

Esercizio di risoluzione di un merge conflict

Il tempo massimo in laboratorio per questo esercizio è di 20 minuti.

Se superato, sospendere l'esercizio e riprenderlo per ultimo!

Si visiti <https://github.com/APICe-at-DISI/OOP-git-merge-conflict-test>.

Questo repository contiene due branch: **master** e **feature**

Per ognuna delle seguenti istruzioni, si annoti l'output ottenuto.

Prima di eseguire ogni operazione sul worktree o sul repository,

si verifichi lo stato del repository con **git status**.

1. Si cloni localmente il repository

```
PS C:\Users\psalv\Documenti\java\esercizi_lab\pss-lab-git-exercises\pss-lab-  
git-exercises\esercizio2test> git clone https://github.com/APICe-at-DISI/OOP-  
git-merge-conflict-test.git  
Cloning into 'OOP-git-merge-conflict-test'...  
remote: Enumerating objects: 12, done.  
remote: Counting objects: 100% (4/4), done.  
remote: Compressing objects: 100% (3/3), done.  
remote: Total 12 (delta 1), reused 1 (delta 1), pack-reused 8 (from 1)  
Receiving objects: 100% (12/12), done.  
Resolving deltas: 100% (2/2), done.
```

2. Ci si assicuri di avere localmente entrambi i branch remoti

```
PS C:\Users\psalv\Documenti\java\esercizi_lab\pss-lab-git-exercises\pss-lab-  
git-exercises\esercizio2test\OOP-git-merge-conflict-test> git branch  
* master  
  
* bed943f (origin/feature) Print author information  
| * 8e0f29c (HEAD -> master, origin/master, origin/HEAD) Change HelloWorld to  
| print the number of available processors  
|/  
* d956df6 Create .gitignore  
* 700ee0b Create HelloWorld
```

3. Si faccia il merge di **feature** dentro **master**, ossia: si posizioni la **HEAD** su **master**
e da qui si esegua il merge di **feature**

```
PS C:\Users\psalv\Documenti\java\esercizi_lab\pss-lab-git-exercises\pss-lab-  
git-exercises\esercizio2test\OOP-git-merge-conflict-test> git merge feature --  
no-edit  
Auto-merging HelloWorld.java
```

```
CONFLICT (content): Merge conflict in HelloWorld.java
Automatic merge failed; fix conflicts and then commit the result.
```

4. Si noti che viene generato un **merge conflict**!

5. Si risolva il merge conflict come segue:

- Il programma Java risultante deve stampare sia il numero di processori disponibili (funzionalità presente su **master**) che il nome dell'autore del file (funzionalità presente su **feature**)

```
PS C:\Users\psalv\Documenti\java\esercizi_lab\pss-lab-git-exercises\pss-lab-
git-exercises\esercizio2test\OOP-git-merge-conflict-test> git merge feature --
no-edit
Auto-merging HelloWorld.java
CONFLICT (content): Merge conflict in HelloWorld.java
Automatic merge failed; fix conflicts and then commit the result.
```

```
PS C:\Users\psalv\Documenti\java\esercizi_lab\pss-lab-git-exercises\pss-lab-
git-exercises\esercizio2test\OOP-git-merge-conflict-test> git add
.\HelloWorld.java
```

```
PS C:\Users\psalv\Documenti\java\esercizi_lab\pss-lab-git-exercises\pss-lab-
git-exercises\esercizio2test\OOP-git-merge-conflict-test> git commit
[master 841f2e5] Merge branch 'feature'
PS C:\Users\psalv\Documenti\java\esercizi_lab\pss-lab-git-exercises\pss-lab-
git-exercises\esercizio2test\OOP-git-merge-conflict-test> git log --graph
*   commit 841f2e5799fa003b32868c8f013efe4347ca9467 (HEAD -> master)
| \ Merge: 8e0f29c bed943f
|  | Author: RED-ITA <enricosalvagiani@gmail.com>
|  | Date:   Tue May 27 20:13:33 2025 +0200
|  |
|  | Merge branch 'feature'
|  |
| * commit bed943fbdd6ba94e64197448e4754a529d984e88 (origin/feature, feature)
|  | Author: Danilo Pianini <daniilo.pianini@gmail.com>
|  | Date:   Thu Oct 27 17:21:22 2016 +0200
|  |
|  | Print author information
|  |
| * commit 8e0f29c12e060f3bdc62540343eff3e473616d61 (origin/master,
origin/HEAD)
| / Author: Danilo Pianini <daniilo.pianini@gmail.com>
|   Date:   Thu Oct 27 17:19:05 2016 +0200
|
```

```
|      Change HelloWorld to print the number of available processors
|
* commit d956df66aeb0829f23b7b3d0d9a1c002c390f87f
| Author: Danilo Pianini <danilo.pianini@gmail.com>
| Date:   Thu Oct 27 17:17:43 2016 +0200
|
|      Create .gitignore
|
:
```

6. Si crei un nuovo repository nel proprio github personale

```
https://github.com/RED-ITA/test_git.git
```

7. Si aggiunga il nuovo repository creato come **remote** e si elenchino i remote

```
PS C:\Users\psalv\Documenti\java\esercizi_lab\pss-lab-git-exercises\pss-lab-
git-exercises\esercizio2test\OOP-git-merge-conflict-test> git remote -v
origin https://github.com/APICe-at-DISI/OOP-git-merge-conflict-test.git
(fetch)
origin https://github.com/APICe-at-DISI/OOP-git-merge-conflict-test.git (push)
```

```
PS C:\Users\psalv\Documenti\java\esercizi_lab\pss-lab-git-exercises\pss-lab-
git-exercises\esercizio2test\OOP-git-merge-conflict-test> git remote add
personale https://github.com/RED-ITA/test_git.git
```

```
PS C:\Users\psalv\Documenti\java\esercizi_lab\pss-lab-git-exercises\pss-lab-
git-exercises\esercizio2test\OOP-git-merge-conflict-test> git remote -v
origin https://github.com/APICe-at-DISI/OOP-git-merge-conflict-test.git
(fetch)
origin https://github.com/APICe-at-DISI/OOP-git-merge-conflict-test.git (push)
personale https://github.com/RED-ITA/test_git.git (fetch)
personale https://github.com/RED-ITA/test_git.git (push)
```

8. Si faccia push del branch **master** sul proprio repository

```
PS C:\Users\psalv\Documenti\java\esercizi_lab\pss-lab-git-exercises\pss-lab-
git-exercises\esercizio2test\OOP-git-merge-conflict-test> git push personale
master
Enumerating objects: 15, done.
Counting objects: 100% (15/15), done.
Delta compression using up to 16 threads
```

```
Compressing objects: 100% (11/11), done.  
Writing objects: 100% (15/15), 1.56 KiB | 1.56 MiB/s, done.  
Total 15 (delta 4), reused 10 (delta 2), pack-reused 0 (from 0)  
remote: Resolving deltas: 100% (4/4), done.  
To https://github.com/RED-ITA/test_git.git  
* [new branch]      master -> master
```

9. Si setti il branch remoto `master` del nuovo repository come *upstream* per il proprio branch `master` locale

```
PS C:\Users\psalv\Documenti\java\esercizi_lab\pss-lab-git-exercises\pss-lab-  
git-exercises\esercizio2test\OOP-git-merge-conflict-test> git branch --set-  
upstream-to=personale/master  
branch 'master' set up to track 'personale/master'.
```