

GIT Assignment

Distributed version control system (DVCS)

1. Create a Distributed version control system, and upload files into it.

Step 1 - On the github page , click on the New Repository button. A form page will appear asking for some information


github.com/new

Create a new repository

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

Owner *


Repository name *


 q-madhuri ▾

/ Assignment ✓

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

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.

Initialize this repository with:
Skip this step if you're importing an existing repository.

☒ **Add a README file**
This is where you can write a long description for your project. [Learn more.](#)

Add .gitignore
Choose which files not to track from a list of templates. [Learn more.](#)

.gitignore template: None ▾

Choose a license
A license tells others what they can and can't do with your code. [Learn more.](#)

License: None ▾

Step 2 - fill the details and click on the create repository button.

Step 3 - create/upload files with the help of add file button.

github.com/q-madhuri/Assignment

Search or jump to...

Pull requests

Issues

Codespaces

Marketplace

Explore

🔔

+

👤

q-madhuri / Assignment

Public

Pin

Unwatch 1

Fork 0

Star 0

<> Code

Issues

Pull requests

Actions

Projects

Wiki

Security

Insights

Settings

main

1 branch

0 tags

Go to file

Add file

<> Code

Create new file

Upload files

3 commits

q-madhuri

Add files via upload

README.md	Initial commit	3 minutes ago
sample.txt	Create sample.txt	2 minutes ago
test.xlsx	Add files via upload	now

README.md

Assignment

About

No description, website, or topics provided.

Readme

0 stars

1 watching

0 forks

Releases

No releases published

Create a new release

Packages

No packages published

Publish your first package

https://github.com/q-madhuri/Assignment/upload/main

Terms

Privacy

Security

Status

Docs

Contact GitHub

Pricing

API

Training

Blog

About

2.Replicate a DVCS on your local machine, make changes to its files, and then put them back into it.

Step1 - git clone "<https://github.com/madhuri2604/Git-Assignment.git>"

Step 2 - make changes in the file sample.txt

Step 3 - git add sample

Step 4 - git commit -m "Modifying the file"

Step 5 - git push origin main

```
91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git (master)
$ git clone https://github.com/madhuri2604/Git-Assignment.git
Cloning into 'Git-Assignment'...
remote: Enumerating objects: 9, done.
remote: Counting objects: 100% (9/9), done.
remote: Compressing objects: 100% (6/6), done.
remote: Total 9 (delta 1), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (9/9), 6.95 KiB | 6.96 MiB/s, done.
Resolving deltas: 100% (1/1), done.

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git (master)
$ ls
Git-Assignment/

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git (master)
$ cd Git-Assignment/

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (main)
$ ls
README.md sample test.xlsx

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean

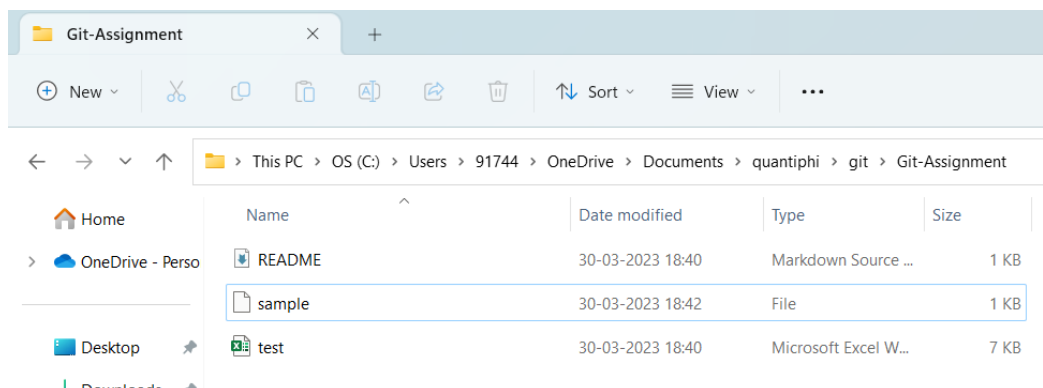
91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

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:   sample

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

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (main)
$ git add sample

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (main)
$ git commit -m "Modifying the file"
[main 5359154] Modifying the file
1 file changed, 1 insertion(+)
```




```
MINGW64;c:/Users/91744/OneDrive/Documents/quantiphi/git/Git-Assignment
91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (main)
$ git commit -m "Modifying the file"
[main 5359154] Modifying the file
1 file changed, 1 insertion(+)

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (main)
$ git pull origin main
From https://github.com/madhuri2604/Git-Assignment
* branch      main      -> FETCH_HEAD
Already up to date.


91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (main)
$ git push origin main
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 8 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
To https://github.com/madhuri2604/Git-Assignment.git
6502bbd..5359154  main -> main


91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (main)
$
```


 Search or jump to... [Pull requests](#) [Issues](#) [Codespaces](#) [Marketplace](#) [Explore](#)

[madhuri2604 / Git-Assignment](#) Public [Pin](#) [Unwatch](#) [Fork](#) [1](#) [0](#)


[Code](#) [Issues](#) [Pull requests](#) [Actions](#) [Projects](#) [Wiki](#) [Security](#) [Insights](#) [Settings](#)

 main




 1 branch

 0 tags

[Go to file](#) [Add file](#) [Code](#)

 madhuri2604 Modifying the file

5359154 4 minutes ago 4 commits





 README.md	Initial commit	16 minutes ago
 sample	Modifying the file	4 minutes ago
 test.xlsx	Add files via upload	15 minutes ago

README.md

Git-Assignment

About

No description, website, or topics provided.

-  README
-  0 stars
-  1 watching
-  0 forks

Releases


No releases published

[Create a new release](#)

Packages

No packages published

[Publish your first package](#)

 © 2023 GitHub, Inc.

[Terms](#) [Privacy](#) [Security](#) [Status](#) [Docs](#) [Contact GitHub](#) [Pricing](#) [API](#) [Training](#) [Blog](#) [About](#)

3. List down all the files which have changed in the last commit. List only the file names.

Step 1 - use git log command to see the commits.

Step 2 - `git show --name-only`

```
MINGW64/c/Users/91744/OneDrive/Documents/quantiphi/git/Git-Assignment
91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (main)
$ git log
commit 5359154566678f8bfc7c503d8c56bdfd0791ec7e (HEAD -> main, origin/main, origin/HEAD)
Author: madhuri2604 <madhurimalviya2601@gmail.com>
Date: Thu Mar 30 18:42:55 2023 +0530

    Modifying the file

commit 6502bbdba40dd4efced6778793790aa341313d6b
Author: madhuri2604 <64089279+madhuri2604@users.noreply.github.com>
Date: Thu Mar 30 18:31:51 2023 +0530

    Add files via upload

commit e9f54a2f8b577f4b460ad885b02ab1af332fd42e
Author: madhuri2604 <64089279+madhuri2604@users.noreply.github.com>
Date: Thu Mar 30 18:31:18 2023 +0530

    Create sample

commit d7dc507312e4ecdafea271f027a624d105c9f771
Author: madhuri2604 <64089279+madhuri2604@users.noreply.github.com>
Date: Thu Mar 30 18:31:00 2023 +0530

    Initial commit

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (main)
$
```

```
91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (main)
$ git show --name-only
commit 5359154566678f8bfc7c503d8c56bdfd0791ec7e (HEAD -> main, origin/main, origin/HEAD)
Author: madhuri2604 <madhurimalviya2601@gmail.com>
Date: Thu Mar 30 18:42:55 2023 +0530

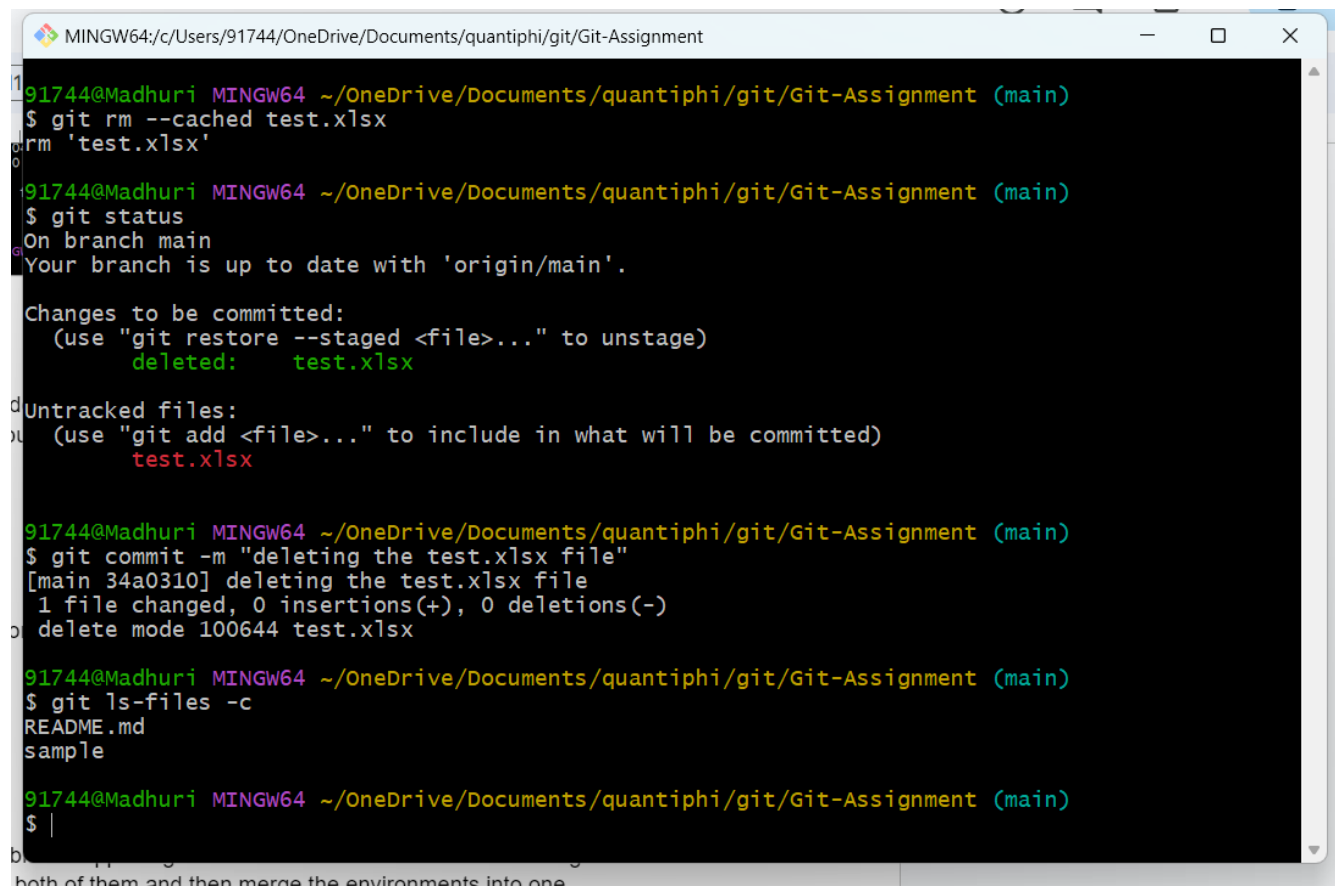
    Modifying the file

sample

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (main)
$
```

4. You've accidentally added some files on a DVCS. Remove those files from the DVCS without deleting them from your local system.

Step 1- Lets add files in it -(test.xlsx) and check the status.



```
MINGW64~/c/Users/91744/OneDrive/Documents/quantiphi/git/Git-Assignment
91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (main)
$ git rm --cached test.xlsx
rm 'test.xlsx'

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        deleted:    test.xlsx

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

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (main)
$ git commit -m "deleting the test.xlsx file"
[main 34a0310] deleting the test.xlsx file
 1 file changed, 0 insertions(+), 0 deletions(-)
 delete mode 100644 test.xlsx

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (main)
$ git ls-files -c
README.md
sample

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (main)
$ |
```

5. You've accidentally committed some files on a DVCS. Revert the DVCS to a previous stable state.

Step 1 : `git revert <commit_ID>`

The `commit_id` is visible in logs

```
commit 5359154566678f8bfc7c503d8c56bdfd0791ec7e (origin/main, origin/HEAD)
Author: madhuri2604 <madhurimalviya2601@gmail.com>
Date: Thu Mar 30 18:42:55 2023 +0530

    Modifying the file

commit 6502bbdba40dd4efced6778793790aa341313d6b
Author: madhuri2604 <64089279+madhuri2604@users.noreply.github.com>
Date: Thu Mar 30 18:31:51 2023 +0530

    Add files via upload

commit e9f54a2f8b577f4b460ad885b02ab1af332fd42e
Author: madhuri2604 <64089279+madhuri2604@users.noreply.github.com>
Date: Thu Mar 30 18:31:18 2023 +0530

    Create sample

commit d7dc507312e4ecdafea271f027a624d105c9f771
Author: madhuri2604 <64089279+madhuri2604@users.noreply.github.com>
Date: Thu Mar 30 18:31:00 2023 +0530

    Initial commit

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (main)
$ git revert ^C

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (main)
$ git revert 5359154566678f8bfc7c503d8c56bdfd0791ec7e
On branch main
Your branch is ahead of 'origin/main' by 2 commits.
(use "git push" to publish your local commits)

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

nothing added to commit but untracked files present (use "git add" to track)

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (main)
$
```

```
MINGW64/c/Users/91744/OneDrive/Documents/quantiphi/git/Git-Assignment

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (main)
$ git log
commit 7231cd535641bcd6575277a0d10f1c9791693244 (HEAD -> main)
Author: madhuri2604 <madhurimalviya2601@gmail.com>
Date: Thu Mar 30 19:17:45 2023 +0530

    Revert "Modifying the file"

    This reverts commit 5359154566678f8bfc7c503d8c56bdfd0791ec7e.
```

6. Create a DVCS capable of supporting 2 similar dev environments. Make changes in the files which are present on both of them and then merge the environments into one.

Step 1 - create second environment(branch)

Step 2 - make some changes to the sample file from newbranch environment & commit it

```
MINGW64:/c/Users/91744/OneDrive/Documents/quantiphi/git/Git-Assignment

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (main)
$ git branch -a
* main
  remotes/origin/HEAD -> origin/main
  remotes/origin/main

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (main)
$ git branch newbranch

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (main)
$ git checkout newbranch
error: pathspec 'branch' did not match any file(s) known to git

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (main)
$ git checkout newbranch
Switched to branch 'newbranch'

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (newbranch)
$ git log --oneline --decorate
7231cd5 (HEAD -> newbranch, main) Revert "Modifying the file"
34a0310 deleting the test.xlsx file
5359154 (origin/main, origin/HEAD) Modifying the file
6502bbd Add files via upload
e9f54a2 Create sample
d7dc507 Initial commit

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (newbranch)
$ vim sample

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (newbranch)
$ git status
On branch newbranch
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:   sample

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

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

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (newbranch)
$ git add sample

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (newbranch)
$ git commit -m "changing from newbranch"
```

Step 3 - go to main branch & modify, commit the sample file there

Step 4 - merge the branches

Git merge <branch-name>

```
MINGW64/c/Users/91744/OneDrive/Documents/quantiphi/git/Git-Assignment
(use "git add <file>..." to update what will be committed)
(use "git restore <file>..." to discard changes in working directory)
    modified:   sample

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

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

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (main)
$ git add sample

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (main)
$ git commit -m "modifying from main"
[main b0aaa45] modifying from main
1 file changed, 2 insertions(+)

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (main)
$ git diff main newbranch
diff --git a/sample b/sample
index e256c76..2feba1c 100644
--- a/sample
+++ b/sample
@@ -1,3 +1,3 @@
-
-new change done by main branch
+this is a new file
+making some changes
+

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (main)
$ git merge newbranch
Auto-merging sample
Merge made by the 'ort' strategy.
 sample | 2 ++
 1 file changed, 2 insertions(+)

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (main)
$ git log --oneline --graph --decorate
* 145e99c (HEAD -> main) Merge branch 'newbranch'
| \
|  * 18e382e (newbranch) changing from newbranch
|  * | b0aaa45 modifying from main
| /
* 7231cd5 Revert "Modifying the file"
* 34a0310 deleting the test.xlsx file
* 5359154 (origin/main, origin/HEAD) Modifying the file
* 6502bbd Add files via upload
```

7. You have a DVCS supporting your prod, test and dev environments, you need to make sure that only a particular set of commits from the test environment are added in the prod environment. What & how would you do it?

Step 1 - create 3 env = prod, dev, test

Step 2 - in test env , add and commit files.

```
MINGW64:/c/Users/91744/OneDrive/Documents/quantiphi/git/Git-Assignment

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (main)
$ git branch test

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (main)
$ git branch dev

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (main)
$ git branch prod

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (main)
$ git checkout test
Switched to branch 'test'

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (test)
$ touch file1.txt

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (test)
$ git add fle1.txt
fatal: pathspec 'fle1.txt' did not match any files

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (test)
$ git add file1.txt

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (test)
$ git commit -m "1st commit"
[test c10de73] 1st commit
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 file1.txt

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (test)
$ touch file2.txt

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (test)
$ git add file2.txt

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (test)
$ git commit -m "2nd commit"
[test 23cald9] 2nd commit
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 file2.txt
```

```

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (test)
$ git log
commit 23ca1d93e8a29cf8a0a98182cbdc6c1103d4e982 (HEAD -> test)
Author: madhuri2604 <madhurimalviya2601@gmail.com>
Date: Thu Mar 30 22:36:11 2023 +0530

    2nd commit

commit c10de738dc3c9737add629e129ad33e1c2182e0f
Author: madhuri2604 <madhurimalviya2601@gmail.com>
Date: Thu Mar 30 22:35:50 2023 +0530

    1st commit

commit 145e99c72402326c0444048bf12c8e32005a5a43 (prod, main, dev)
Merge: b0aaa45 18e382e
Author: madhuri2604 <madhurimalviya2601@gmail.com>
Date: Thu Mar 30 21:05:59 2023 +0530

    Merge branch 'newbranch'

```

Step 3 - now go to prod env and use cherry-pick command to select the commits you want to add to prod env

```
git cherry-pick <commit-ids>
```

Step 4- we only added 1st commit from test env therefore prod env does not contain file2.txt

```

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (test)
$ git checkout prod
Switched to branch 'prod'

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (prod)
$ git cherry-pick ^C

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (prod)
$ git cherry-pick c10de738dc3c9737add629e129ad33e1c2182e0f
[prod 3490a18] 1st commit
Date: Thu Mar 30 22:35:50 2023 +0530
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 file1.txt

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (prod)
$ git commit -m "Only first commit from test env added to prov env"
On branch prod
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    test.xlsx

nothing added to commit but untracked files present (use "git add" to track)

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (prod)
$ ls
README.md  file1.txt  sample  test.xlsx

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (prod)
$ git checkout test
Switched to branch 'test'

91744@Madhuri MINGW64 ~/OneDrive/Documents/quantiphi/git/Git-Assignment (test)
$ ls
README.md  file1.txt  file2.txt  sample  test.xlsx

```

Scenario Questions:

1.If the central repository is two commits ahead of your local repository, how will you push your code ?

First we need to pull the changes(`git pull`) from the remote repository to local repository and resolve any conflicts that may arise during the merge ,

`git commit`

Commit the changes to the local repository.

and then push the code `git push`

2.I have two branches, A and B, where A is my main branch, and B is a the branch with new feature, describe the step by step process getting all of B's code in A

Step 1 - In B branch , stage and commit the files

Step 2 - checkout to branch A

Step 3 - merge branch B with A

`git merge B`

All the changes in B branch will merge into A

3.Say you have to do an experimental change but still want to have a clean copy of your old code, should you use branching ? Explain your decision

Yes definitely , branching is the best choice to keep a clean copy of your code.Branching allows you to create a separate copy of your code where you can try and experiment on without affecting the main codebase.

It also helps in collaborating with others.