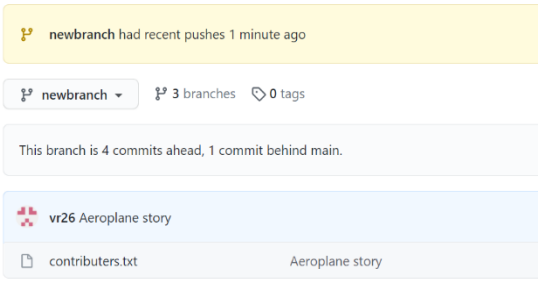
GIT ASSIGNMENTS

QUESTION 1:

While traveling, you meet an IT professional who writes plays in their spare time. They are working on a play that really resonates with you. Part of the concept of the play is that it is open source. After a spirited discussion on your flight, they tell you about the repository where the play is kept and encourage you to get involved. To contribute, you will first need to install Git and configure it with your name and email address. Once that is done, you need to pull down the repository into /home/<user-name>/play/ on the provided workstation host. Create a new branch named after your first name to begin your piece of the story. Create a new text file with your story and add it to the branch. Also, add your name to the contributers.txt file within your branch. Be sure to track your work with git and commit your changes to your local repository and use the commit message "airplane story" so your travel buddy will remember who you are!

ANSWER 1:

```
vr26@LIN82000705 MINGW64 ~/desktop/git/test (master)
$ git branch newbranch

vr26@LIN82000705 MINGW64 ~/desktop/git/test (master)
$ git checkout newbranch
Switched to branch 'newbranch'

vr26@LIN82000705 MINGW64 ~/desktop/git/test (newbranch)
$ touch contributers.txt

vr26@LIN82000705 MINGW64 ~/desktop/git/test (newbranch)
$ vi contributers.txt

vr26@LIN82000705 MINGW64 ~/desktop/git/test (newbranch)
$ git add -A
warning: LF will be replaced by CRLF in contributers.txt.
The file will have its original line endings in your working directory

vr26@LIN82000705 MINGW64 ~/desktop/git/test (newbranch)
$ git commit -m "Aeroplane story"
[newbranch e0a2084] Aeroplane story
 1 file changed, 1 insertion(+)
 create mode 100644 contributers.txt

vr26@LIN82000705 MINGW64 ~/desktop/git/test (newbranch)
$ git push origin master
Everything up-to-date

vr26@LIN82000705 MINGW64 ~/desktop/git/test (newbranch)
$ git push origin newbranch
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Writing objects: 100% (3/3), 283 bytes | 31.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'newbranch' on GitHub by visiting:
remote:      https://github.com/venkateshkrishnan7053/ASSIGNMENT/pull/new/newbranch
remote:
To https://github.com/venkateshkrishnan7053/ASSIGNMENT.git
 * [new branch]      newbranch -> newbranch
```



SUMMARY 1:

STEP 1: CREATE A NEW BRANCH NAMED NEWBRANCH
STEP 2: CREATED A FILE NAMED CONTRIBUTERS.TXT
STEP 3: CHANGED MADE IN FILE
STEP 4: PUSHED TO GITHUB

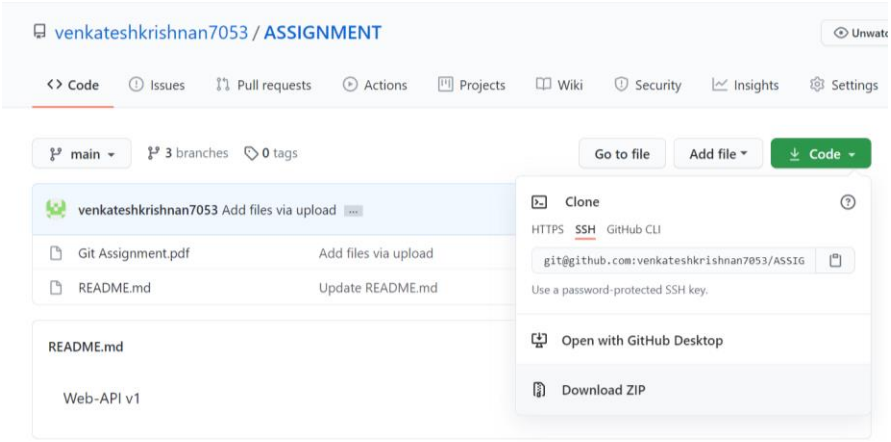
GIT ASSIGNMENTS

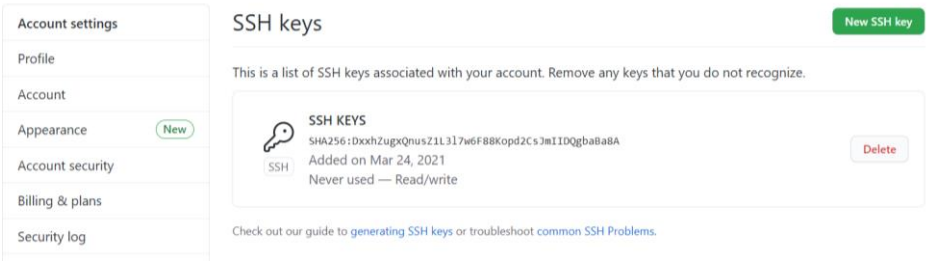
QUESTION 2:

In your efforts to better support a development organization, you have begun experimenting with Git source control and its interactions with GitHub. In order to be able to provide secure access to any repositories that are in use, you will create a special SSH key (separate from any that may have been created for other local systems to authenticate through) to be able to make an exchange with GitHub, and identify a service account called 'cloud_user' with your own account.

You will need to use the OpenSSH Client's utilities to create this key. First, create a subdirectory in your account's home directory called 'githubkeys'. Using the appropriate tool, you will create a new key (the default encryption type and size will be acceptable according to your organization's security policy) in this new directory called 'github-service-key'. Once you have this key, find the public key file that matches then copy and paste that into the appropriate account profile information in GitHub. After giving the key a reference name, you should see it appear in the list of your authorized keys, that will confirm it is available for use when trying to securely connect to your account.

ANSWER 2:





SUMMARY 2:

STEP1: CHECK GIT, IS THERE ANY CONNECTION EXIST WITH SSH
STEP 2: IF NO, GENERATE A SSH KEY USING

ssh-keygen -t ed25519 -C "your_email@example.com"

GIT ASSIGNMENTS

STEP 3: AND USE THE PATH MENTIONED IN CMD TO FIND THE KEY.
STEP 4: ONCE DONE, PASTE IT IN GITHUB>SSH KEY> ADD KEY.

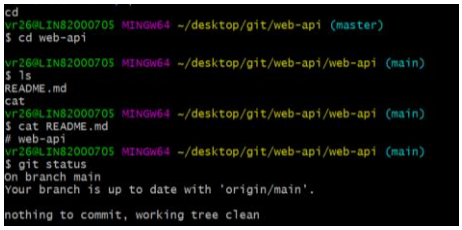
QUESTION 3:

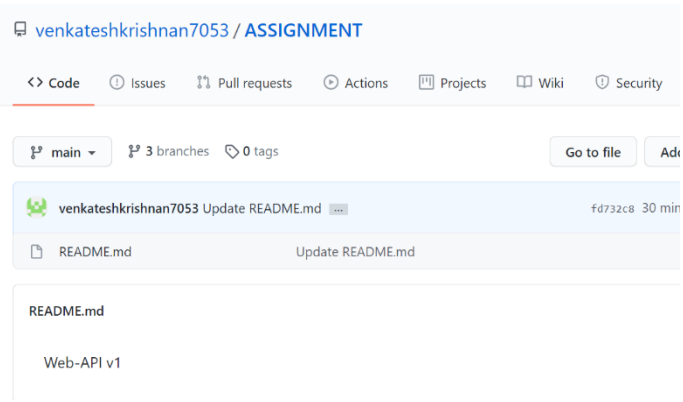
The development team in your organization is working on a new Web-based API. They have deployed a central server that they intend to use as a local Git repository for source control. They have managed to configure some of the global settings for the service account they want to use as the Git user, but are having difficulties in getting started with an empty repository.

As a result, they have provided you with the credentials for the service user and credentials to access their server. You have been asked to initialize a new Git repository in the /home/<user-name> directory called 'web-api'.

Once you have this new repository created, create a file called 'README.md' (be careful to mind the case of the file in order to follow your corporate standards for README files) that contains "Web-API v1" text. You will then need to add that file to the repository and commit the changes to the repository. Once you confirm that the default branch is up-to-date with no changes to commit, you may turn it back over to the team for use.

ANSWER 3:





SUMMARY 3:

STEP 1: CREATED ANOTHER REPO IN GITHUB NAMED AS web-api
STEP 2: CREATED THE README.md
STEP 3: MODIFIED THE README FILE IN GITHUB
STEP 4: PULLED TO LOCAL
STEP 5: COMMITED IT
STEP 6: GIT STATUS – NO COMMITS TO DO

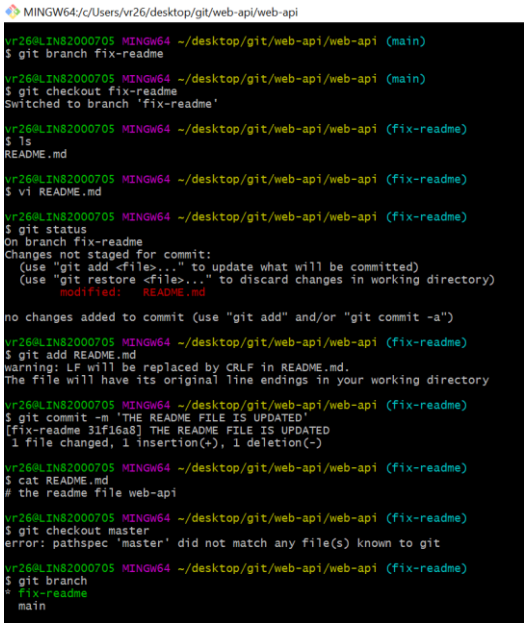
GIT ASSIGNMENTS

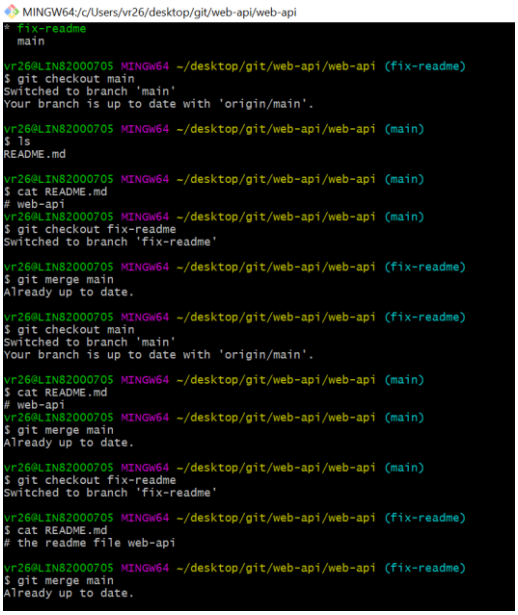
QUESTION 4:

One of your company's developers noticed that the word readme was misspelled in the web api's git repository readme file.

The company policy dictates that all modifications are to be made in branches, and that these branches are to remain intact and not deleted. until reviewed by a senior developer. You have been asked to log into the central server used for maintaining the git repository for the web api, which is located in the <user-name>'s home directory, and correct the spelling within the file. You are to use a branch named 'fix-readme', and merge the corrected file back into the master branch when finished. Again, do not remove your branch. Once you have completed this task, hand the system over to the lead developer for approval.

ANSWER 4:





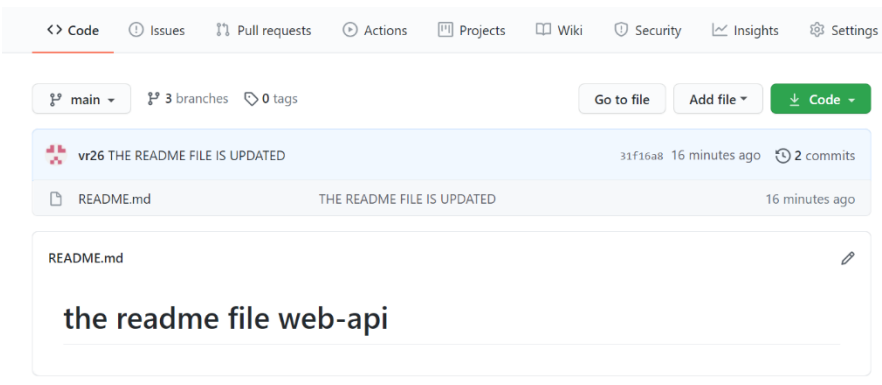
GIT ASSIGNMENTS

MINGW64:/c/Users/vr26/desktop/git/web-api/web-api

```
vr26@LIN82000705 MINGW64 ~/desktop/git/web-api/web-api (fix-readme)
$ git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.

vr26@LIN82000705 MINGW64 ~/desktop/git/web-api/web-api (main)
$ git merge fix-branch
merge: fix-branch - not something we can merge

vr26@LIN82000705 MINGW64 ~/desktop/git/web-api/web-api (main)
$ cat README.md
# web-api
vr26@LIN82000705 MINGW64 ~/desktop/git/web-api/web-api (main)
$ git branch
  fix-readme
* main

vr26@LIN82000705 MINGW64 ~/desktop/git/web-api/web-api (main)
$ git merge fix-readme
Updating b0626a0..31f16a8
Fast-forward
 README.md | 2 +-
 1 file changed, 1 insertion(+), 1 deletion(-)

vr26@LIN82000705 MINGW64 ~/desktop/git/web-api/web-api (main)
$ cat README.md
# the readme file web-api

vr26@LIN82000705 MINGW64 ~/desktop/git/web-api/web-api (main)
$ git push origin master
error: src refspec master does not match any
error: failed to push some refs to 'https://github.com/venkateshkrishnan7053/web-api.git'

vr26@LIN82000705 MINGW64 ~/desktop/git/web-api/web-api (main)
$ git status
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean

vr26@LIN82000705 MINGW64 ~/desktop/git/web-api/web-api (main)
$ git checkout main
Already on 'main'
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)

vr26@LIN82000705 MINGW64 ~/desktop/git/web-api/web-api (main)
$ git push
```

MINGW64:/c/Users/vr26/desktop/git/web-api/web-api

```
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean

vr26@LIN82000705 MINGW64 ~/desktop/git/web-api/web-api (main)
$ git checkout main
Already on 'main'
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)

vr26@LIN82000705 MINGW64 ~/desktop/git/web-api/web-api (main)
$ git push
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Writing objects: 100% (3/3), 299 bytes | 23.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/venkateshkrishnan7053/web-api.git
   b0626a0..31f16a8  main -> main

vr26@LIN82000705 MINGW64 ~/desktop/git/web-api/web-api (main)
$ git checkout fix-readme
Switched to branch 'fix-readme'

vr26@LIN82000705 MINGW64 ~/desktop/git/web-api/web-api (fix-readme)
$ git push
fatal: The current branch fix-readme has no upstream branch.
To push the current branch and set the remote as upstream, use

    git push --set-upstream origin fix-readme


vr26@LIN82000705 MINGW64 ~/desktop/git/web-api/web-api (fix-readme)
$ git push origin fix-readme
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'fix-readme' on GitHub by visiting:
remote:      https://github.com/venkateshkrishnan7053/web-api/pull/new/fix-readme
remote:
To https://github.com/venkateshkrishnan7053/web-api.git
 * [new branch]      fix-readme -> fix-readme

vr26@LIN82000705 MINGW64 ~/desktop/git/web-api/web-api (fix-readme)
$ git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.
```



GIT ASSIGNMENTS

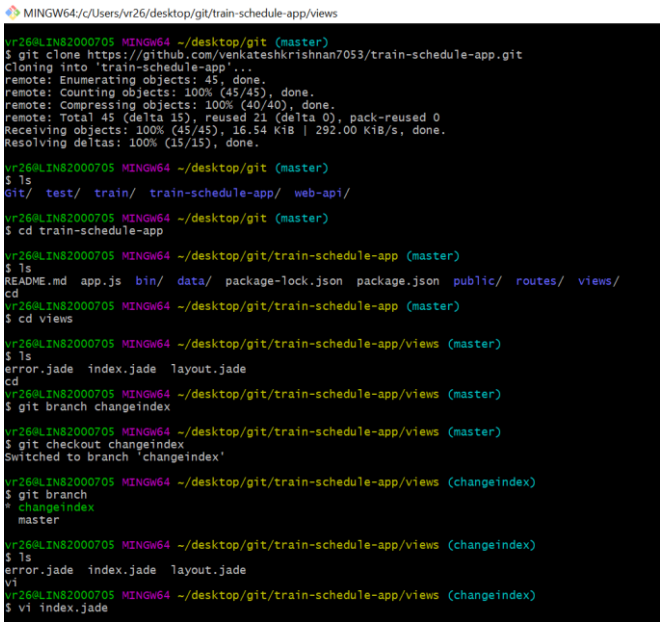
SUMMARY 4:

STEP 1: CREATED A FIX-README BRANCH
STEP 2: FIXED THE ERROR
STEP 3: COMMITED THE CHANGES
STEP 4: PUSHED THE BRANCH TO REMOTE
STEP 5: SWITCHED TO MAINA AND MERGED THE NEWLY CREAATED BRANCH
STEP 6: PUSHED TO REMOTE

QUESTION 5:

In this exercise, you are one of a team of developers. Your team uses feature branches to manage changes. You have been asked to make a change to the train-schedule app. Instead of the app's main page header reading "Train Schedule," the customer wants it to read “Find your train!” or vice-versa (You may find the text the reverse of what is said so please do the reverse). This text can be found in the project file views/index.jade, and you're going to have to change it.

ANSWER 5:



GIT ASSIGNMENTS

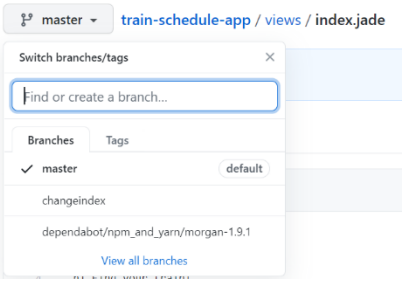
MINGW64:/c/Users/vr26/desktop/git/train-schedule-app/views

```
vr26@LIN82000705 MINGW64 ~/desktop/git/train-schedule-app/views (changeindex)
$ vi index.jade

vr26@LIN82000705 MINGW64 ~/desktop/git/train-schedule-app/views (changeindex)
$ git add index.jade

vr26@LIN82000705 MINGW64 ~/desktop/git/train-schedule-app/views (changeindex)
$ git commit -m "the changes made in index.jade"
[changeindex 9459282] the changes made in index.jade
 1 file changed, 2 insertions(+), 2 deletions(-)

vr26@LIN82000705 MINGW64 ~/desktop/git/train-schedule-app/views (changeindex)
$ cat index.jade
extends layout

block content
  h1 Find your train!
  p Select your train below to see its current schedule.
  #wrapper
    #trainList
      h2 Trains
      #trains.d-flex.flex-row
    #trainInfo
      strong  <span id='trainName'></span>
      #trainSchedule
        strong Select a train to view its current schedule.

vr26@LIN82000705 MINGW64 ~/desktop/git/train-schedule-app/views (changeindex)
$ git checkout index.jade
Updated 0 paths from the index

vr26@LIN82000705 MINGW64 ~/desktop/git/train-schedule-app/views (changeindex)
$ git checkout master
Switched to branch 'master'
Your branch is up to date with 'origin/master'.

vr26@LIN82000705 MINGW64 ~/desktop/git/train-schedule-app/views (master)
$ ls
error.jade  index.jade  layout.jade

vr26@LIN82000705 MINGW64 ~/desktop/git/train-schedule-app/views (master)
$ cat index.jade
extends layout

block content
  h1 Train Schedule
  p Select your train below to see its current schedule.
  #wrapper
    #trainList
```

MINGW64:/c/Users/vr26/desktop/git/train-schedule-app/views

```
$ cat index.jade
extends layout

block content
  h1 Train Schedule
  p Select your train below to see its current schedule.
  #wrapper
    #trainList
      h2 Trains
      #trains.d-flex.flex-row
    #trainInfo
      strong  <span id='trainName'></span>
      #trainSchedule
        strong Select a train to view its current schedule.
vr26@LIN82000705 MINGW64 ~/desktop/git/train-schedule-app/views (master)
$ git merge changeindex
Updating 89312ae..9459282
Fast-forward
 views/index.jade | 4 ++--
 1 file changed, 2 insertions(+), 2 deletions(-)

vr26@LIN82000705 MINGW64 ~/desktop/git/train-schedule-app/views (master)
$ cat index.jade
extends layout

block content
  h1 Find your train!
  p Select your train below to see its current schedule.
  #wrapper
    #trainList
      h2 Trains
      #trains.d-flex.flex-row
    #trainInfo
      strong  <span id='trainName'></span>
      #trainSchedule
        strong Select a train to view its current schedule.

vr26@LIN82000705 MINGW64 ~/desktop/git/train-schedule-app/views (master)
$ git checkout changeindex
Switched to branch 'changeindex'

vr26@LIN82000705 MINGW64 ~/desktop/git/train-schedule-app/views (changeindex)
$ git push
fatal: The current branch changeindex has no upstream branch.
To push the current branch and set the remote as upstream, use

    git push --set-upstream origin changeindex
```

MINGW64:/c/Users/vr26/desktop/git/train-schedule-app/views

```
$ git checkout changeindex
Switched to branch 'changeindex'

vr26@LIN82000705 MINGW64 ~/desktop/git/train-schedule-app/views (changeindex)
$ git push
fatal: The current branch changeindex has no upstream branch.
To push the current branch and set the remote as upstream, use

    git push --set-upstream origin changeindex


vr26@LIN82000705 MINGW64 ~/desktop/git/train-schedule-app/views (changeindex)
$ git push -u orgin changeindex
fatal: 'orgin' does not appear to be a git repository
fatal: Could not read from remote repository.

Please make sure you have the correct access rights
and the repository exists.
vr26@LIN82000705 MINGW64 ~/desktop/git/train-schedule-app/views (changeindex)
$ git push -u origin changeindex
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 8 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 434 bytes | 25.00 KiB/s, done.
Total 4 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
remote:
remote: Create a pull request for 'changeindex' on GitHub by visiting:
remote:      https://github.com/venkateshkrishnan7053/train-schedule-app/pull/new/changeindex
remote:
To https://github.com/venkateshkrishnan7053/train-schedule-app.git
 * [new branch]      changeindex -> changeindex
Branch 'changeindex' set up to track remote branch 'changeindex' from 'origin'.

vr26@LIN82000705 MINGW64 ~/desktop/git/train-schedule-app/views (changeindex)
$ git checkout master
Switched to branch 'master'
Your branch is ahead of 'origin/master' by 1 commit.
  (use "git push" to publish your local commits)

vr26@LIN82000705 MINGW64 ~/desktop/git/train-schedule-app/views (master)
$ git push origin master
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/venkateshkrishnan7053/train-schedule-app.git
   89312ae..9459282  master -> master
```

GIT ASSIGNMENTS