1. **Creating and Managing Branches**

**Write the commands to stash your changes, switch branches, and then apply the stashed changes.**

**SOLUTION:**

1. Open your terminal and navigate to the directory where you want to create the Git repository.

2. Initialize a new Git repository in that directory:

$ git init

3. Check the status of the created Git repository.

$git status

5. Create an empty file in the “master” branch.

touch “filename”

6. Add the created file to the git repository.

$git add .

7. Commit the created file

$git commit –m “file added”

8.Create a new branch

$git branch branchname

9. Switch to new branch

$git checkout branchname

10.Create some new files inside the new branch

touch “filename”

11. Add the created file to the git repository.

$git add .

12. Commit the created file

$git commit –m “file added”

12. Switch to the master branch

$git checkout master

13. Check the status

$git status

14. Switch back to new branch

$git checkout branchname

15.Insert some content to the file

Vi “file name”

16.Don’t commit the file instead switch back to master

$git checkout master

**17. Stash Your Changes**

Before switching branches, it's important to save any uncommitted changes. This is where the git stash command comes in handy.

$git stash

18. Switch Branches

After stashing your changes, you can safely switch to another branch.

$git checkout <branch-name>

19. Apply the Stashed Changes

Now that you’ve switched to the desired branch, you can apply the changes you stashed earlier.

$git stash apply

TEJAKSHI M.S@user MINGW64 ~/Desktop/SDIT/PROJECT MANAGEMENT USING GIT/exp3 (master)

$ git init

Reinitialized existing Git repository in C:/Users/TEJAKSHI M.S/Desktop/SDIT/PROJECT MANAGEMENT USING GIT/exp3/.git/

TEJAKSHI M.S@user MINGW64 ~/Desktop/SDIT/PROJECT MANAGEMENT USING GIT/exp3 (master)

$ git status

On branch master

Changes to be committed:

(use "git restore --staged <file>..." to unstage)

new file: t.txt

TEJAKSHI M.S@user MINGW64 ~/Desktop/SDIT/PROJECT MANAGEMENT USING GIT/exp3 (master)

$ git add .

TEJAKSHI M.S@user MINGW64 ~/Desktop/SDIT/PROJECT MANAGEMENT USING GIT/exp3 (master)

$ git status

On branch master

Changes to be committed:

(use "git restore --staged <file>..." to unstage)

new file: t.txt

TEJAKSHI M.S@user MINGW64 ~/Desktop/SDIT/PROJECT MANAGEMENT USING GIT/exp3 (master)

$ touch "tej.txt"

TEJAKSHI M.S@user MINGW64 ~/Desktop/SDIT/PROJECT MANAGEMENT USING GIT/exp3 (master)

$ vi tej.txt

TEJAKSHI M.S@user MINGW64 ~/Desktop/SDIT/PROJECT MANAGEMENT USING GIT/exp3 (master)

$ git add .

warning: in the working copy of 'tej.txt', LF will be replaced by CRLF the next time Git touches it

TEJAKSHI M.S@user MINGW64 ~/Desktop/SDIT/PROJECT MANAGEMENT USING GIT/exp3 (master)

$ git commit -m "file added"

[master ff4277e] file added

2 files changed, 4 insertions(+)

create mode 100644 t.txt

create mode 100644 tej.txt

TEJAKSHI M.S@user MINGW64 ~/Desktop/SDIT/PROJECT MANAGEMENT USING GIT/exp3 (master)

$ git status

On branch master

nothing to commit, working tree clean

TEJAKSHI M.S@user MINGW64 ~/Desktop/SDIT/PROJECT MANAGEMENT USING GIT/exp3 (master)

$ git branch sdit

TEJAKSHI M.S@user MINGW64 ~/Desktop/SDIT/PROJECT MANAGEMENT USING GIT/exp3 (master)

$ git checkout sdit

Switched to branch 'sdit'

TEJAKSHI M.S@user MINGW64 ~/Desktop/SDIT/PROJECT MANAGEMENT USING GIT/exp3 (sdit)

$ touch "ns.txt"

TEJAKSHI M.S@user MINGW64 ~/Desktop/SDIT/PROJECT MANAGEMENT USING GIT/exp3 (sdit)

$ vi ns.txt

TEJAKSHI M.S@user MINGW64 ~/Desktop/SDIT/PROJECT MANAGEMENT USING GIT/exp3 (sdit)

$ git commit -m "file added"

On branch sdit

Untracked files:

(use "git add <file>..." to include in what will be committed)

ns.txt

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

TEJAKSHI M.S@user MINGW64 ~/Desktop/SDIT/PROJECT MANAGEMENT USING GIT/exp3 (sdit)

$ git add .

warning: in the working copy of 'ns.txt', LF will be replaced by CRLF the next time Git touches it

TEJAKSHI M.S@user MINGW64 ~/Desktop/SDIT/PROJECT MANAGEMENT USING GIT/exp3 (sdit)

$ git commit -m "file added"

[sdit 4688f7e] file added

1 file changed, 2 insertions(+)

create mode 100644 ns.txt

TEJAKSHI M.S@user MINGW64 ~/Desktop/SDIT/PROJECT MANAGEMENT USING GIT/exp3 (sdit)

$ git status

On branch sdit

nothing to commit, working tree clean

TEJAKSHI M.S@user MINGW64 ~/Desktop/SDIT/PROJECT MANAGEMENT USING GIT/exp3 (sdit)

$ git checkout master

Switched to branch 'master'

TEJAKSHI M.S@user MINGW64 ~/Desktop/SDIT/PROJECT MANAGEMENT USING GIT/exp3 (master)

$ git stash

No local changes to save

TEJAKSHI M.S@user MINGW64 ~/Desktop/SDIT/PROJECT MANAGEMENT USING GIT/exp3 (master)

$ git checkout sdit

Switched to branch 'sdit'

TEJAKSHI M.S@user MINGW64 ~/Desktop/SDIT/PROJECT MANAGEMENT USING GIT/exp3 (sdit)

$ vi ns.txt

TEJAKSHI M.S@user MINGW64 ~/Desktop/SDIT/PROJECT MANAGEMENT USING GIT/exp3 (sdit)

$ git checkout master

error: Your local changes to the following files would be overwritten by checkout:

ns.txt

Please commit your changes or stash them before you switch branches.

Aborting

TEJAKSHI M.S@user MINGW64 ~/Desktop/SDIT/PROJECT MANAGEMENT USING GIT/exp3 (sdit)

$ git stash

Saved working directory and index state WIP on sdit: 4688f7e file added

TEJAKSHI M.S@user MINGW64 ~/Desktop/SDIT/PROJECT MANAGEMENT USING GIT/exp3 (sdit)

$ git checkout master

Switched to branch 'master'

TEJAKSHI M.S@user MINGW64 ~/Desktop/SDIT/PROJECT MANAGEMENT USING GIT/exp3 (master)

$