

Lab: (Sourcetree) Rewriting History

Estimated time: 10 minutes

Note: This lab assumes that you are using Sourcetree. If you would prefer to use a command line interface, there are separate instructions.

In this lab, you will:

1. Amend a commit.
2. Use interactive rebase to remove a commit.

1: Amend a commit.

1. Use Sourcetree to create a local repository named `projecth`.
2. Create an initial commit that adds a `fileA.txt` file containing the text "mistake". The commit message should be "add fileA.txt with mistake". Note the first 10 characters of the commit's SHA-1.
3. Amend the commit so that `fileA.txt` is empty and the commit message simply reads "add fileA.txt".
 - Modify `fileA.txt` so that it is empty.
 - Add `fileA.txt` to the staging area.
 - **(Mac)** Select `Commit`. Under `Commit Options...`, select `Amend last commit`. Specify a commit message of "add fileA.txt". Click `Commit`.
 - **(Windows)** Under `Commit Options...` (lower right corner), select `Amend latest commit`. Specify a commit message of "add fileA.txt". Click `Commit`.
4. Notice the SHA-1 of your amended commit. It should be different than the one you noted prior to amending the commit.

Congratulations, you have amended a commit.

2: Use interactive rebase to remove a commit.

1. Create a `feature1` branch.
2. Create a commit on the `feature1` branch. Add the string "feature 1 wip" to `fileA.txt`. The commit message should be "add feature 1 wip".

3. Create another commit on the `feature1` branch. The contents of `fileA.txt` should be "feature 1". The commit message should be "add feature 1".
4. Let's say that you think that the "feature 1 wip" commit adds little value to the commit history. Use interactive rebase to remove that commit by squashing it.
 - Checkout the `feature1` branch.
 - Right-click on the initial commit in the repository (which is also the first commit of the `feature1` branch) and select `Rebase children of (SHA-1) interactively...`.
 - Explore the interactive rebase form.
 - Remove the commit with the commit message of "feature 1 wip". Do this by selecting somewhere on the **most recent** commit and clicking `Squash with previous`. You should see that two commits will be squashed into one commit.
 - Click `Edit message` and change the commit message to simply read `add feature 1`.
 - Click OK to perform the interactive rebase.
5. View your commit graph. Verify that the commit with a message of "feature 1 wip" has been removed.
6. You will not use the `projecth` repository in future labs. You can delete it.

Congratulations, you have performed an interactive rebase and have completed this lab.