Faiza Gulzar Ahmed (2303.khi.deg.001)

Haseebullah Shaikh (2303.KHI.DEG.015)

Assignment no 1.3

Date 06-Apr-2023

Version Control and Git:

Version Control System:

Git is a distributed version control system which means that you can actually clone the complete protectory on the local machine and you can work there.

Version Control Systems are software tools that help in recording changes by keeping track of modifications done in the code.

Git Snapshot:

Git Snapshot store the date in a snapshot.

Git states:

There are three states in a Git.

Whenever you modify it goes to three states.

- Modified
- Staged
- Committed

Purpose of version control system:

Repository:

A repository is a data structure that stores directories and sets the files with their metadata.

Metadata is a data about data given an information describes the data repository and set the files and directories.

Centralized:

Maintain in a single server.

Distributed:

May be duplicated on every user's system (Git, Mercurial).

Share in a multiple database system.

Daily Assignment 1.3

1. Create a new git repository in a local directory.

```
(base) all@all-MS-7D35:~/Desktop/git_repo$ 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 /home/all/Desktop/git_repo/.git/
(base) all@all-MS-7D35:~/Desktop/git_repo$
```

2. Create an empty file "numbers.txt" and commit it to the current ("main" or "master") branch.

```
Initialized empty Git repository in /home/all/Desktop/git_repo/.git/
(base) all@all-MS-7D35:~/Desktop/git_repo$ touch numbers.txt
(base) all@all-MS-7D35:~/Desktop/git_repo$
```

```
/home/all/Desktop/git_repo
(base) all@all-MS-7D35:~/Desktop/git_repo$ touch numbers.txt
(base) all@all-MS-7D35:~/Desktop/git_repo$ git add numbers.txt
(base) all@all-MS-7D35:~/Desktop/git_repo$ git status
On branch master

No commits yet

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

(base) all@all-MS-7D35:~/Desktop/git_repo$ git commit
Aborting commit due to empty commit message.
(base) all@all-MS-7D35:~/Desktop/git_repo$ git commit -m "add_numbers.txt"
[master (root-commit) ad4c22d] add_numbers.txt
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 numbers.txt
(base) all@all-MS-7D35:~/Desktop/git_repo$
```

3. Create two new branches: "odd" and "even".

```
create mode 100644 numbers.txt
(base) all@all-MS-7D35:~/Desktop/git_repo$ git branch odd
(base) all@all-MS-7D35:~/Desktop/git_repo$ git branch even
(base) all@all-MS-7D35:~/Desktop/git_repo$ git show
commit ad4c22dba3c1b7e17cc7a2e89355cd3387c505c1 (HEAD -> master, odd, even)
Author: faizagulzarahmed2303-001-KHI-DEG <faiza.gulzar@xloopdigital.com>
Date:   Thu Apr 6 09:33:27 2023 +0500

    add_numbers.txt

diff --git a/numbers.txt b/numbers.txt
new file mode 100644
index 0000000..e69de29
```

4. Switch to the "odd" branch.

```
index 0000000..e69de29
(base) all@all-MS-7D35:~/Desktop/git_repo$ git checkout odd
Switched to branch 'odd'
(base) all@all-MS-7D35:~/Desktop/git_repo$
```

5. Edit the "numbers.txt" file by adding some odd numbers, then commit the change.

```
(base) all@all-MS-7D35:~/Desktop/git_repo$ gedit numbers.txt
(base) all@all-MS-7D35:~/Desktop/git_repo$ git add numbers.txt
(base) all@all-MS-7D35:~/Desktop/git_repo$ git status
On branch odd
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   numbers.txt

(base) all@all-MS-7D35:~/Desktop/git_repo$ git commit
Aborting commit due to empty commit message.
(base) all@all-MS-7D35:~/Desktop/git_repo$ git commit -m "updated numbers.txt"
[odd 06a98cd] updated numbers.txt
 1 file changed, 2 insertions(+)
(base) all@all-MS-7D35:~/Desktop/git_repo$
```

6. Switch to the "even" branch.

```
(base) all@all-MS-7D35:~/Desktop/git_repo$ git checkout even
Switched to branch 'even'
(base) all@all-MS-7D35:~/Desktop/git_repo$ 
```

7. Edit the "numbers.txt" file by adding some even numbers, then commit the change.

```
(base) all@all-MS-7D35:~/Desktop/git_repo$ git checkout even
Switched to branch 'even'
(base) all@all-MS-7D35:~/Desktop/git_repo$ gedit numbers.txt
(base) all@all-MS-7D35:~/Desktop/git_repo$ git add numbers.txt
(base) all@all-MS-7D35:~/Desktop/git_repo$ git status
On branch even
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   numbers.txt

(base) all@all-MS-7D35:~/Desktop/git_repo$ git commit -m "updated numbers.txt"
[even 76f0517] updated numbers.txt
 1 file changed, 1 insertion(+)
(base) all@all-MS-7D35:~/Desktop/git_repo$
```

8. Switch to the main branch.

```
(base) all@all-MS-7D35:~/Desktop/git_repo$ git checkout master
Switched to branch 'master'
(base) all@all-MS-7D35:~/Desktop/git_repo$ 
```

9. Merge the "odd" branch to the current one.

```
Switched to branch 'master'
(base) all@all-MS-7D35:~/Desktop/git_repo$ git merge odd
Updating ad4c22d..06a98cd
Fast-forward
 numbers.txt | 2 ++
 1 file changed, 2 insertions(+)
(base) all@all-MS-7D35:~/Desktop/git_repo$
```

10. Try to merge the "even" branch too – it should cause a conflict.

```
 1 file changed, 2 insertions(+)
(base) all@all-MS-7D35:~/Desktop/git_repo$ git merge even
Auto-merging numbers.txt
CONFLICT (content): Merge conflict in numbers.txt
Automatic merge failed; fix conflicts and then commit the result.
```

11. Resolve the conflict by combining all the numbers and commit the merge.

```
(base) all@all-MS-7D35:~/Desktop/git_repo$ gedit numbers.txt
(base) all@all-MS-7D35:~/Desktop/git_repo$ git add numbers.txt
(base) all@all-MS-7D35:~/Desktop/git_repo$ git commit -m "solved_conflict"
[master ab4d283] solved_conflict
(base) all@all-MS-7D35:~/Desktop/git_repo$ gedit numbers.txt
(base) all@all-MS-7D35:~/Desktop/git_repo$ git merge even
Already up to date.
```

List the commits using git log -p command and save the output in a file.

```
(base) all@all-MS-7D35:~/Desktop/git_repo$ git log -p
commit ab4d283a6f5521e7449dea2269f4c5b08d379957 (HEAD -> master)
Merge: 06a98cd 76f0517
Author: faizagulzarahmed2303-001-KHI-DEG <faiza.gulzar@xloopdigital.com>
Date:   Thu Apr 6 09:42:24 2023 +0500

    solved_conflict

commit 76f05179ded3d26ba24307716023e9abd8089ed0 (even)
Author: faizagulzarahmed2303-001-KHI-DEG <faiza.gulzar@xloopdigital.com>
Date:   Thu Apr 6 09:39:26 2023 +0500

    updated numbers.txt

diff --git a/numbers.txt b/numbers.txt
index e69de29..3d8be79 100644
--- a/numbers.txt
+++ b/numbers.txt
@@ -0,0 +1 @@
+2,4,6,8,10,12

commit 06a98cdb5211f4152a515b02006645322bd92da3 (odd)
Author: faizagulzarahmed2303-001-KHI-DEG <faiza.gulzar@xloopdigital.com>
Date:   Thu Apr 6 09:37:34 2023 +0500

    updated numbers.txt

diff --git a/numbers.txt b/numbers.txt
index e69de29..2b91a21 100644
--- a/numbers.txt
+++ b/numbers.txt
@@ -0,0 +1,2 @@
+1,3,5,7,9,11
+

commit ad4c22dba3c1b7e17cc7a2e89355cd3387c505c1
Author: faizagulzarahmed2303-001-KHI-DEG <faiza.gulzar@xloopdigital.com>
Date:   Thu Apr 6 09:33:27 2023 +0500

    add_numbers.txt

diff --git a/numbers.txt b/numbers.txt
new file mode 100644
index 0000000..e69de29
(base) all@all-MS-7D35:~/Desktop/git_repo$
```