

UNIT 1.3 GRADED ASSIGNMENT

Group members

Ifra Saleem (2303.khi.deg.003)

Muhammad Khan (2303.khi.deg.027)

UNIT 1.3 GRADED ASSIGNMENT

Task:

1. Create a new git repository in a local directory.
2. Create an empty file “numbers.txt” and commit it to the current (“main” or “master”) branch.
3. Create two new branches: “odd” and “even”.
4. Switch to the “odd” branch.
5. Edit the “numbers.txt” file by adding some odd numbers, then commit the change.
6. Switch to the “even” branch.
7. Edit the “numbers.txt” file by adding some even numbers, then commit the change.
8. Switch to the main branch.
9. Merge the “odd” branch to the current one.
10. Try to merge the “even” branch too – it should cause a conflict.
11. Resolve the conflict by combining all the numbers and commit the merge.
12. List the commits using git log -p command and save the output in a file.

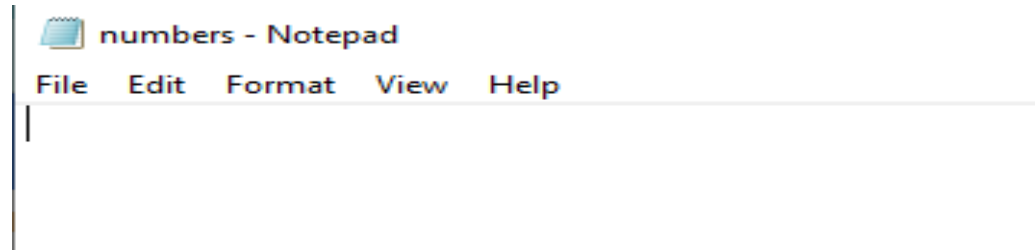
Solution:

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

```
C:\Users\asdd\Desktop\Git_Assignment>git init
Initialized empty Git repository in C:/Users/asdd/Desktop/Git_Assignment/.git/
```

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

```
C:\Users\asdd\Desktop\Git>touch numbers.txt
```



```
C:\Users\asdd\Desktop\Git_Assignment>git add numbers.txt
```

```
C:\Users\asdd\Desktop\Git>git commit -m "commit numbers.txt to main branch"
[master (root-commit) b15d242] commit numbers.txt to main branch
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 numbers.txt
```

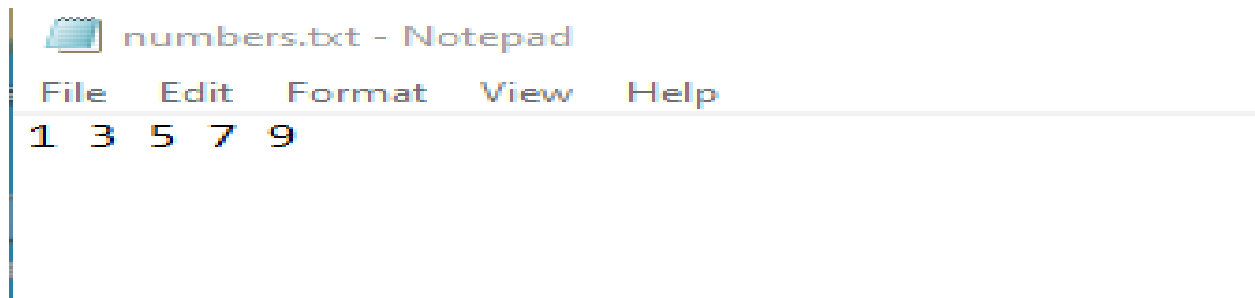
3. Create two new branches: "odd" and "even".

```
C:\Users\asdd\Desktop\Git>git branch odd
C:\Users\asdd\Desktop\Git>git branch even
```

4. Switch to the "odd" branch.

```
C:\Users\asdd\Desktop\Git>git checkout odd
Switched to branch 'odd'
```

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

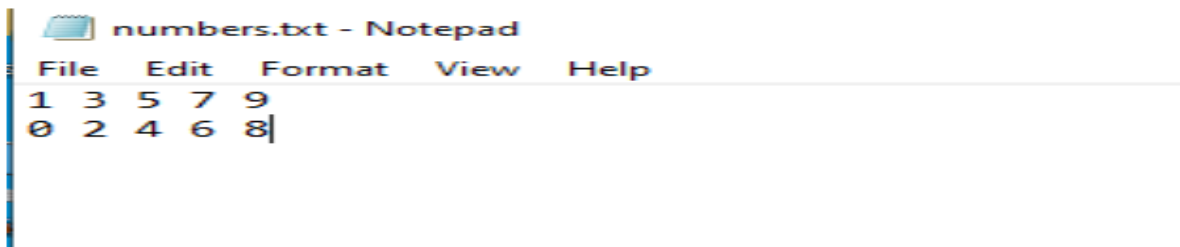


```
C:\Users\asdd\Desktop\Git>git add numbers.txt
C:\Users\asdd\Desktop\Git>git commit -m "commit numbers.txt to odd branch"
[odd d70e4f8] commit numbers.txt to odd branch
1 file changed, 1 insertion(+)
```

6. Switch to the "even" branch.

```
C:\Users\asdd\Desktop\Git>git checkout even
Switched to branch 'even'
```

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



```
C:\Users\asdd\Desktop\Git>git add numbers.txt
C:\Users\asdd\Desktop\Git>git commit -m "commit numbers.txt to even branch"
[even 2911bed] commit numbers.txt to even branch
1 file changed, 2 insertions(+)
```

8. Switch to the main branch.

```
C:\Users\asdd\Desktop\Git>git checkout master
Switched to branch 'master'
```

9. Merge the “odd” branch to the current one.

```
C:\Users\asdd\Desktop\Git>git merge odd
Updating b15d242..d70e4f8
Fast-forward
 numbers.txt | 1 +
1 file changed, 1 insertion(+)
```

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

```
C:\Users\asdd\Desktop\Git>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.



numbers.txt - Notepad

File	Edit	Format	View	Help					
1	3	5	7	9	0	2	4	6	8

```
C:\Users\asdd\Desktop\Git>git add numbers.txt
```

```
C:\Users\asdd\Desktop\Git>git commit -m "resolve the conflict and merge the even branch to main branch"  
[master 35b0892] resolve the conflict and merge the even branch to main branch
```

```
C:\Users\asdd\Desktop\Git>git merge even  
Already up to date.
```

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

```
C:\Users\asdd\Desktop\Git>git log -p > results.txt
```