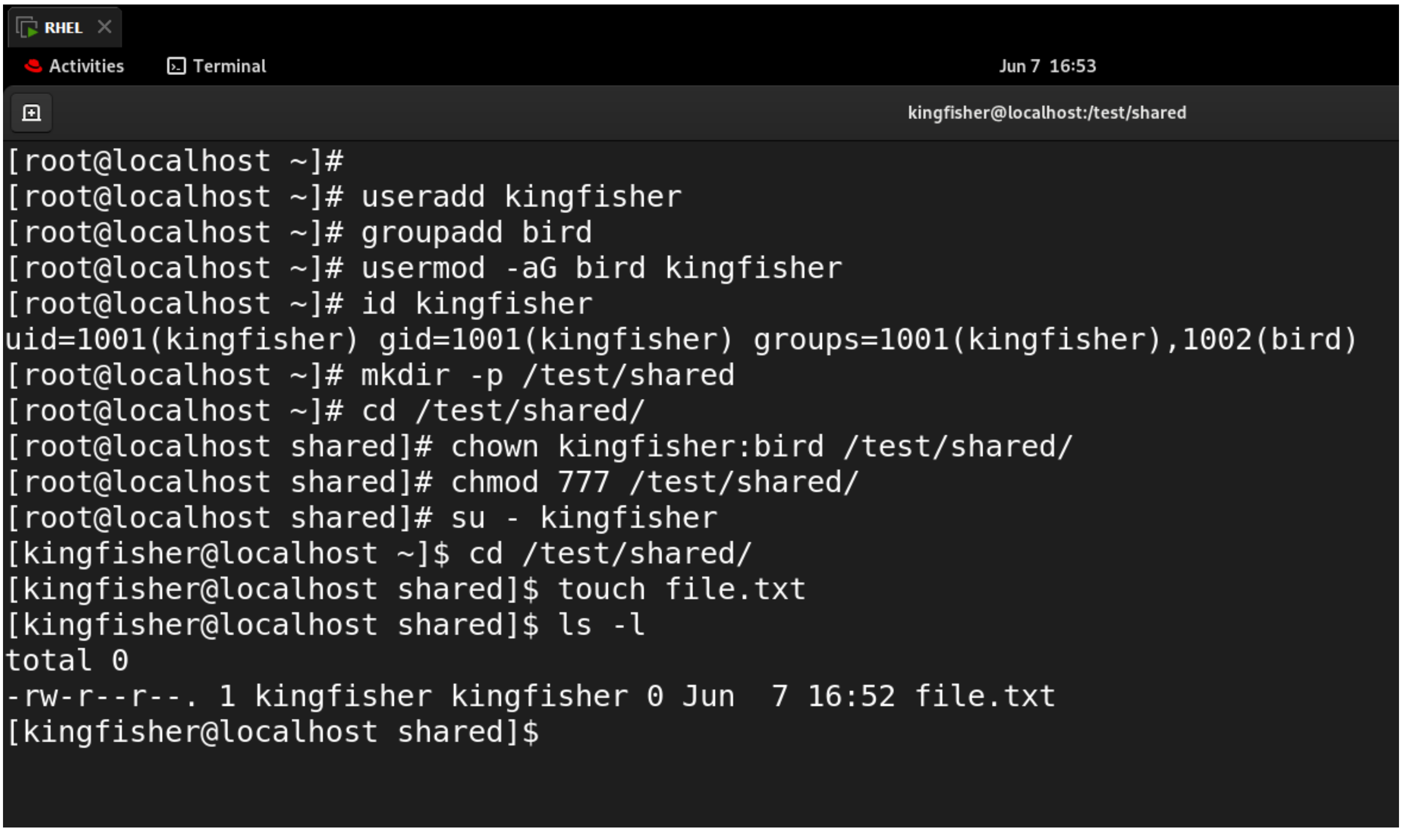
```
[root@localhost ~]# mkdir /bat
[root@localhost ~]# cd /bat
[root@localhost bat]# touch sg.txt nb.txt
[root@localhost bat]# ls
nb.txt  sg.txt
[root@localhost bat]# ls -l
total 0
-rw-r--r--. 1 root root 0 Jun  7 01:19 nb.txt
-rw-r--r--. 1 root root 0 Jun  7 01:19 sg.txt
[root@localhost bat]# rm sg.txt
rm: remove regular empty file 'sg.txt'? y
[root@localhost bat]# ls
nb.txt
[root@localhost bat]# cd /
[root@localhost /]# rm -r bat
rm: descend into directory 'bat'? y
rm: remove regular empty file 'bat/nb.txt'? y
rm: remove directory 'bat'? y
[root@localhost /]# 
```

- ❖ **Using the terminal, practice navigating through directories, listing file contents, and moving files to different locations.**



```
[root@localhost ~]# mkdir -p /cse/2024 /cse/2025
[root@localhost ~]# cd /cse
[root@localhost cse]# ls
2024  2025
[root@localhost cse]# cd 2024
[root@localhost 2024]# echo Hello! > sub.txt
[root@localhost 2024]# echo Bye! > bus.txt
[root@localhost 2024]# ls
bus.txt  sub.txt
[root@localhost 2024]# ls -l
total 8
-rw-r--r--. 1 root root 5 Jun  7 01:46 bus.txt
-rw-r--r--. 1 root root 7 Jun  7 01:46 sub.txt
[root@localhost 2024]# cat bus.txt
Bye!
[root@localhost 2024]# cat sub.txt
Hello!
[root@localhost 2024]# mv /cse/2024/sub.txt /cse/2025/
[root@localhost 2024]# mv /cse/2024/bus.txt /cse/2025/
[root@localhost 2024]# ls
[root@localhost 2024]# cd /cse/2025
[root@localhost 2025]# ls
bus.txt  sub.txt
[root@localhost 2025]#
```

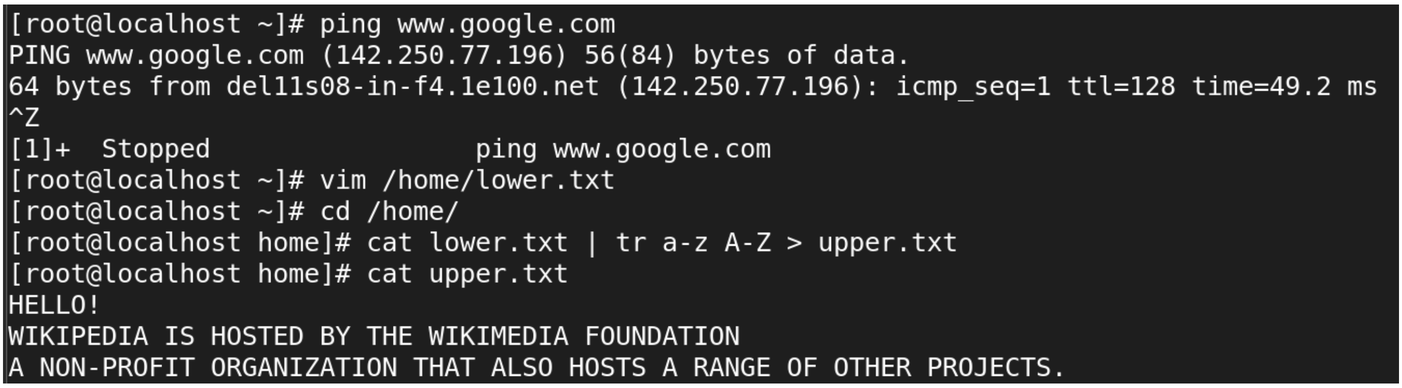
❖ Create a new user and group, set their permissions, and explore user management commands like useradd, usermod, and userdel.



```
[root@localhost ~]#
[root@localhost ~]# useradd kingfisher
[root@localhost ~]# groupadd bird
[root@localhost ~]# usermod -aG bird kingfisher
[root@localhost ~]# id kingfisher
uid=1001(kingfisher) gid=1001(kingfisher) groups=1001(kingfisher),1002(bird)
[root@localhost ~]# mkdir -p /test/shared
[root@localhost ~]# cd /test/shared/
[root@localhost shared]# chown kingfisher:bird /test/shared/
[root@localhost shared]# chmod 777 /test/shared/
[root@localhost shared]# su - kingfisher
[kingfisher@localhost ~]$ cd /test/shared/
[kingfisher@localhost shared]$ touch file.txt
[kingfisher@localhost shared]$ ls -l
total 0
-rw-r--r--. 1 kingfisher kingfisher 0 Jun  7 16:52 file.txt
[kingfisher@localhost shared]$
```

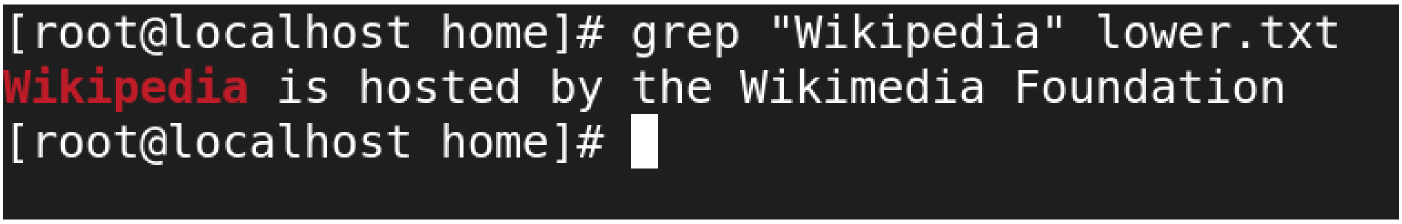
❖ Practice more linux commands

1. ping and translate command



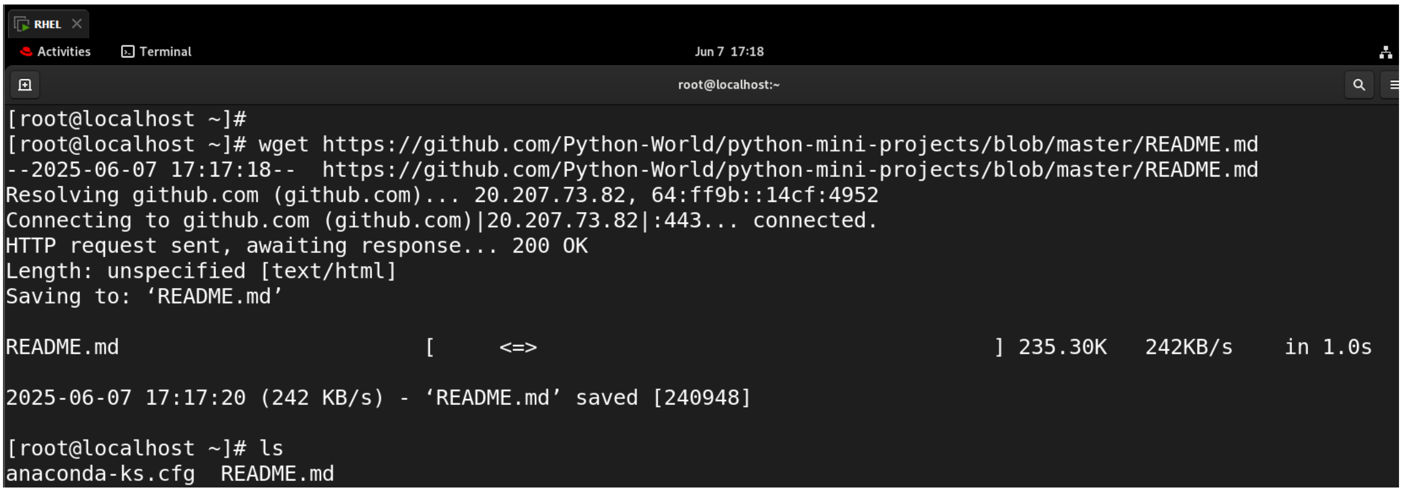
```
[root@localhost ~]# ping www.google.com
PING www.google.com (142.250.77.196) 56(84) bytes of data.
64 bytes from del11s08-in-f4.1e100.net (142.250.77.196): icmp_seq=1 ttl=128 time=49.2 ms
^Z
[1]+  Stopped                 ping www.google.com
[root@localhost ~]# vim /home/lower.txt
[root@localhost ~]# cd /home/
[root@localhost home]# cat lower.txt | tr a-z A-Z > upper.txt
[root@localhost home]# cat upper.txt
HELLO!
WIKIPEDIA IS HOSTED BY THE WIKIMEDIA FOUNDATION
A NON-PROFIT ORGANIZATION THAT ALSO HOSTS A RANGE OF OTHER PROJECTS.
```

2. grep command



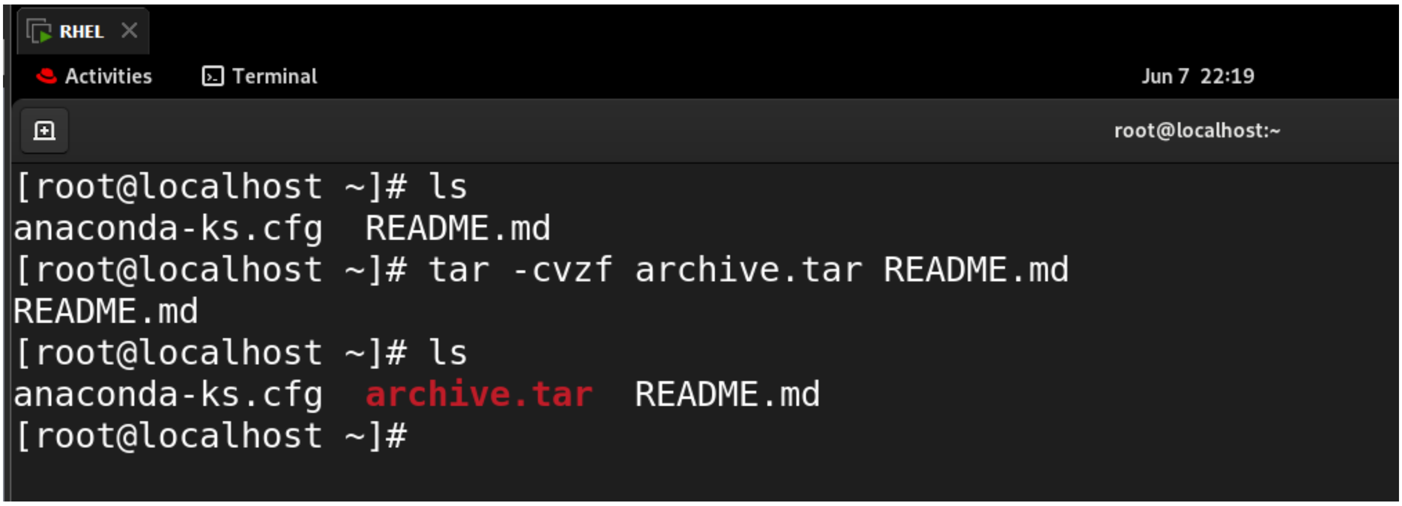
```
[root@localhost home]# grep "Wikipedia" lower.txt
Wikipedia is hosted by the Wikimedia Foundation
[root@localhost home]# 
```

3. wget command



```
[root@localhost ~]#
[root@localhost ~]# wget https://github.com/Python-World/python-mini-projects/blob/master/README.md
--2025-06-07 17:17:18--  https://github.com/Python-World/python-mini-projects/blob/master/README.md
Resolving github.com (github.com)... 20.207.73.82, 64:ff9b::14cf:4952
Connecting to github.com (github.com)|20.207.73.82|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [text/html]
Saving to: ‘README.md’

README.md                        [     <=>                                     ] 235.30K   242KB/s    in 1.0s

2025-06-07 17:17:20 (242 KB/s) - ‘README.md’ saved [240948]

[root@localhost ~]# ls
anaconda-ks.cfg  README.md
```

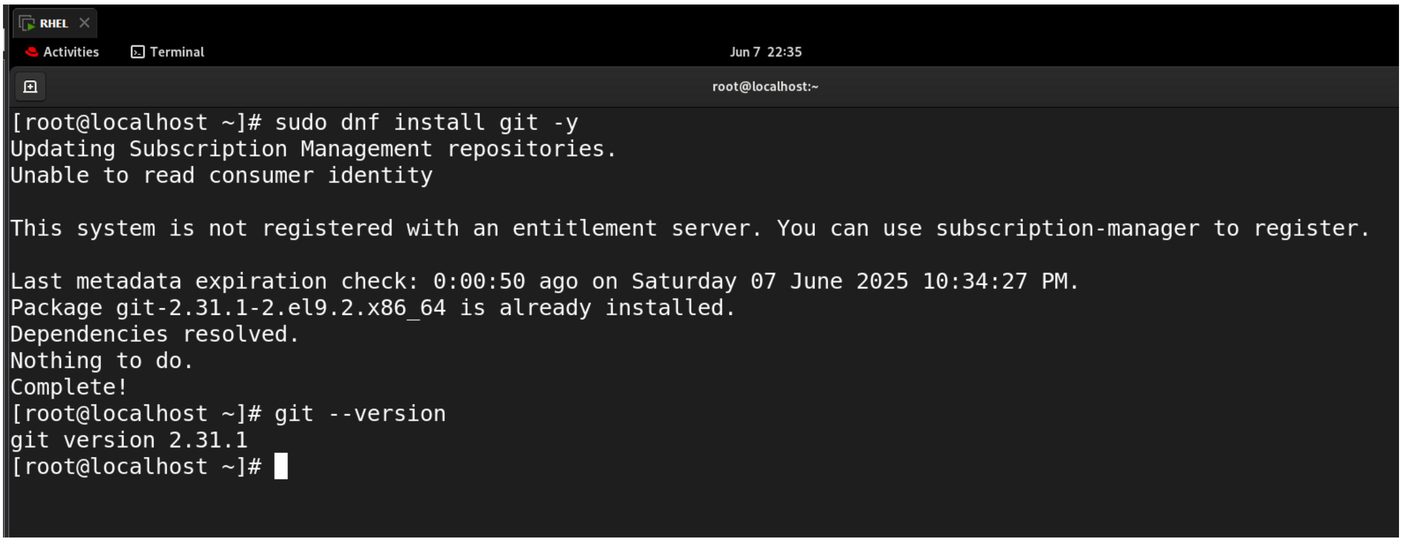
4. tar command



```
[root@localhost ~]# ls
anaconda-ks.cfg  README.md
[root@localhost ~]# tar -cvzf archive.tar README.md
README.md
[root@localhost ~]# ls
anaconda-ks.cfg  archive.tar  README.md
[root@localhost ~]#
```

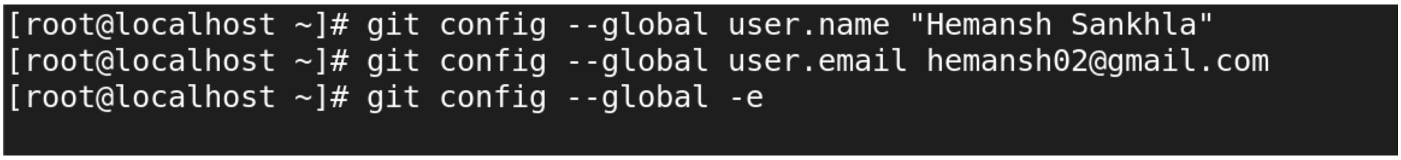
❖ Introduction to version control and Git basics, Git installation and configuration, Basic Git commands (init, add, commit, push, pull)

1. Git Installation

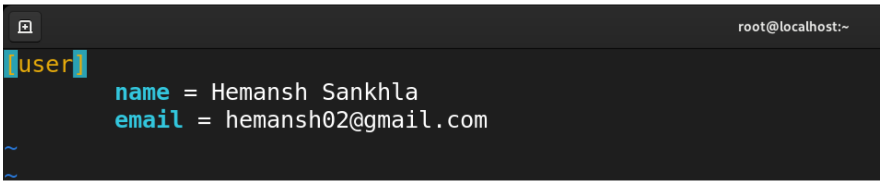


```
[root@localhost ~]# sudo dnf install git -y
Updating Subscription Management repositories.
Unable to read consumer identity

This system is not registered with an entitlement server. You can use subscription-manager to register.

Last metadata expiration check: 0:00:50 ago on Saturday 07 June 2025 10:34:27 PM.
Package git-2.31.1-2.el9.2.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[root@localhost ~]# git --version
git version 2.31.1
[root@localhost ~]#
```

2. Git Configuration

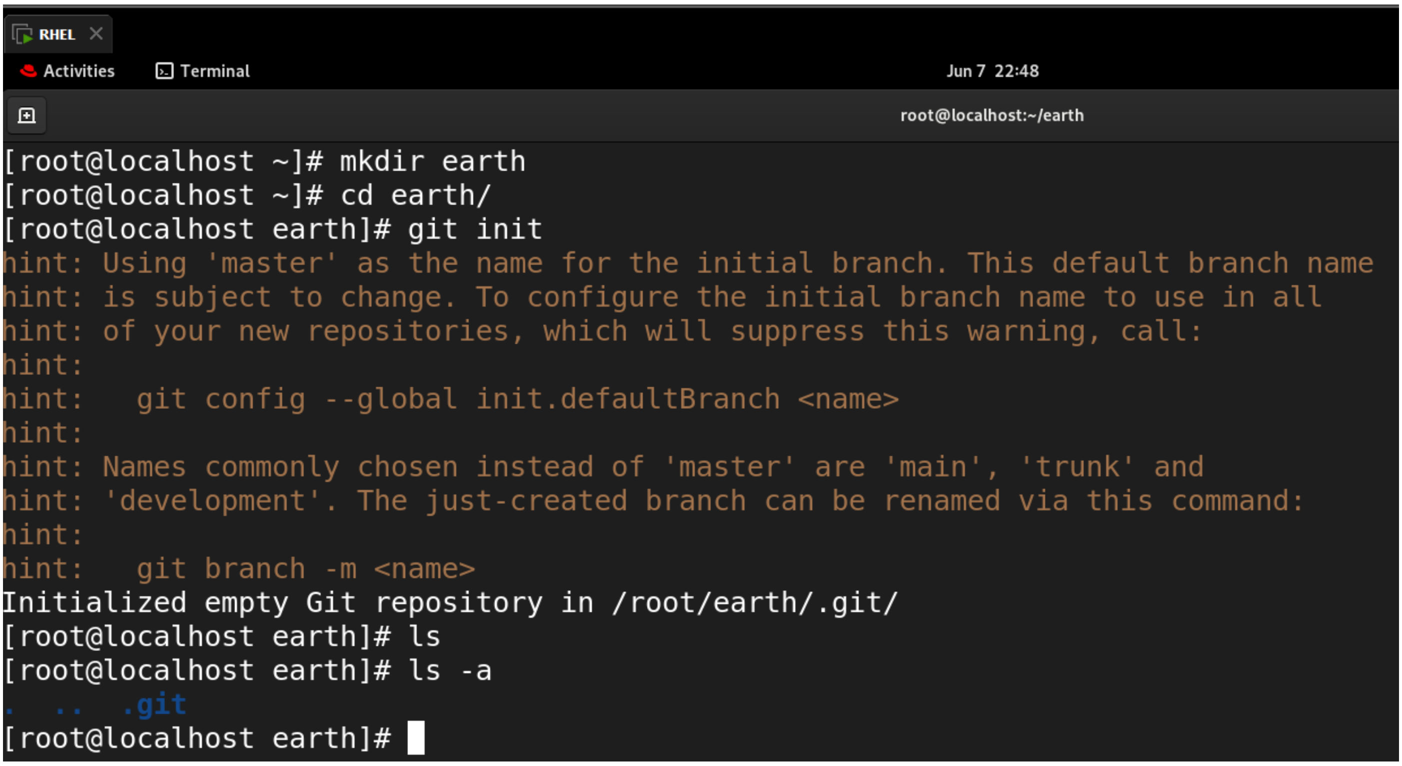


```
[root@localhost ~]# git config --global user.name "Hemansh Sankhla"
[root@localhost ~]# git config --global user.email hemansh02@gmail.com
[root@localhost ~]# git config --global -e
```



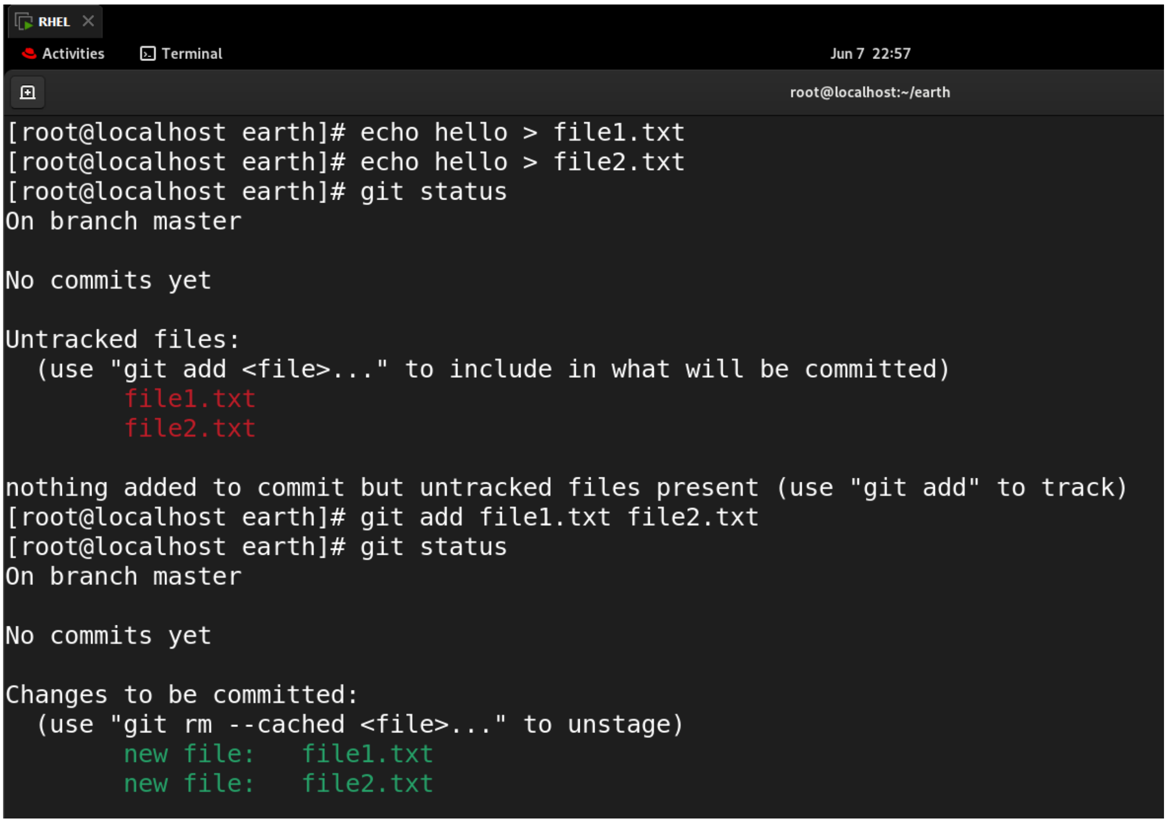
```
[user]
        name = Hemansh Sankhla
        email = hemansh02@gmail.com
~
~
```

3. Initializing Git Repository



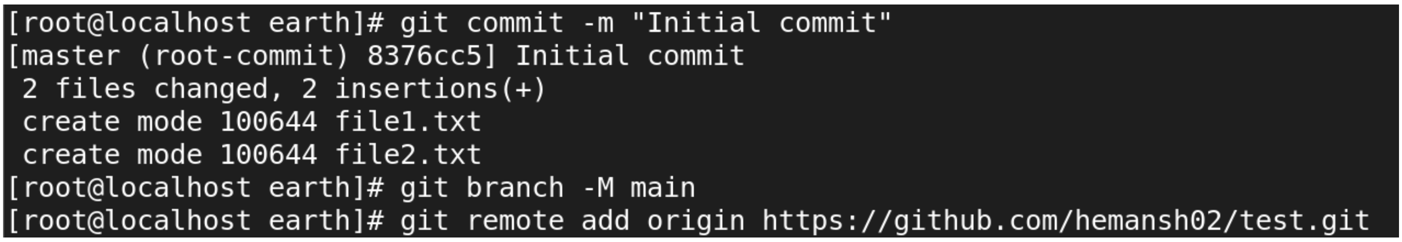
```
[root@localhost ~]# mkdir earth
[root@localhost ~]# cd earth/
[root@localhost earth]# git init
hint: Using 'master' as the name for the initial branch. This default branch name
hint: is subject to change. To configure the initial branch name to use in all
hint: of your new repositories, which will 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 /root/earth/.git/
[root@localhost earth]# ls
[root@localhost earth]# ls -a
.  ..  .git
[root@localhost earth]#
```

4. Creating files, adding them, and checking the status



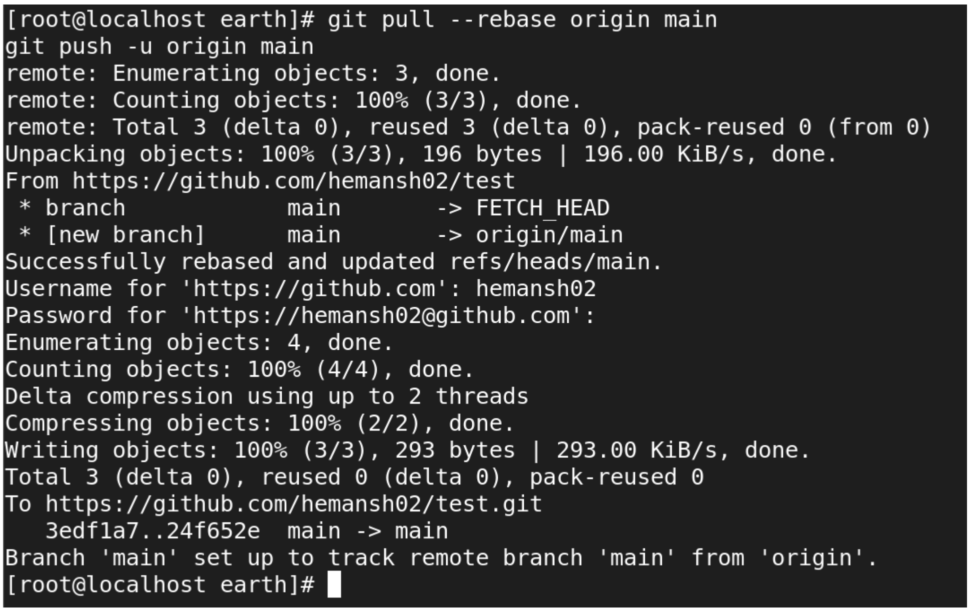
```
[root@localhost earth]# echo hello > file1.txt
[root@localhost earth]# echo hello > file2.txt
[root@localhost earth]# git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        file1.txt
        file2.txt

nothing added to commit but untracked files present (use "git add" to track)
[root@localhost earth]# git add file1.txt file2.txt
[root@localhost earth]# git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   file1.txt
        new file:   file2.txt
```

5. Commit the files, Create main branch, and Add remote origin



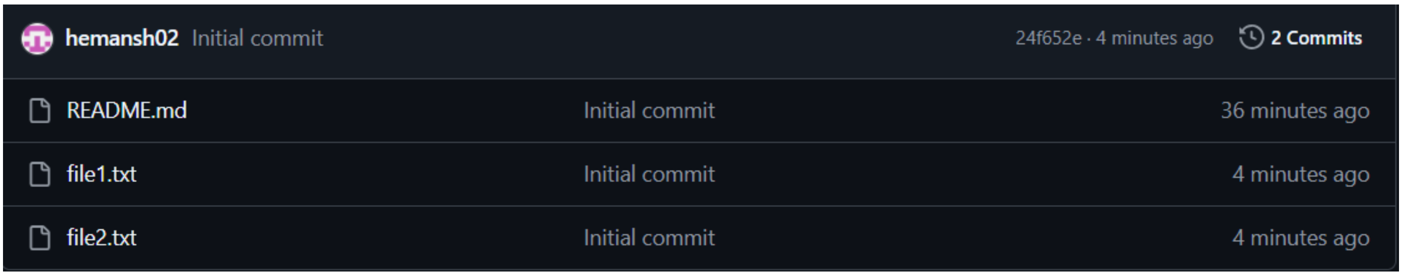
```
[root@localhost earth]# git commit -m "Initial commit"
[master (root-commit) 8376cc5] Initial commit
 2 files changed, 2 insertions(+)
 create mode 100644 file1.txt
 create mode 100644 file2.txt
[root@localhost earth]# git branch -M main
[root@localhost earth]# git remote add origin https://github.com/hemansh02/test.git
```

6. Pull the remote changes, then push.



```
[root@localhost earth]# git pull --rebase origin main
git push -u origin main
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 3 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (3/3), 196 bytes | 196.00 KiB/s, done.
From https://github.com/hemansh02/test
 * branch            main       -> FETCH_HEAD
 * [new branch]      main       -> origin/main
Successfully rebased and updated refs/heads/main.
Username for 'https://github.com': hemansh02
Password for 'https://hemansh02@github.com':
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 2 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 293 bytes | 293.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/hemansh02/test.git
   3edf1a7..24f652e  main -> main
Branch 'main' set up to track remote branch 'main' from 'origin'.
[root@localhost earth]# █
```

7. Finalized results in the github repo (remote branch)

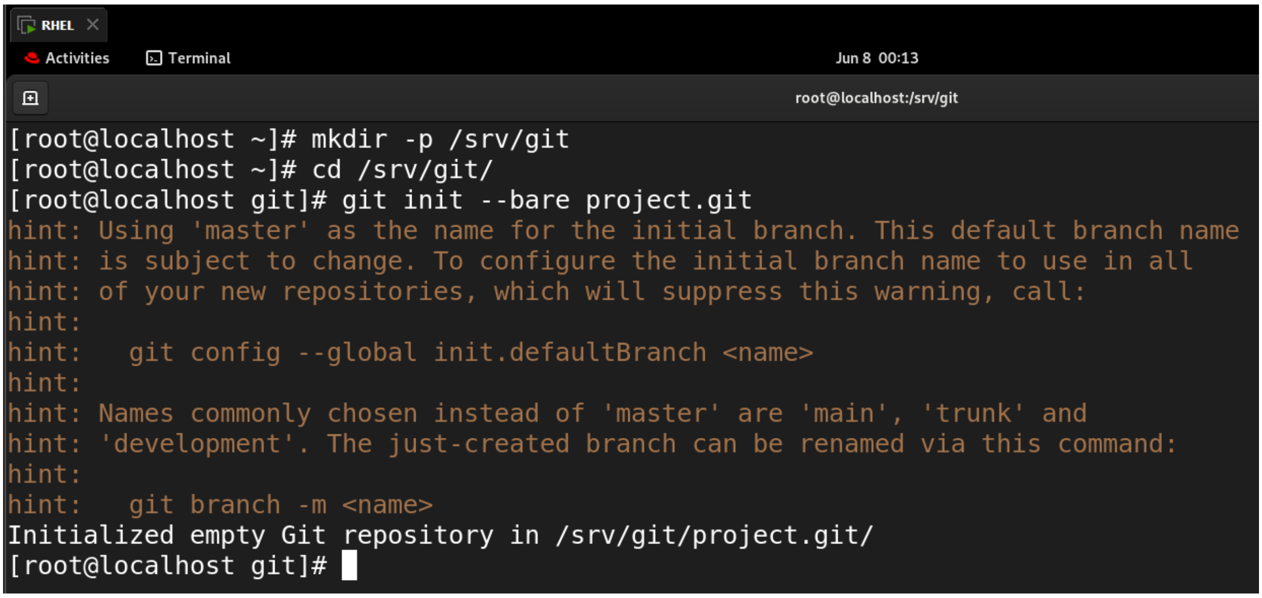


hemansh02 Initial commit 24f652e · 4 minutes ago 2 Commits

README.md	Initial commit	36 minutes ago
file1.txt	Initial commit	4 minutes ago
file2.txt	Initial commit	4 minutes ago

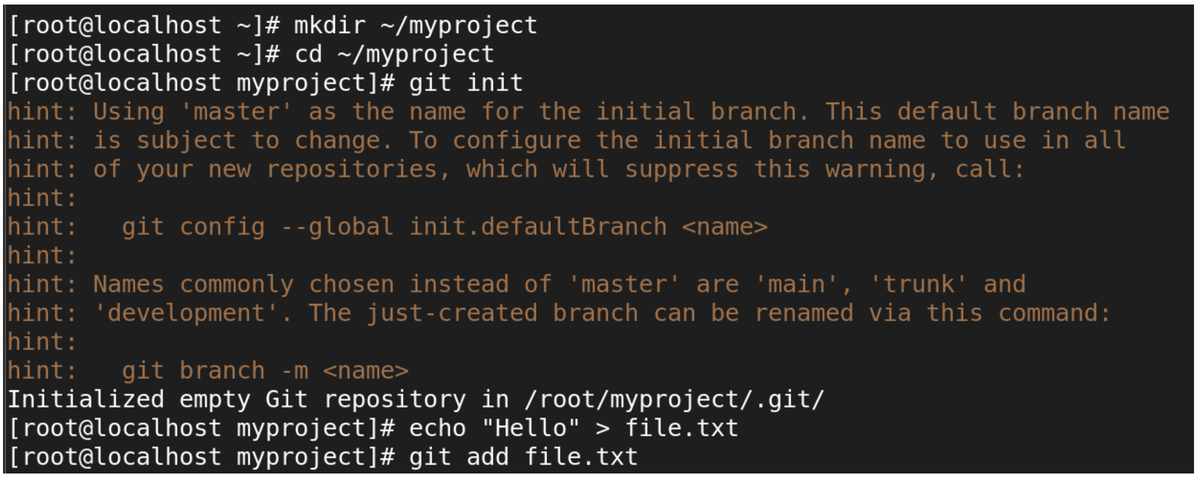
- ❖ **Setup a remote repository in Local, add a file and commit or save the changes and push to master branch**

1. Create a local bare remote repo

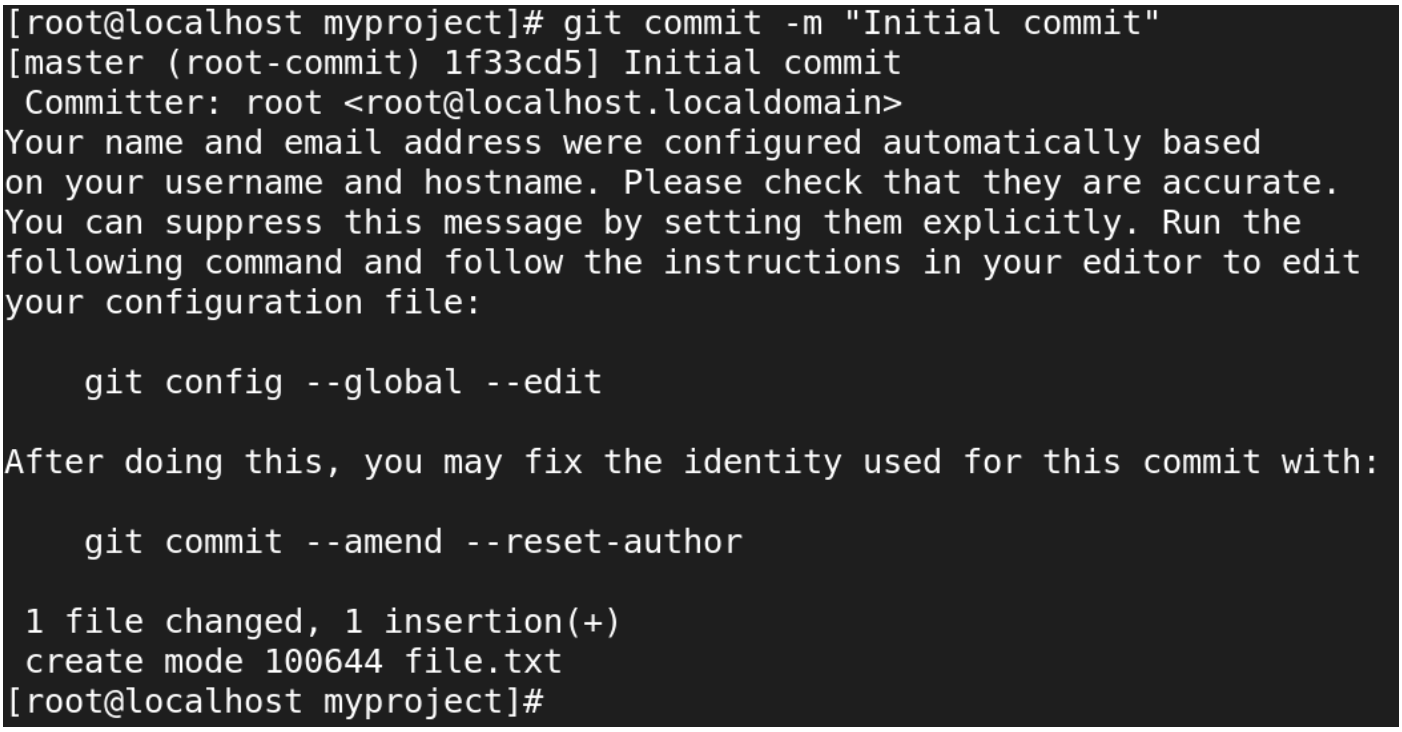


```
[root@localhost ~]# mkdir -p /srv/git
[root@localhost ~]# cd /srv/git/
[root@localhost git]# git init --bare project.git
hint: Using 'master' as the name for the initial branch. This default branch name
hint: is subject to change. To configure the initial branch name to use in all
hint: of your new 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 /srv/git/project.git/
[root@localhost git]# 
```

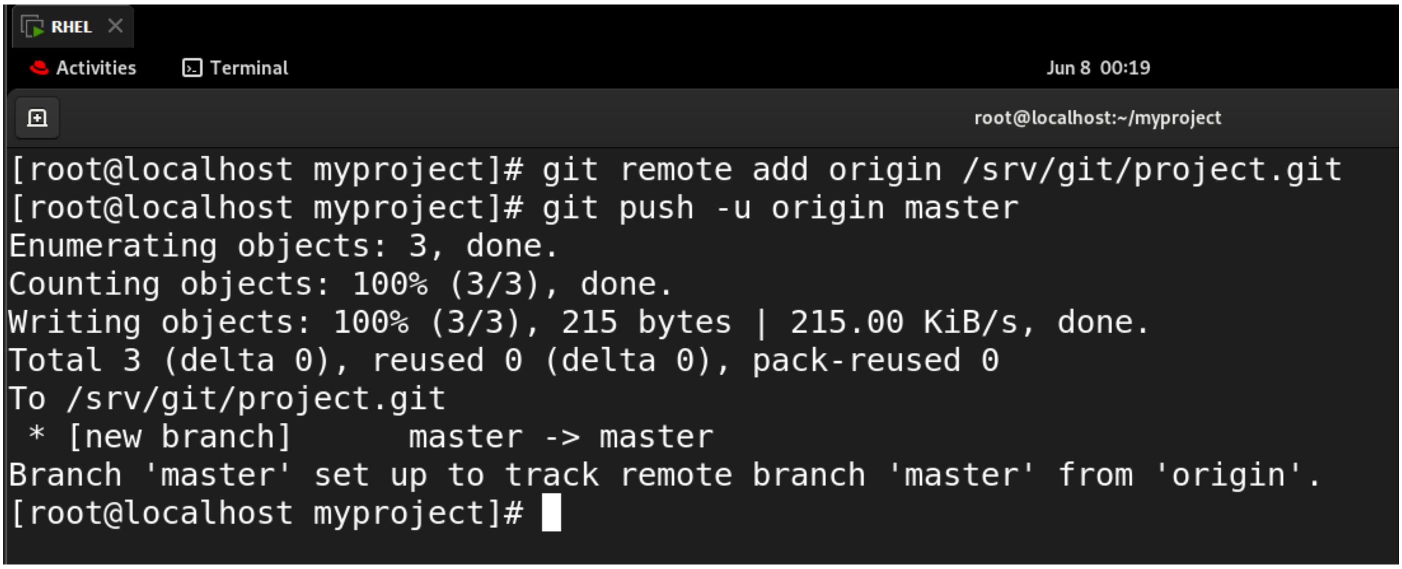
2. Create a working project and link the local remote



```
[root@localhost ~]# mkdir ~/myproject
[root@localhost ~]# cd ~/myproject
[root@localhost myproject]# 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 /root/myproject/.git/
[root@localhost myproject]# echo "Hello" > file.txt
[root@localhost myproject]# git add file.txt
```

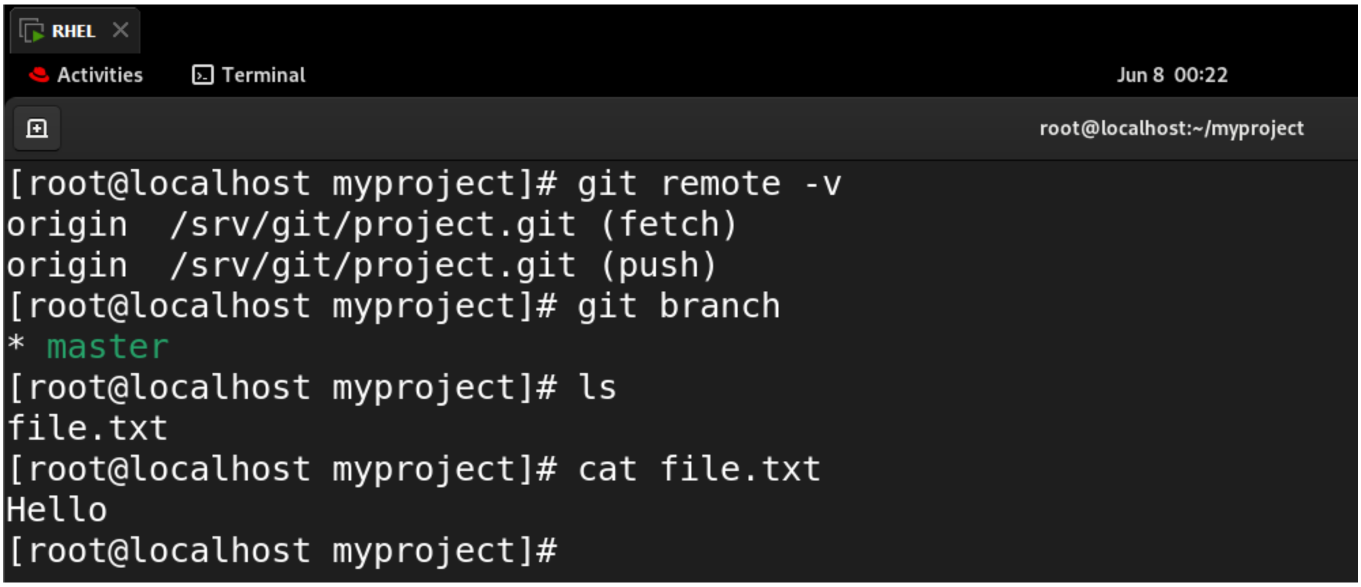


```
[root@localhost myproject]# git commit -m "Initial commit"
[master (root-commit) 1f33cd5] Initial commit
 Committer: root <root@localhost.localdomain>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress 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(+)
 create mode 100644 file.txt
[root@localhost myproject]#
```



```
[root@localhost myproject]# git remote add origin /srv/git/project.git
[root@localhost myproject]# git push -u origin master
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 215 bytes | 215.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To /srv/git/project.git
 * [new branch]      master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.
[root@localhost myproject]#
```

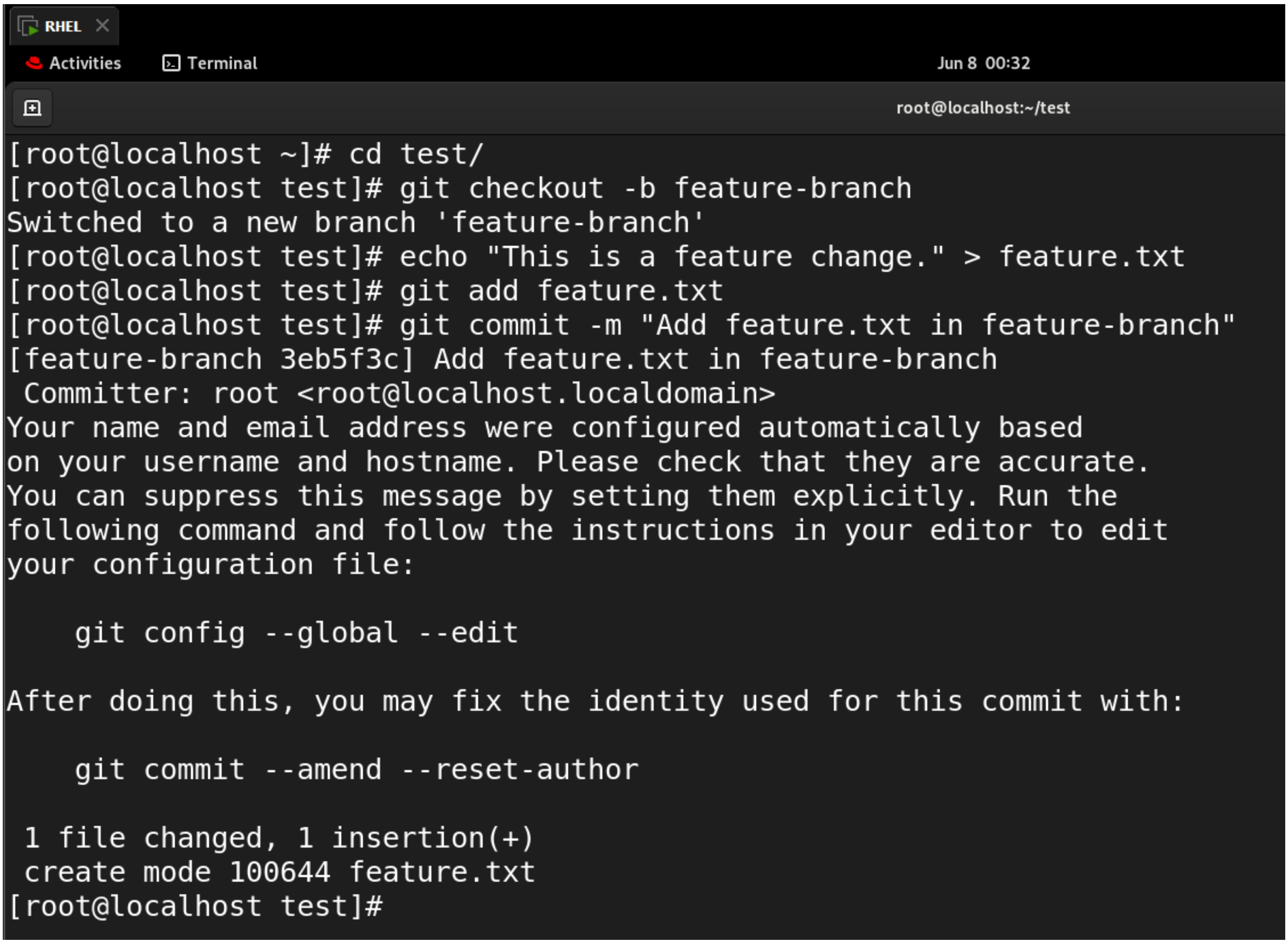
3. Check the remote origin, current branch, and file.



```
[root@localhost myproject]# git remote -v
origin  /srv/git/project.git (fetch)
origin  /srv/git/project.git (push)
[root@localhost myproject]# git branch
* master
[root@localhost myproject]# ls
file.txt
[root@localhost myproject]# cat file.txt
Hello
[root@localhost myproject]#
```

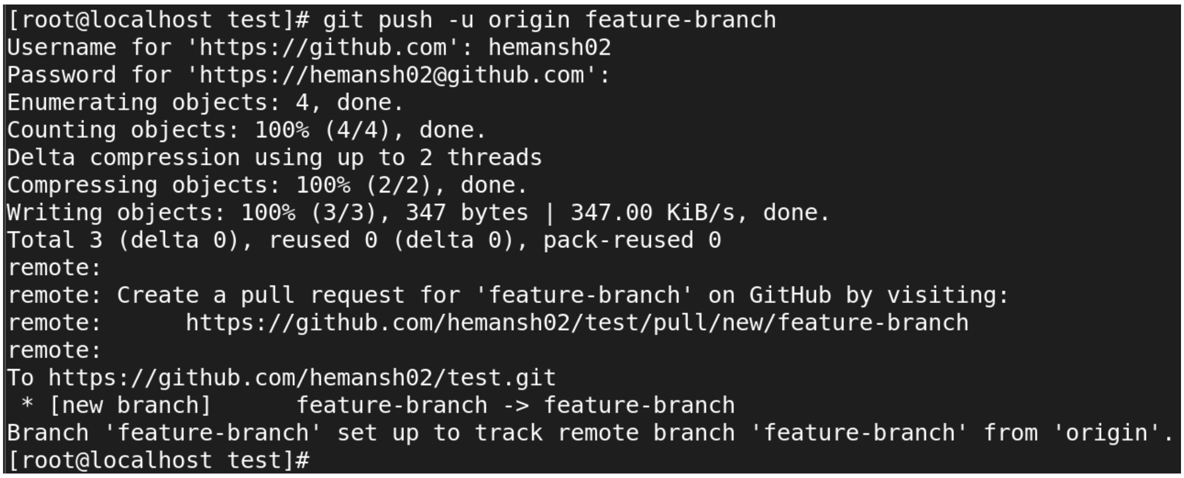
❖ Merge types, Create a new branch then commit and push the changes to new branch and merge it with the master branch using pull request.

1. Create a New Branch, Add a File, and Commit.



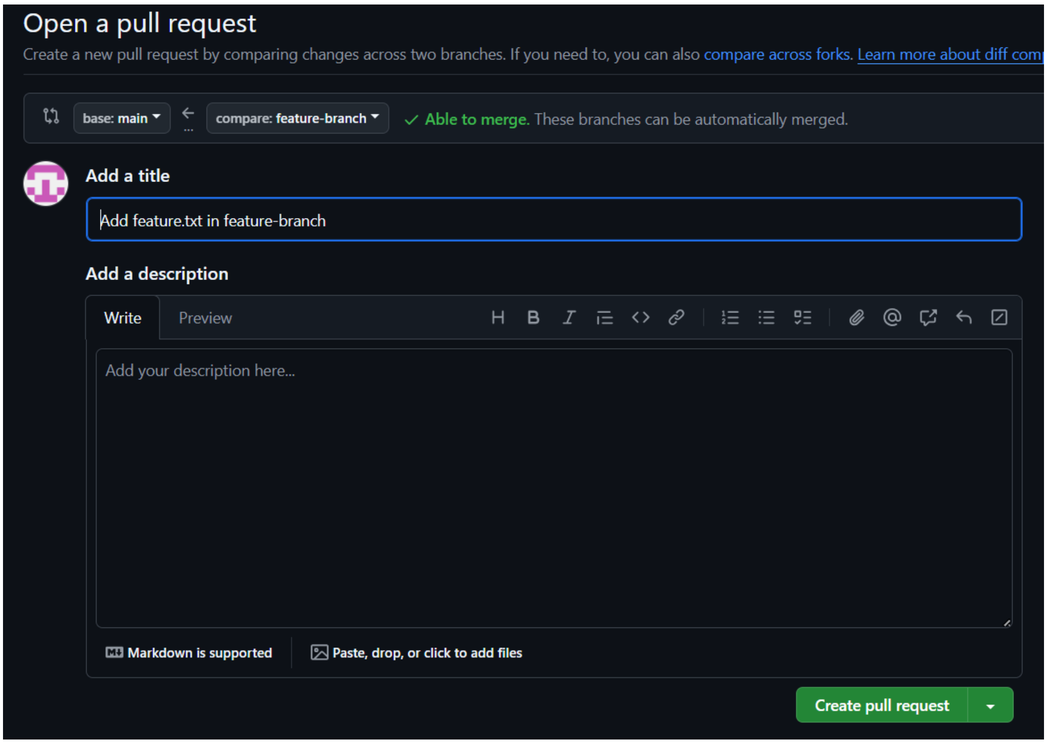
```
[root@localhost ~]# cd test/
[root@localhost test]# git checkout -b feature-branch
Switched to a new branch 'feature-branch'
[root@localhost test]# echo "This is a feature change." > feature.txt
[root@localhost test]# git add feature.txt
[root@localhost test]# git commit -m "Add feature.txt in feature-branch"
[feature-branch 3eb5f3c] Add feature.txt in feature-branch
 Committer: root <root@localhost.localdomain>
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(+)
 create mode 100644 feature.txt
[root@localhost test]#
```

2. Push the Branch to Github



```
[root@localhost test]# git push -u origin feature-branch
Username for 'https://github.com': hemansh02
Password for 'https://hemansh02@github.com':
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 2 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 347 bytes | 347.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'feature-branch' on GitHub by visiting:
remote:      https://github.com/hemansh02/test/pull/new/feature-branch
remote:
To https://github.com/hemansh02/test.git
 * [new branch]      feature-branch -> feature-branch
Branch 'feature-branch' set up to track remote branch 'feature-branch' from 'origin'.
[root@localhost test]#
```

3. In Github Create Pull Request



Open a pull request

Create a new pull request by comparing changes across two branches. If you need to, you can also compare across forks. Learn more about diff com

✓ Able to merge. These branches can be automatically merged.

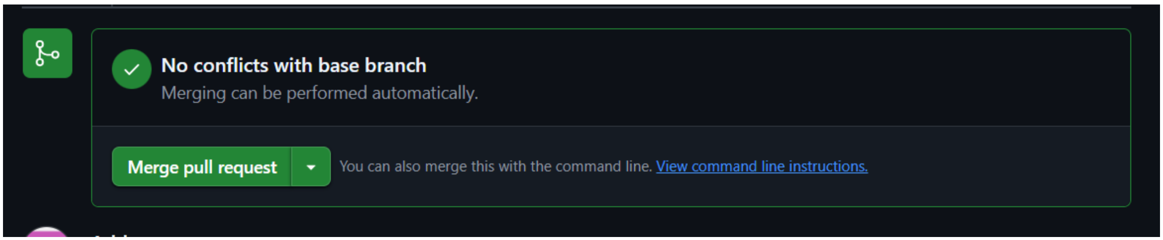
Add a title

Add feature.txt in feature-branch

Add a description

Add your description here...

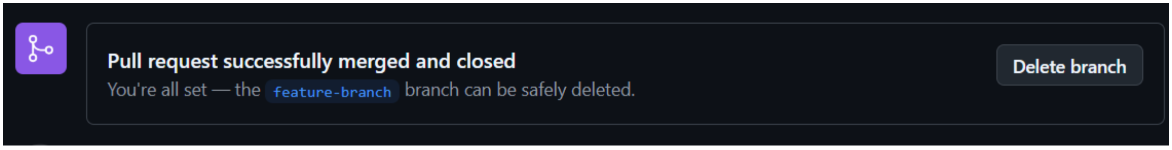
4. Merge the Pull Request



No conflicts with base branch

Merging can be performed automatically.

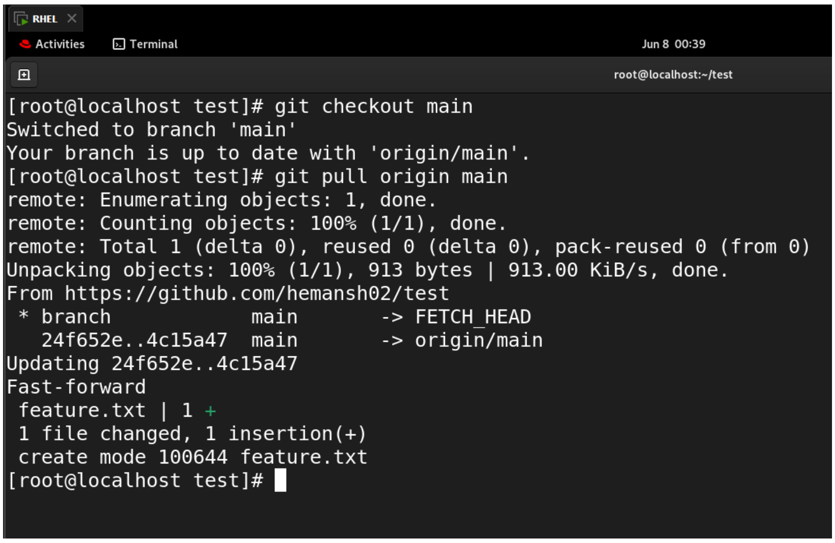
You can also merge this with the command line. View command line instructions.



Pull request successfully merged and closed

You're all set — the `feature-branch` branch can be safely deleted.

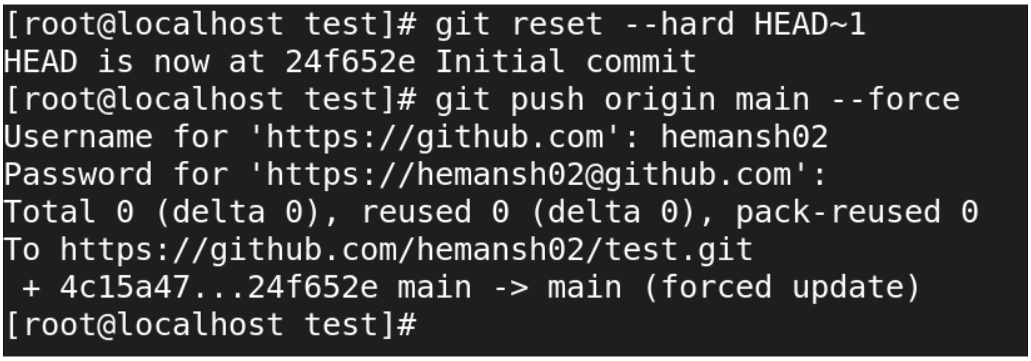
5. Update Local main Branch



```
[root@localhost test]# git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.
[root@localhost test]# git pull origin main
remote: Enumerating objects: 1, done.
remote: Counting objects: 100% (1/1), done.
remote: Total 1 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (1/1), 913 bytes | 913.00 KiB/s, done.
From https://github.com/hemansh02/test
 * branch            main       -> FETCH_HEAD
   24f652e..4c15a47  main       -> origin/main
Updating 24f652e..4c15a47
Fast-forward
 feature.txt | 1 +
 1 file changed, 1 insertion(+)
 create mode 100644 feature.txt
[root@localhost test]# 
```

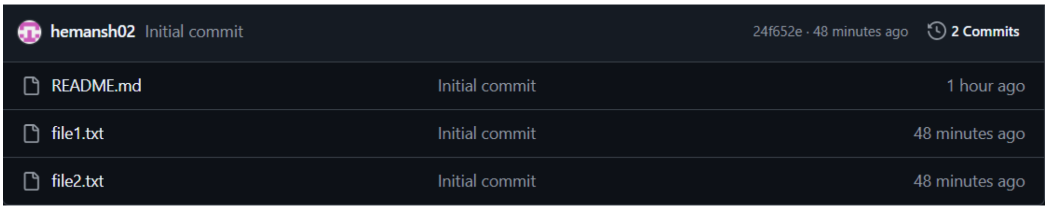
❖ Undo the last commit or remove the last created file from remote repo using CLI

1. Undo the last commit



```
[root@localhost test]# git reset --hard HEAD~1
HEAD is now at 24f652e Initial commit
[root@localhost test]# git push origin main --force
Username for 'https://github.com': hemansh02
Password for 'https://hemansh02@github.com':
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/hemansh02/test.git
 + 4c15a47...24f652e main -> main (forced update)
[root@localhost test]#
```

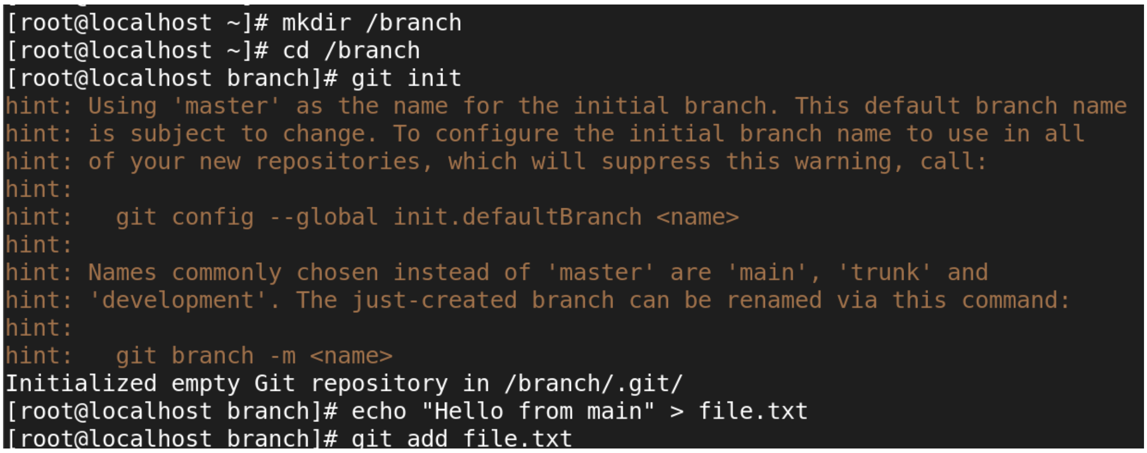
2. Verify in Github repo



hemansh02 Initial commit		24f652e · 48 minutes ago · 2 Commits
README.md	Initial commit	1 hour ago
file1.txt	Initial commit	48 minutes ago
file2.txt	Initial commit	48 minutes ago

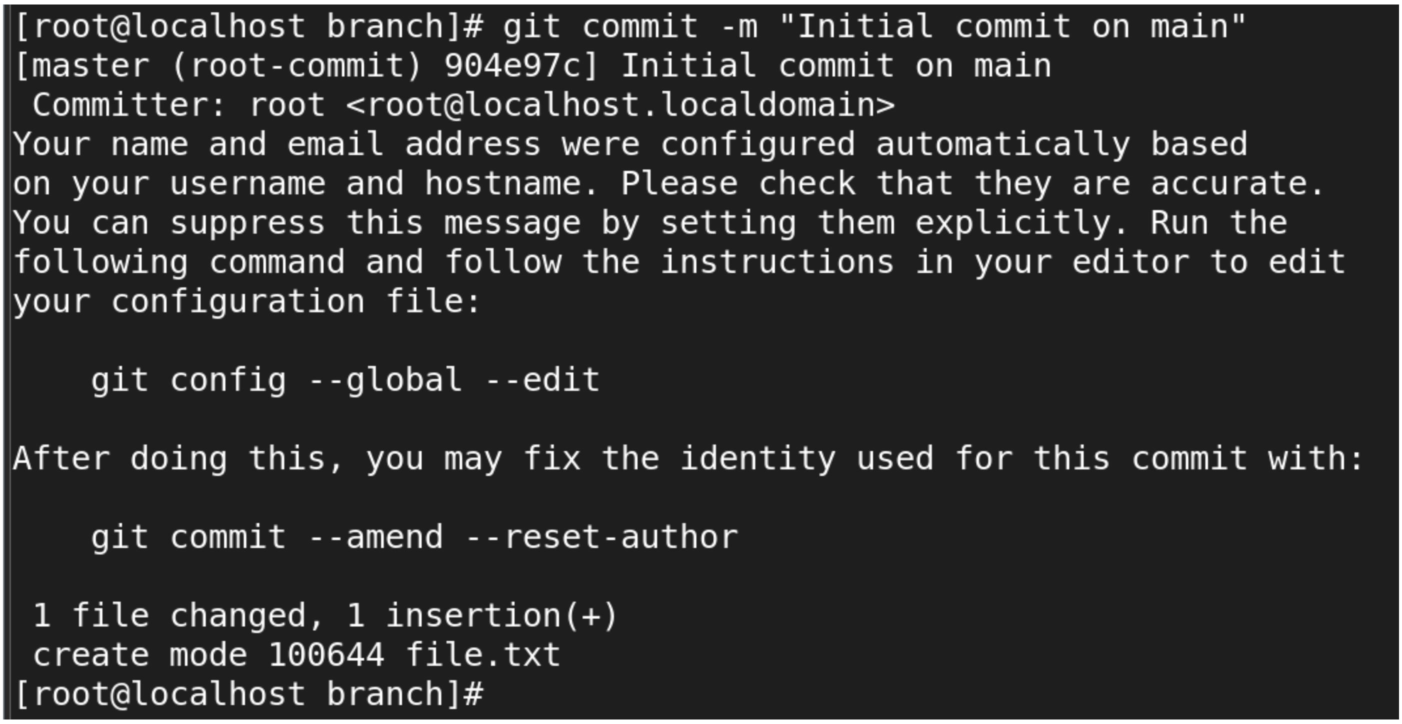
❖ Branching and merging in Git, Resolve a merge conflicts, and Practice More git commands

1. Initialize a Git repo, and Create a file on main



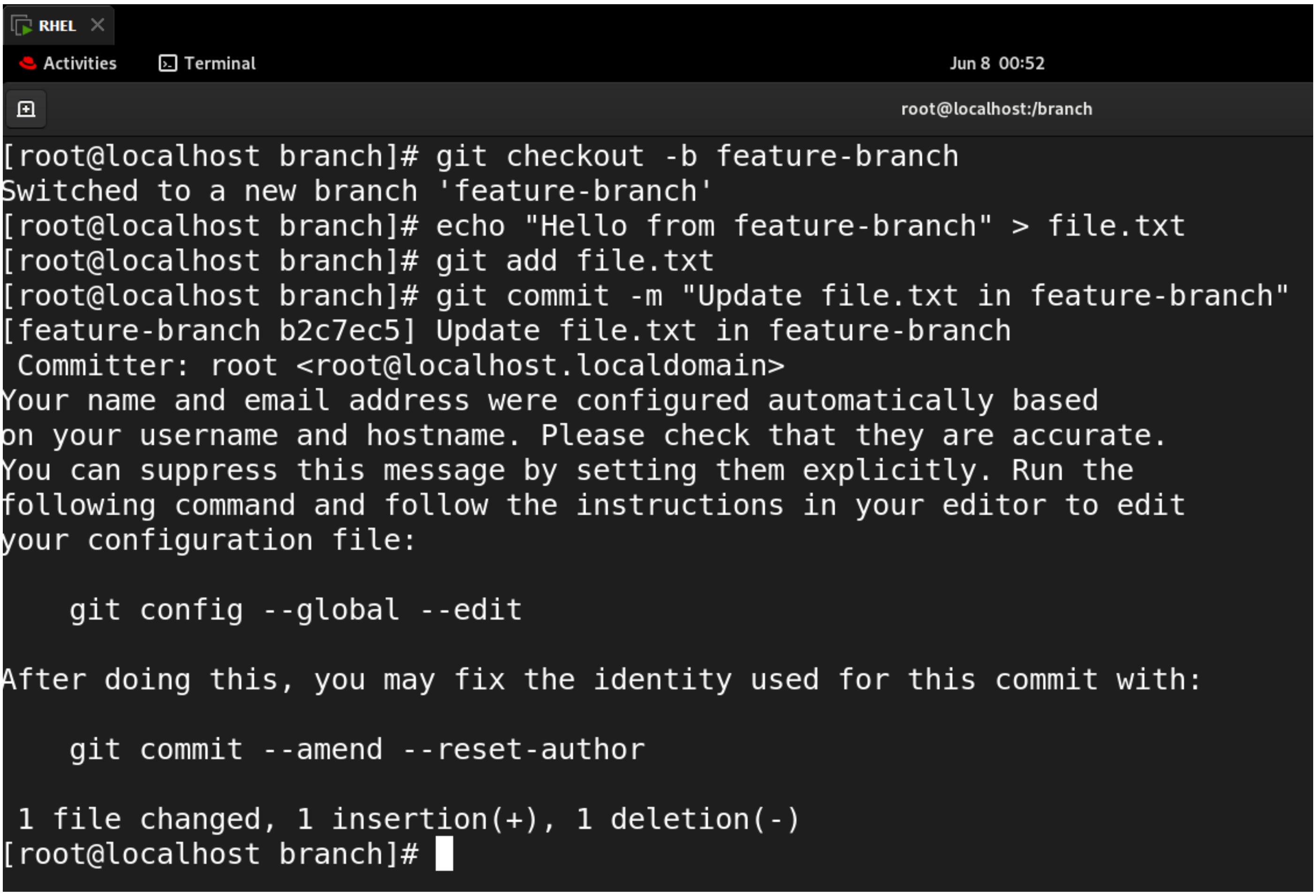
```
[root@localhost ~]# mkdir /branch
[root@localhost ~]# cd /branch
[root@localhost branch]# git init
hint: Using 'master' as the name for the initial branch. This default branch name
hint: is subject to change. To configure the initial branch name to use in all
hint: of your new repositories, which 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 /branch/.git/
[root@localhost branch]# echo "Hello from main" > file.txt
[root@localhost branch]# git add file.txt
```

2. Commit the files



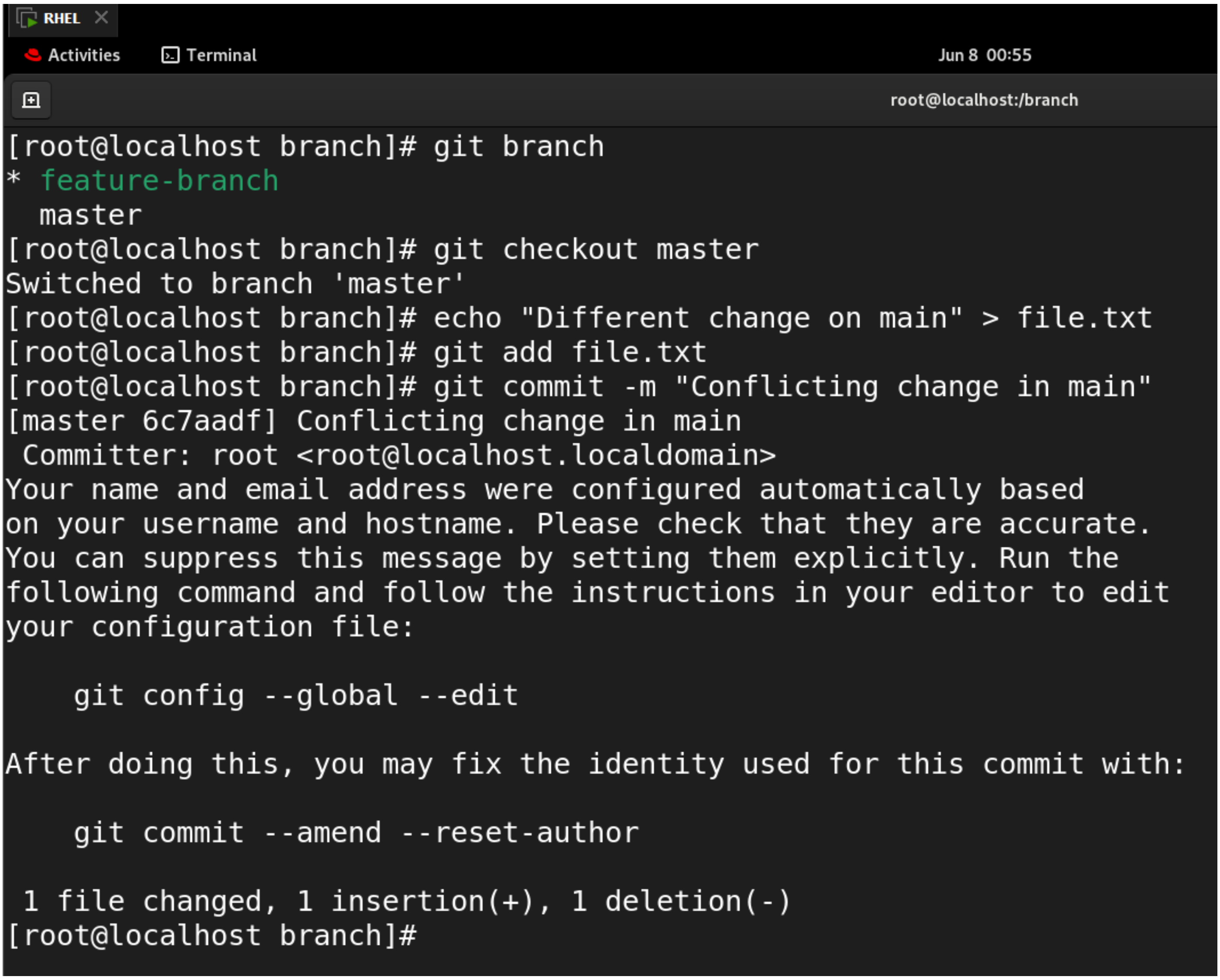
```
[root@localhost branch]# git commit -m "Initial commit on main"
[master (root-commit) 904e97c] Initial commit on main
 Committer: root <root@localhost.localdomain>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You 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(+)
 create mode 100644 file.txt
[root@localhost branch]#
```

3. Create a new branch and made changes



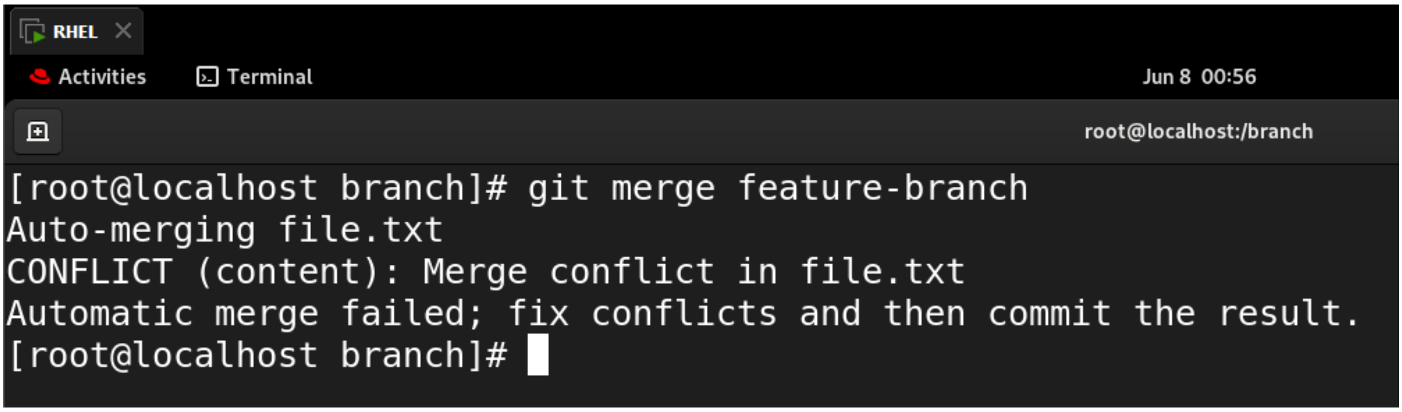
```
[root@localhost branch]# git checkout -b feature-branch
Switched to a new branch 'feature-branch'
[root@localhost branch]# echo "Hello from feature-branch" > file.txt
[root@localhost branch]# git add file.txt
[root@localhost branch]# git commit -m "Update file.txt in feature-branch"
[feature-branch b2c7ec5] Update file.txt in feature-branch
 Committer: root <root@localhost.localdomain>
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(+), 1 deletion(-)
[root@localhost branch]#
```

4. Switch back to main and made conflicting change



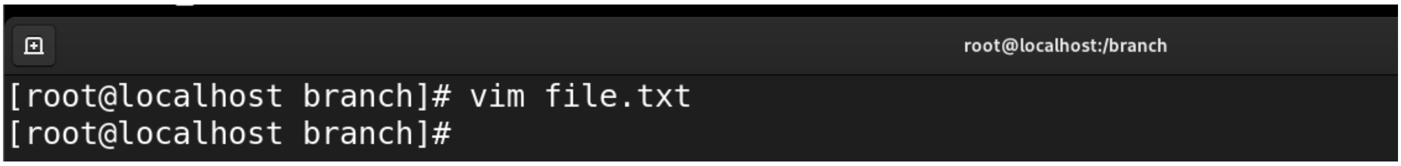
```
[root@localhost branch]# git branch
* feature-branch
  master
[root@localhost branch]# git checkout master
Switched to branch 'master'
[root@localhost branch]# echo "Different change on main" > file.txt
[root@localhost branch]# git add file.txt
[root@localhost branch]# git commit -m "Conflicting change in main"
[master 6c7aadf] Conflicting change in main
 Committer: root <root@localhost.localdomain>
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(+), 1 deletion(-)
[root@localhost branch]#
```

5. Try merging the feature-branch into main which gives merge conflict

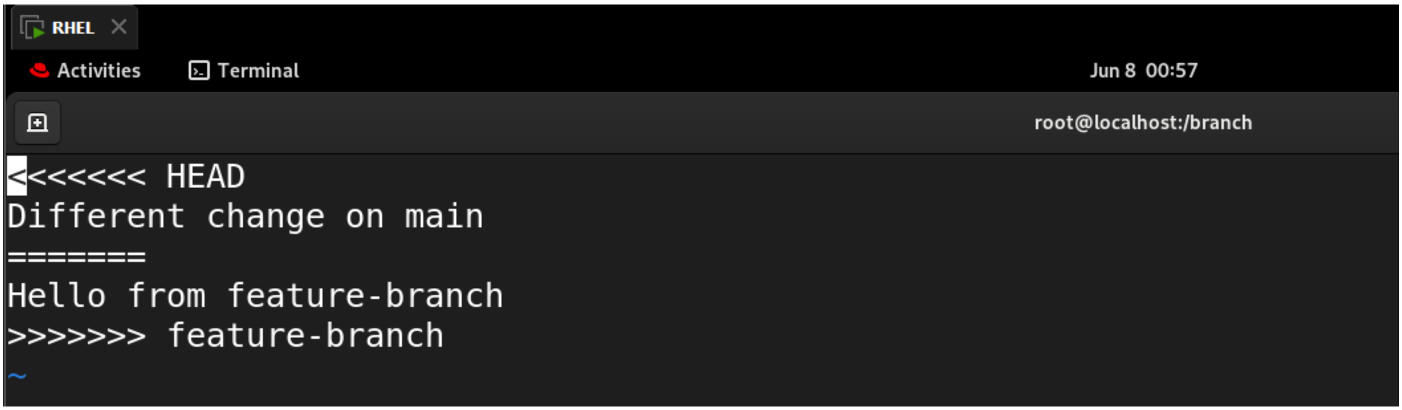


```
[root@localhost branch]# git merge feature-branch
Auto-merging file.txt
CONFLICT (content): Merge conflict in file.txt
Automatic merge failed; fix conflicts and then commit the result.
[root@localhost branch]# 
```

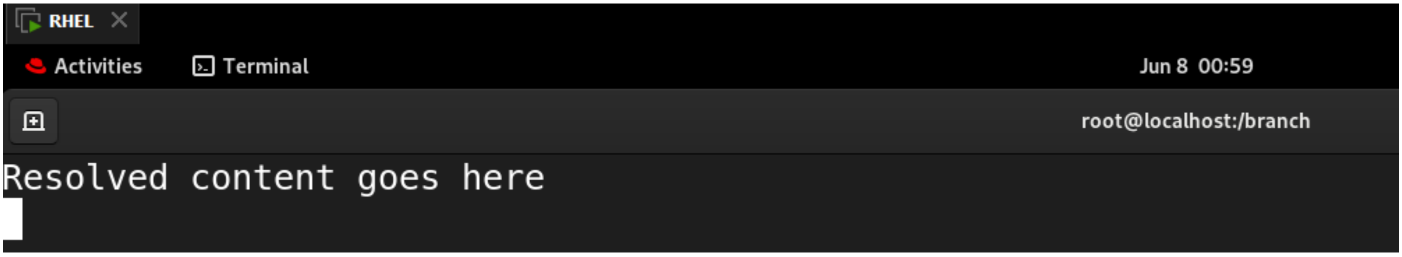
6. Resolve the conflict, open file.txt and rewrite it.



```
[root@localhost branch]# vim file.txt
[root@localhost branch]#
```

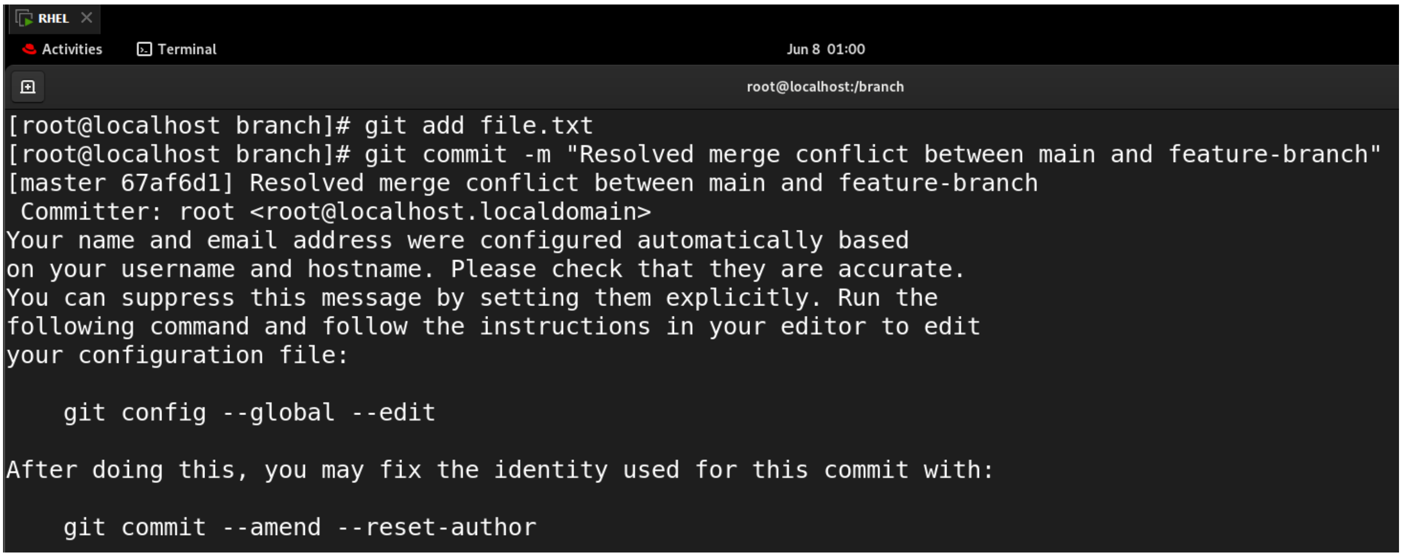


```
<<<<<<< HEAD
Different change on main
=======
Hello from feature-branch
>>>>>>> feature-branch
~
```



```
Resolved content goes here
```

7. Resolved conflict and now commit the merge



```
[root@localhost branch]# git add file.txt
[root@localhost branch]# git commit -m "Resolved merge conflict between main and feature-branch"
[master 67af6d1] Resolved merge conflict between main and feature-branch
 Committer: root <root@localhost.localdomain>
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
```

THANK YOU!