

Simulating and Resolving Merge Conflicts - Task 4

➤ Creating a sample file

```
C:\k\Presidio\Git\Task4>echo "hello world">conflict.txt
```

```
C:\k\Presidio\Git\Task4>git add conflict.txt
```

```
C:\k\Presidio\Git\Task4>git commit -m "Initial commit with conflict.txt"
[master (root-commit) 742945c] Initial commit with conflict.txt
1 file changed, 1 insertion(+)
create mode 100644 conflict.txt
```

➤ Creating two branches

```
C:\k\Presidio\Git\Task4>git checkout -b branch1
Switched to a new branch 'branch1'
```

```
C:\k\Presidio\Git\Task4>echo "Change from branch1">conflict.txt
```

```
C:\k\Presidio\Git\Task4>git commit -am "Updated conflict.txt in branch1"
[branch1 524365d] Updated conflict.txt in branch1
1 file changed, 1 insertion(+), 1 deletion(-)
```

- Switching to master and creating another branch:

```
C:\k\Presidio\Git\Task4>git checkout master
Switched to branch 'master'
```

```
C:\k\Presidio\Git\Task4>git checkout -b branch2
Switched to a new branch 'branch2'
```

- Modifying the same line in “conflict.txt”:

```
C:\k\Presidio\Git\Task4>echo "Change from branch2">conflict.txt
```

```
C:\k\Presidio\Git\Task4>git commit -am "Updated conflict.txt in branch2"
[branch2 fe5f155] Updated conflict.txt in branch2
1 file changed, 1 insertion(+), 1 deletion(-)
```

➤ **Merging the branches**

- Merging branch1:

```
C:\k\Presidio\Git\Task4>git checkout master
Switched to branch 'master'

C:\k\Presidio\Git\Task4>git merge branch1
Updating 742945c..524365d
Fast-forward
 conflict.txt | 2 + -
 1 file changed, 1 insertion(+), 1 deletion(-)
```

- Merging branch2 into main gives a **merge conflict** message:

```
C:\k\Presidio\Git\Task4>git merge branch2
Auto-merging conflict.txt
CONFLICT (content): Merge conflict in conflict.txt
Automatic merge failed; fix conflicts and then commit the result.
```

➤ **Checking the conflict**

```
C:\k\Presidio\Git\Task4>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:   conflict.txt

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    Task4.docx
    ~$Task4.docx
```

➤ **Content in conflict.txt:**

```
|<<<<<< HEAD
"Change from branch1"
=====
"Change from branch2"
>>>>>> branch2
```

- Manually editing the file:

"Change from both branches"

➤ **Mark the conflict as resolved and complete the merge**

```
C:\k\Presidio\Git\Task4>git commit -m "Resolved merge conflict in conflict.txt"
[master af1e5b9] Resolved merge conflict in conflict.txt
```

➤ **Checking the merge success**

```
C:\k\Presidio\Git\Task4>git log --oneline --graph
*   af1e5b9 (HEAD -> master) Resolved merge conflict in conflict.txt
| \
| * fe5f155 (branch2) Updated conflict.txt in branch2
* | 524365d (branch1) Updated conflict.txt in branch1
|/
* 742945c Initial commit with conflict.txt
```

➤ **Clean Up** (deleting the branches)

```
C:\k\Presidio\Git\Task4>git branch -d branch1
Deleted branch branch1 (was 524365d).
```

```
C:\k\Presidio\Git\Task4>git branch -d branch2
Deleted branch branch2 (was fe5f155).
```