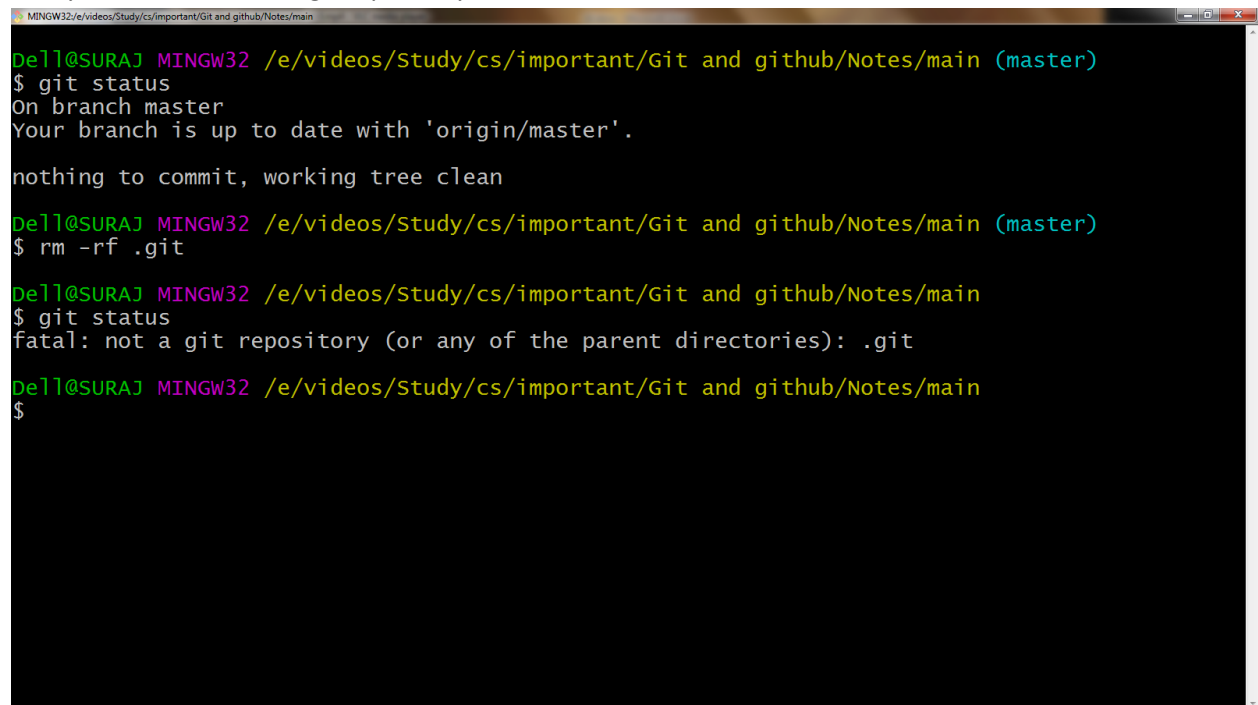


Firstly delete the current git repository

A terminal window with a black background and white text. The window title is 'MINGW32/e/videos/Study/cs/important/Git and github/Notes/main'. The user 'Dell@SURAJ' is in the 'MINGW32' shell at the directory '/e/videos/Study/cs/important/Git and github/Notes/main' on the 'master' branch. They run '\$ git status', which shows they are on the master branch and up to date with 'origin/master', with nothing to commit. Then they run '\$ rm -rf .git', which successfully removes the repository. Finally, they run '\$ git status' again, which results in a 'fatal: not a git repository (or any of the parent directories): .git' error.

```
Dell@SURAJ MINGW32 /e/videos/Study/cs/important/Git and github/Notes/main (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean

Dell@SURAJ MINGW32 /e/videos/Study/cs/important/Git and github/Notes/main (master)
$ rm -rf .git

Dell@SURAJ MINGW32 /e/videos/Study/cs/important/Git and github/Notes/main
$ git status
fatal: not a git repository (or any of the parent directories): .git

Dell@SURAJ MINGW32 /e/videos/Study/cs/important/Git and github/Notes/main
$
```

Open learnC folder in VS Code

Open git bash from learnC folder & initialize git repository

And then add files to staging area and commit the changes

```
MINGW32/e/videos/Study/cs/important/Git and github/Notes/main/learnC
Dell@SURAJ MINGW32 /e/videos/Study/cs/important/Git and github/Notes/main/learnC (master)
$ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    Appendix/
    Chapter01/
    Chapter02/
    Chapter03/
    Chapter04/
    Chapter05/
    Chapter06/
    Chapter07/
    Chapter08/
    Chapter09/
    Chapter10/
    Chapter11/
    Chapter12/
    Chapter13/
    Chapter14/
    Chapter15/
    README.md

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

Dell@SURAJ MINGW32 /e/videos/Study/cs/important/Git and github/Notes/main/learnC (master)
$ git add .

Dell@SURAJ MINGW32 /e/videos/Study/cs/important/Git and github/Notes/main/learnC (master)
$ git commit -m "Initial Commit"
[master (root-commit) 5f9291a] Initial Commit
126 files changed, 10566 insertions(+)
create mode 100644 Appendix/bstr_hello.c
create mode 100644 Appendix/bstr_split.c
create mode 100644 Appendix/bstr_trim.c
create mode 100644 Chapter01/comments.c
create mode 100644 Chapter01/hello1.c
create mode 100644 Chapter01/hello2.c
create mode 100644 Chapter02/hello.c
create mode 100644 Chapter02/hello2.c
create mode 100644 Chapter02/hello3.c
create mode 100644 Chapter02/hello4.c
create mode 100644 Chapter02/hello5.c
create mode 100644 Chapter02/hello6.c
create mode 100644 Chapter02/hello7.c
```

Make new branch from here

```
MINGW32/e/videos/Study/cs/important/Git and github/Notes/main/learnC
Dell@SURAJ MINGW32 /e/videos/Study/cs/important/Git and github/Notes/main/learnC (master)
$ git checkout -b newFeature
Switched to a new branch 'newFeature'

Dell@SURAJ MINGW32 /e/videos/Study/cs/important/Git and github/Notes/main/learnC (newFeature)
$
```

Change some files. And then stage and commit changes in it.

```
MINGW32/e/videos/Study/cs/important/Git and github/Notes/main/learnC
$ git checkout -b newFeature
Switched to a new branch 'newFeature'

Dell@SURAJ MINGW32 /e/videos/Study/cs/important/Git and github/Notes/main/learnC (newFeature)
$ git status
On branch newFeature
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:   Chapter01/hello1.c

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

Dell@SURAJ MINGW32 /e/videos/Study/cs/important/Git and github/Notes/main/learnC (newFeature)
$ git add .

Dell@SURAJ MINGW32 /e/videos/Study/cs/important/Git and github/Notes/main/learnC (newFeature)
$ git commit -m "changed C tutorial"
[newFeature f853075] changed C tutorial
1 file changed, 1 insertion(+)

Dell@SURAJ MINGW32 /e/videos/Study/cs/important/Git and github/Notes/main/learnC (newFeature)
$ git status
On branch newFeature
nothing to commit, working tree clean

Dell@SURAJ MINGW32 /e/videos/Study/cs/important/Git and github/Notes/main/learnC (newFeature)
$ |
```

And then switch to master branch again. This will revert changes as same as in master branch

```
MINGW32/e/videos/Study/cs/important/Git and github/Notes/main/learnC
$ git status
On branch newFeature
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:   Chapter01/hello1.c

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

Dell@SURAJ MINGW32 /e/videos/Study/cs/important/Git and github/Notes/main/learnC (newFeature)
$ git add .

Dell@SURAJ MINGW32 /e/videos/Study/cs/important/Git and github/Notes/main/learnC (newFeature)
$ git commit -m "changed C tutorial"
[newFeature f853075] changed C tutorial
1 file changed, 1 insertion(+)

Dell@SURAJ MINGW32 /e/videos/Study/cs/important/Git and github/Notes/main/learnC (newFeature)
$ git status
On branch newFeature
nothing to commit, working tree clean

Dell@SURAJ MINGW32 /e/videos/Study/cs/important/Git and github/Notes/main/learnC (newFeature)
$ git checkout master
Switched to branch 'master'

Dell@SURAJ MINGW32 /e/videos/Study/cs/important/Git and github/Notes/main/learnC (master)
$ |
```

Now change something and then stage and commit that changes and then again switch to newFeature branch

```
MINGW32/e/videos/Study/cs/important/Git and github/Notes/main/learnC
Dell@SURAJ MINGW32 /e/videos/Study/cs/important/Git and github/Notes/main/learnC (newFeature)
$ git checkout master
Switched to branch 'master'

Dell@SURAJ MINGW32 /e/videos/Study/cs/important/Git and github/Notes/main/learnC (master)
$ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   Chapter01/hello1.c

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

Dell@SURAJ MINGW32 /e/videos/Study/cs/important/Git and github/Notes/main/learnC (master)
$ git add .

Dell@SURAJ MINGW32 /e/videos/Study/cs/important/Git and github/Notes/main/learnC (master)
$ git commit -m "changes in master"
[master de0772b] changes in master
1 file changed, 1 insertion(+)

Dell@SURAJ MINGW32 /e/videos/Study/cs/important/Git and github/Notes/main/learnC (master)
$ git checkout newFeature
Switched to branch 'newFeature'

Dell@SURAJ MINGW32 /e/videos/Study/cs/important/Git and github/Notes/main/learnC (newFeature)
$ |
```

Now change again something here and the stage and commit that changes and then again switch to master branch or to skip staging area and directly commit type → “git commit -a -m “message””

```
MINGW32/e/videos/Study/cs/important/Git and github/Notes/main/learnC
Dell@SURAJ MINGW32 /e/videos/Study/cs/important/Git and github/Notes/main/learnC (master)
$ git checkout newFeature
Switched to branch 'newFeature'

Dell@SURAJ MINGW32 /e/videos/Study/cs/important/Git and github/Notes/main/learnC (newFeature)
$ git status
On branch newFeature
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:   Chapter01/hello1.c

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

Dell@SURAJ MINGW32 /e/videos/Study/cs/important/Git and github/Notes/main/learnC (newFeature)
$ git add .

Dell@SURAJ MINGW32 /e/videos/Study/cs/important/Git and github/Notes/main/learnC (newFeature)
$ git commit -m "new changes in newFeature"
[newFeature 7e39b34] new changes in newFeature
 1 file changed, 1 insertion(+), 1 deletion(-)

Dell@SURAJ MINGW32 /e/videos/Study/cs/important/Git and github/Notes/main/learnC (newFeature)
$ git checkout master
Switched to branch 'master'

Dell@SURAJ MINGW32 /e/videos/Study/cs/important/Git and github/Notes/main/learnC (master)
$ |
```

To show all branches → “git branch” and then go to newFeature branch

```
MINGW32/e/videos/Study/cs/important/Git and github/Notes/main/learnC
  (use "git restore <file>..." to discard changes in working directory)
    modified:   Chapter01/hello1.c

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

Dell@SURAJ MINGW32 /e/videos/Study/cs/important/Git and github/Notes/main/learnC (newFeature)
$ git add .

Dell@SURAJ MINGW32 /e/videos/Study/cs/important/Git and github/Notes/main/learnC (newFeature)
$ git commit -m "new changes in newFeature"
[newFeature 7e39b34] new changes in newFeature
 1 file changed, 1 insertion(+), 1 deletion(-)

Dell@SURAJ MINGW32 /e/videos/Study/cs/important/Git and github/Notes/main/learnC (newFeature)
$ git checkout master
Switched to branch 'master'

Dell@SURAJ MINGW32 /e/videos/Study/cs/important/Git and github/Notes/main/learnC (master)
$ git branch
* master
  newFeature

Dell@SURAJ MINGW32 /e/videos/Study/cs/important/Git and github/Notes/main/learnC (master)
$ git checkout newFeature
Switched to branch 'newFeature'

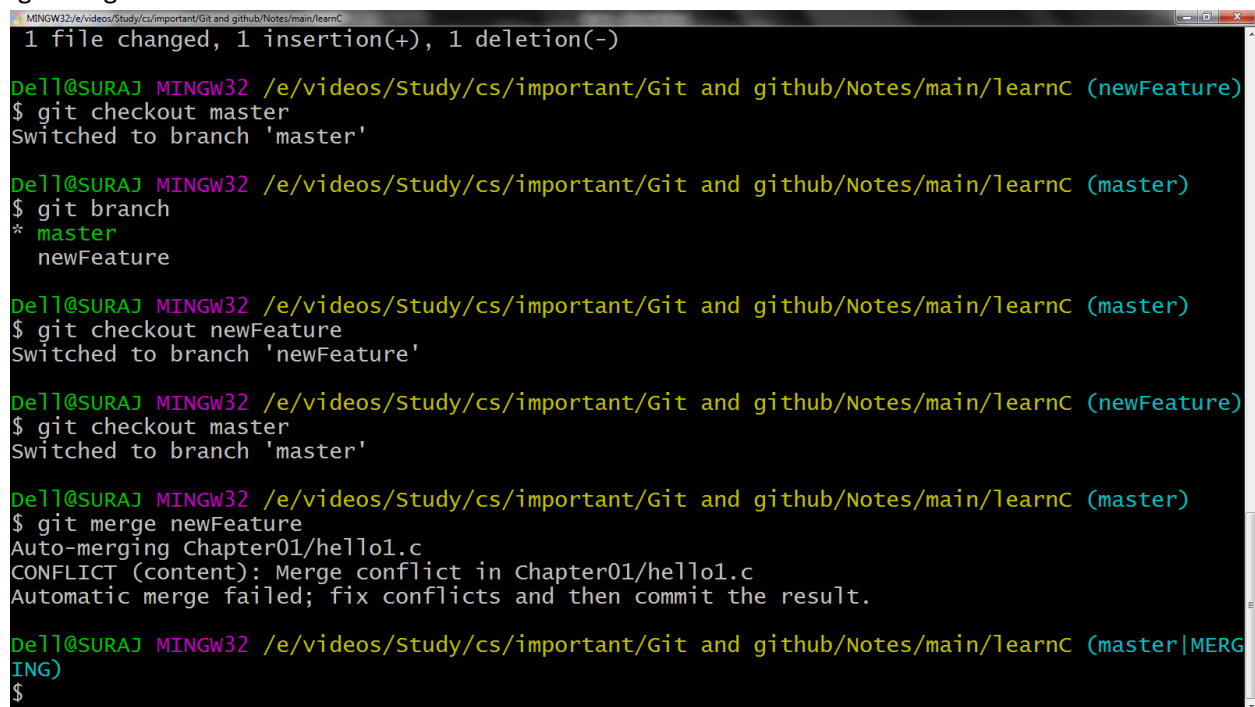
Dell@SURAJ MINGW32 /e/videos/Study/cs/important/Git and github/Notes/main/learnC (newFeature)
$
```

Stage and commit any changes if required

Note: Always switch branch after the commit

How to merge changes from newFeature branch to master branch?

“git merge newFeature” from master branch

A terminal window showing a series of git commands and their outputs. The user starts on the 'newFeature' branch, checks out 'master', lists branches, checks out 'newFeature', and then attempts to merge 'master' back into 'newFeature'. This results in a merge conflict in 'Chapter01/hello1.c'. The terminal text is as follows:

```
1 file changed, 1 insertion(+), 1 deletion(-)

Dell@SURAJ MINGW32 /e/videos/Study/cs/important/Git and github/Notes/main/learnC (newFeature)
$ git checkout master
Switched to branch 'master'

Dell@SURAJ MINGW32 /e/videos/Study/cs/important/Git and github/Notes/main/learnC (master)
$ git branch
* master
  newFeature

Dell@SURAJ MINGW32 /e/videos/Study/cs/important/Git and github/Notes/main/learnC (master)
$ git checkout newFeature
Switched to branch 'newFeature'

Dell@SURAJ MINGW32 /e/videos/Study/cs/important/Git and github/Notes/main/learnC (newFeature)
$ git checkout master
Switched to branch 'master'

Dell@SURAJ MINGW32 /e/videos/Study/cs/important/Git and github/Notes/main/learnC (master)
$ git merge newFeature
Auto-merging Chapter01/hello1.c
CONFLICT (content): Merge conflict in Chapter01/hello1.c
Automatic merge failed; fix conflicts and then commit the result.

Dell@SURAJ MINGW32 /e/videos/Study/cs/important/Git and github/Notes/main/learnC (master|MERG
ING)
$
```

But is says there is some merge conflict to overcome this accept changes from VS Code (in this case we accepted incoming changes).

And then add files to staging area and then commit the changes → write “git commit” here or also write message “git commit -m “message””

```
MINGW32/e/videos/Study/cs/important/Git and github/Notes/main/learnC
Merge branch 'newFeature'

# Conflicts:
#   Chapter01/hello1.c
#
# It looks like you may be committing a merge.
# If this is not correct, please run
#   git update-ref -d MERGE_HEAD
# and try again.
#
# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the commit.
#
# On branch master
# All conflicts fixed but you are still merging.
#
# Changes to be committed:
#   modified:   Chapter01/hello1.c
#
~
~
~
~
~
~
.git/COMMIT_EDITMSG [unix] (19:25 06/05/2022) 1,1 All
:wq
```

Press “:wq” and then press enter to exit current screen

```
MINGW32/e/videos/Study/cs/important/Git and github/Notes/main/learnC
CONFLICT (content): Merge conflict in Chapter01/hello1.c
Automatic merge failed; fix conflicts and then commit the result.

Dell@SURAJ MINGW32 /e/videos/Study/cs/important/Git and github/Notes/main/learnC (master|MERC
ING)
$ git status
On branch master
You have unmerged paths.
  (fix conflicts and run "git commit")
  (use "git merge --abort" to abort the merge)

Unmerged paths:
  (use "git add <file>..." to mark resolution)
    both modified:   Chapter01/hello1.c

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

Dell@SURAJ MINGW32 /e/videos/Study/cs/important/Git and github/Notes/main/learnC (master|MERC
ING)
$ git add .

Dell@SURAJ MINGW32 /e/videos/Study/cs/important/Git and github/Notes/main/learnC (master|MERC
ING)
$ git commit
[master 695b4cd] Merge branch 'newFeature'

Dell@SURAJ MINGW32 /e/videos/Study/cs/important/Git and github/Notes/main/learnC (master)
$ |
```