Explain how to resolve the conflict during merge.

A **merge conflict** happens when:

* Two branches change the **same part of the same file**,
* and Git doesn’t know which change to keep.

Example:

* In master, line 10 of hello.xml says:

php-template

CopyEdit

<greeting>Hello from master</greeting>

* In GitWork branch, line 10 of hello.xml says:

php-template

CopyEdit

<greeting>Hello from branch</greeting>

When you try to merge, Git detects both changed that line → **conflict**.

**How Git Shows a Conflict**

In the file, Git inserts conflict markers:

php-template

CopyEdit

<<<<<<< HEAD

<greeting>Hello from master</greeting>

=======

<greeting>Hello from branch</greeting>

>>>>>>> GitWork

* <<<<<<< HEAD → your current branch (the one you’re merging into)
* ======= → separator between the two versions
* >>>>>>> branch-name → the branch being merged

**How to Resolve a Conflict**

1. **Open the file** in your text editor (Notepad++, VS Code, etc.).
2. **Decide what to keep**:
   * Keep one side entirely, or
   * Combine changes from both.
3. **Remove the Git markers** (<<<<<<<, =======, >>>>>>>).
4. **Save the file**.

**Mark Conflict as Resolved**

Once fixed:

bash

CopyEdit

git add <file>

git commit -m "Resolve merge conflict in <file>"

**Tools for Easier Conflict Resolution**

* **Manual Editing** (Notepad++, Vim, etc.)
* **Visual Merge Tools**:
  + P4Merge
  + VS Code Merge Editor
  + Meld

If using P4Merge:

bash

CopyEdit

git mergetool --tool=p4merge





















