

竞争行更改合并冲突

要解决由竞争行更改导致的合并冲突，您必须从新提交的不同分支中选择要合并的更改。

例如，如果你和另一个人都在同一 Git 存储库不同分支的同一行中编辑了 `styleguide.md` 文件，则在尝试合并这些分支时会发生合并冲突错误。必须使用新提交解决这一合并冲突，然后才能合并这些分支。

打开 Git Bash。

导航到有合并冲突的本地 Git 仓库中。

cd REPOSITORY-NAME

生成受合并冲突影响的文件列表。在此示例中，`styleguide.md` 文件存在合并冲突。

```
$ git status
```

```
> # On branch branch-b
```

```
> # You have unmerged paths.
```

```
> #   (fix conflicts and run "git commit")
```

```
> #
```

```
> # Unmerged paths:
```

```
> #   (use "git add ..." to mark resolution)
```

```
> #
```

```
> # both modified:      styleguide.md
```

```
> #
```

```
> no changes added to commit (use "git add" and/or "git commit -a")
```

打开你首选的文本编辑器，例如 Visual Studio Code，然后导航到有合并冲突的文件。

要在文件中查看合并冲突的开头，请在文件中搜索冲突标记 <<<<<<。当你在文本编辑器中打开文件时，你将在行 <<<<<< HEAD 后看到头部分支或基本分支。接下来，你将看到 =====，它将你的更改与其他分支中的更改分开，后跟 >>>>>> BRANCH-NAME。在本例中，一个人在基本或头部分支中编写了“open an issue”，而另一个人在比较分支或 branch-a 中编写了“ask your question in IRC”。

- head 到 =====里面的内容是自己的 commit 的内容

- =====到 >>>>>>里面的内容是下拉的内容

If you have questions, please

<<<<<< HEAD

open an issue

=====

ask your question in IRC.

>>>>>> branch-a

决定您是否想只保持分支的更改、只保持其他分支的更改, 还是进行全新的更改(可能包含两个分支的更改)。删除冲突标记 <<<<<<、=====、>>>>>>, 并在最终合并中进行所需的更改。在本例中，两个更改均整合到最终合并：

If you have questions, please open an issue or ask in our IRC channel if it's more urgent.

添加或暂存您的更改。

\$ git add .

提交您的更改及注释。

\$ git commit -m "Resolved merge conflict by incorporating both suggestions."

现在，可以在命令行上合并分支，或者在 GitHub 上将更改推送到远程存储库，然后在拉取请求中合并更改。

删除的文件合并冲突

要解决由对文件进行竞争更改而导致的合并冲突，对于一个人删除分支中的文件而另一个人编辑同一文件的情况，您必须选择是删除还是将删除的文件保留在新提交中。

例如，如果你编辑了一个文件（如 README.md），而另一个人在同一 Git 存储库的另一个分支中删除了同一文件，那么当你尝试合并这些分支时将发生合并冲突错误。 必须使用新提交解决这一合并冲突，然后才能合并这些分支。

打开 Git Bash。

导航到有合并冲突的本地 Git 仓库中。

cd REPOSITORY-NAME

生成受合并冲突影响的文件列表。 在此示例中，文件 README.md 存在合并冲突。

```
$ git status
> # On branch main
> # Your branch and 'origin/main' have diverged,
> # and have 1 and 2 different commits each, respectively.
> # (use "git pull" to merge the remote branch into yours)
> # You have unmerged paths.
> # (fix conflicts and run "git commit")
> #
> # Unmerged paths:
> # (use "git add/rm ..." as appropriate to mark resolution)
> #
> #   deleted by us:   README.md
```

```
> #
```

```
> # no changes added to commit (use "git add" and/or "git commit -a")
```

打开你首选的文本编辑器，例如 Visual Studio Code，然后导航到有合并冲突的文件。

决定是否要保留删除的文件。您可能想要在文本编辑器中查看对删除的文件所做的最新更改。

要将已删除的文件重新添加到仓库：

```
$ git add README.md
```

要从仓库中删除此文件：

```
$ git rm README.md
```

```
> README.md: needs merge
```

```
> rm 'README.md'
```

提交您的更改及注释。

```
$ git commit -m "Resolved merge conflict by keeping README.md file."
```

```
> [branch-d 6f89e49] Merge branch 'branch-c' into branch-d
```

现在，可以在命令行上合并分支，或者在 GitHub 上将更改推送到远程存储库，然后在拉取请求中合并更改。

延伸阅读