


Git

Clase 5: Tagging, Branching, and Merging

Intro

SEND FEEDBACK

INGLÉS - English
Haz clic en  para acceder a la configuración.

Now that you're a pro at making commits,

0:00 / 1:05

YouTube

Windows taskbar: 11:35 p. m. 28/02/2022

Clase 5: Tagging, Branching, and Merging

Tagging

SEND FEEDBACK

The Terminal application showing the removal of a tag by using the `-d` flag. The command that is run is `git tag -d v1.0`.

QUESTION 1 OF 3

By default, a Git tag will not appear in a log. What flag must be used to display the tag information in the output of `git log`?

- ☐ --show-tags
- ☐ --tags
- ☐ --display-all
- ☒ --decorate

ENVIAR

Windows taskbar: 11:45 p. m. 28/02/2022

Introducción x Git Home x TareaGit x TareaGit x POO/Ejerc x TareaGit x POO/Tarea x Version Control x Using the x

https://classroom.udacity.com/courses/ud123/lessons/61776182-d3f-4767-9558-5d1591c1a709/co...

Clase 5: Tagging, Branching, and Merging

BUSCAR

RECURSOS

CONTENIDO

- 1. Intro
- 2. Tagging
- 3. Branching
- 4. Branching Effectively
- 5. Merging
- 6. Merge Conflicts
- 7. Outro

Tagging

SEND FEEDBACK

ENVIAR

QUESTION 2 OF 3

Which of the following will delete the tag `v-1`?

- ☒ `git tag --delete v-1`
- ☐ `git remove v-1`
- ☒ `git tag -d v-1`
- ☐ `git delete v-1`

ENVIAR

Windows taskbar: 11:45 p. m. 28/02/2022

Introducción x Git Home x TareaGit x TareaGit x POO/Ejerc x TareaGit x POO/Tarea x Version Control x Using the x

https://classroom.udacity.com/courses/ud123/lessons/61776182-d3f-4767-9558-5d1591c1a709/co...

Clase 5: Tagging, Branching, and Merging

BUSCAR

RECURSOS

CONTENIDO

- 1. Intro
- 2. Tagging
- 3. Branching
- 4. Branching Effectively
- 5. Merging
- 6. Merge Conflicts
- 7. Outro

Tagging

SEND FEEDBACK

commit with the SHA `a87084` with the tag `v1.0`. Using this technique, you can tag any commit in the entire git repository! Pretty neat, right?...and it's just a simple addition to add the SHA of a commit to the Git tagging command you already know.

Tag Older Commit?

Using the following `git log --oneline` information, what command would you run to give the commit with the message "style page header" a tag of `beta`?

```
2a9e9f3 add breakpoint for large-sized screens
137a0bd add breakpoint for medium-sized screens
c5ee895 add space around page edge
b552fa5 style page header
f8c87c7 convert social links from text to images
```

`$ git tag -a beta b552fa5`

Git Tag Recap

Windows taskbar: 11:45 p. m. 28/02/2022

Introducción x Git Home x TareaGit x TareaGit x POO/Ejerc x TareaGit x POO/Tarea x Version Control x Using the x

https://classroom.udacity.com/courses/ud123/lessons/61776182-d3f-4767-9558-5d1591c1a709/co...

Clase 5: Tagging, Branching, and Merging

BUSCAR

RECURSOS

CONTENIDO

- 1. Intro
- 2. Tagging
- 3. Branching
- 4. Branching Effectively
- 5. Merging
- 6. Merge Conflicts
- 7. Outro

Create A Branch

To create a branch, all you have to do is use `git branch` and provide it the name of the branch you want it to create. So if you want a branch called "sidebar", you'd run this command:

```
$ git branch sidebar
```

QUESTION 1 OF 3

Remember that there are a number of branches in the repository, but that the command prompt displays the *current branch*.

Now that we just created a new "sidebar" branch, does the command prompt display `sidebar` or `master`?

☐ sidebar

☒ master

ENVIAR

ESP LAA 11:46 p. m. 28/02/2022

Introducción x Git Home x TareaGit x TareaGit x POO/Ejerc x TareaGit x POO/Tarea x Version Control x Using the x

https://classroom.udacity.com/courses/ud123/lessons/61776182-d3f-4767-9558-5d1591c1a709/co...

Clase 5: Tagging, Branching, and Merging

BUSCAR

RECURSOS

CONTENIDO

- 1. Intro
- 2. Tagging
- 3. Branching
- 4. Branching Effectively
- 5. Merging
- 6. Merge Conflicts
- 7. Outro

Branching

SEND FEEDBACK

QUESTION 2 OF 3

From what you know about both the `git branch` and `git tag` commands, what do you think the following command will do?

```
$ git branch alt-sidebar-loc 42a69f
```

☐ will create a branch `alt` at the same commit as the `master` branch

☐ will create the 3 branches `alt`, `sidebar`, `loc`

☐ will move the master branch to the commit with SHA `42a69f`

☒ will create the `alt-sidebar-loc` branch and have it point to the commit with SHA `42a69f`

ENVIAR

ESP LAA 11:46 p. m. 28/02/2022

Introducción x Git Home x TareaGit x TareaGit x POO/Ejerc x TareaGit x POO/Tarea x Version C x Using the x + -

https://classroom.udacity.com/courses/ud123/lessons/61776182-d3f-4767-9558-5d1591c1a709/co...

Clase 5: Tagging, Branching, and Merging

BUSCAR

RECURSOS

CONTENIDO

- 1. Intro
- 2. Tagging
- 3. Branching
- 4. Branching Effectively
- 5. Merging
- 6. Merge Conflicts
- 7. Outro

Branching

SEND FEEDBACK

ENVIAR

Which Branch Is Active?
Given the following output from `git branch`:

```
$ git branch
barbara
* footer-fix
master
richard
sidebar
social-icons
```

Which branch is the active branch?

footer-fix

11:47 p. m. 28/02/2022

Introducción x Git Home x TareaGit x TareaGit x POO/Ejerc x TareaGit x POO/Tarea x Version C x Using the x + -

https://classroom.udacity.com/courses/ud123/lessons/61776182-d3f-4767-9558-5d1591c1a709/co...

Clase 5: Tagging, Branching, and Merging

BUSCAR

RECURSOS

CONTENIDO

- 1. Intro
- 2. Tagging
- 3. Branching
- 4. Branching Effectively
- 5. Merging
- 6. Merge Conflicts
- 7. Outro

Branching Effectively

SEND FEEDBACK

Don't worry, it's still there, we'll see how to get it to display in just a second.

But first, in your code editor, switch to the `app.css` file. Notice that it does not have the CSS we previously entered! Because of this, if you load the project up in the browser, the page won't have a colored background. This makes sense since the CSS file is empty, but do you know why?

QUESTION 1 OF 2

Thinking back to the branching repository video from the previous lesson, why would the CSS file be empty?

- ☐ The content has been erased.
- ☒ The content is stored safely on another branch.
- ☐ The content is in a temporary file that needs to be saved.
- ☐ A bear ate it.

ENVIAR

11:48 p. m. 28/02/2022

Introducción

Git Home

TareaGit

TareaGit

POO/TareaGit

TareaGit

POO/TareaGit

Version Control

Using the

https://classroom.udacity.com/courses/ud123/lessons/61776182-d3f-4767-9558-5d1591c1a709/co...

Clase 5:
Tagging, Branching, and Merging

BUSCAR

RECURSOS

CONTENIDO

1. Intro

2. Tagging

3. Branching

4. Branching Effectively

5. Merging

6. Merge Conflicts

7. Outro

Branching Effectively

SEND FEEDBACK

the new sidebar is no longer there(!) because all that code is stored safely on the `sidebar` branch.)

Now change the `<h1>` heading of the page from "Expedition" to something else. How about something exciting like the word "Adventure"!?

QUESTION 2 OF 2

Pop quiz time! How do you have Git show you the changes you've saved, but not yet committed?

☐ `git show --diff`

☐ `git log -p`

☒ `git diff`

☐ `git log --stat`

ENVIAR

Windows Taskbar

11:48 p. m.
28/02/2022

Introducción

Git Home

TareaGit

TareaGit

POO/Ejerc

TareaGit

POO/Tarea

Version Control

Using the

https://classroom.udacity.com/courses/ud123/lessons/61776182-d3f-4767-9558-5d1591c1a709/co...

Clase 5:
Tagging, Branching, and Merging

BUSCAR

RECURSOS

CONTENIDO

1. Intro

2. Tagging

3. Branching

4. Branching Effectively

5. Merging

6. Merge Conflicts

7. Outro

Merging

SEND FEEDBACK

branch_

QUESTION 1 OF 2

Take a look at the repository, again, now that you've merged the two branches together. Do the `master` branch and the `footer` branch point to the same commit?

☒ Yes

☐ No

ENVIAR

Perform A Regular Merge

Fantastic work doing a Fast-forward merge! That wasn't too hard, was it?

But you might say - "Of course that was easy, all of the commits are already there and the branch pointer just moved forward!"...and you'd be right. It's the simplest of merges.

Introducción

Git Home

TareaGit

TareaGit

POO/Ejerc

TareaGit

POO/Tarea

Version Control

Using the

https://classroom.udacity.com/courses/ud123/lessons/61776182-d3f-4767-9558-5d1591c1a709/co...

Clase 5:
Tagging, Branching, and Merging

BUSCAR

RECURSOS

CONTENIDO

1. Intro

2. Tagging

3. Branching

4. Branching Effectively

5. Merging

6. Merge Conflicts

7. Outro

Merging

SEND FEEDBACK

QUESTION 2 OF 2

How about a quick quiz to see if you're following along with how merging works.

Let's say a repository has 4 branches in it:

- `master`
- `allisons-mobile-footer-fix`
- `nav-updates`
- `jonathans-seo-changes`

The changes on `master` and `allisons-mobile-footer-fix` need to be merged together. If HEAD points to `allisons-mobile-footer-fix`, which branch will move when the merge is performed?

☐ master

☒ allisons-mobile-footer-fix

☐ nav-updates

Introducción x Git Home x TareaGit x TareaGit x POO/Ejerc x TareaGit x POO/Tarea x Version C x Using the x

https://classroom.udacity.com/courses/ud123/lessons/61776182-d3f-4767-9558-5d1591c1a709/co...

Clase 5: Tagging, Branching, and Merging

BUSCAR

RECURSOS

CONTENIDO

- 1. Intro
- 2. Tagging
- 3. Branching
- 4. Branching Effectively
- 5. Merging
- 6. Merge Conflicts
- 7. Outro

Merge Conflicts

SEND FEEDBACK

changed on both branches (`index.html`), but that there was a conflict. Also, notice that it tells you what happened - "Automatic merge failed; fix conflicts and then commit the result".

Remember our good friend `git status`? Well he'll come in really handy when working with merge conflicts.

QUESTION 1 OF 2

Try running `git status` right now. Which of the following information does it show?

- ☒ You have unmerged paths
- ☒ Unmerged paths
- ☒ fix conflicts and run "git commit"
- ☒ use "git add ..." to mark resolution

ENVIAR

Windows taskbar: 11:52 p. m., 28/02/2022

Introducción x Git Home x TareaGit x TareaGit x POO/Ejerc x TareaGit x POO/Tarea x Version C x Using the x

https://classroom.udacity.com/courses/ud123/lessons/61776182-d3f-4767-9558-5d1591c1a709/co...

Clase 5: Tagging, Branching, and Merging

BUSCAR

RECURSOS

CONTENIDO

- 1. Intro
- 2. Tagging
- 3. Branching
- 4. Branching Effectively
- 5. Merging
- 6. Merge Conflicts
- 7. Outro

Merge Conflicts

SEND FEEDBACK

will pop open your code editor for you to supply a commit message. Just like before, it's common to use the provided merge commit message, so after the editor opens, just close it to use the provided commit message.

And that's it! Merge conflicts really aren't all that challenging once you understand what the merge conflict indicators are showing you.

QUESTION 2 OF 2

You've made numerous commits so far in your exploration of Git. If a merge conflict occurs in a file and you edit the file, save it, stage it, and commit it but *forget* to remove the merge conflict indicators, will Git commit the file?

- ☒ Yes
- ☐ No

ENVIAR

Windows taskbar: 11:53 p. m., 28/02/2022

Introducción x Git Home x TareaGit x TareaGit x POO/Ejerc x TareaGit x POO/Tarea x Version Control x Using the x

https://classroom.udacity.com/courses/ud123/lessons/02167ad-3ba7-40e0-a157-e5320a5b0dc8/co...

Clase 6: Undoing Changes

BUSCAR

RECURSOS

CONTENIDO

- ✓ 1. Intro
- ✓ 2. Modifying The Last Commit
- ✓ 3. Reverting A Commit
- ✓ 4. Resetting Commits
- 5. Lesson Outro
- 6. Course Outro

Resetting Commits

SEND FEEDBACK

QUESTION 1 OF 4

Which commit is referenced by HEAD~6?

- ☐ 4c9749e
- ☐ 0c5975a
- ☐ 1a56a81
- ☐ f69811c
- ☐ e014d91
- ☒ 209752a

ENVIAR

Windows taskbar: 11:54 p. m. 28/02/2022

Introducción x Git Home x TareaGit x TareaGit x POO/Ejerc x TareaGit x POO/Tarea x Version Control x Using the x

https://classroom.udacity.com/courses/ud123/lessons/02167ad-3ba7-40e0-a157-e5320a5b0dc8/co...

Clase 6: Undoing Changes

BUSCAR

RECURSOS

CONTENIDO

- ✓ 1. Intro
- ✓ 2. Modifying The Last Commit
- ✓ 3. Reverting A Commit
- ✓ 4. Resetting Commits
- 5. Lesson Outro
- 6. Course Outro

Resetting Commits

SEND FEEDBACK

209752a

ENVIAR

You did so well on that last one, why not give this one a go! Using the same repository, which commit is referenced by HEAD~4^2?

f69811c

The `git reset` Command

The `git reset` command is used to reset (erase) commits:

```
$ git reset <reference-to-commit>
```

Windows taskbar: 11:55 p. m. 28/02/2022

Introducción x Git Home x TareaGit x TareaGit x POO/Ejer x TareaGit x POO/Tare x Version C x Using the x

https://classroom.udacity.com/courses/ud123/lessons/02167ad-3ba7-40e0-a157-e5320a5b0dc8/co...

Clase 6: Undoing Changes

BUSCAR

RECURSOS

CONTENIDO

- 1. Intro
- 2. Modifying The Last Commit
- 3. Reverting A Commit
- 4. Resetting Commits
- 5. Lesson Outro
- 6. Course Outro

Resetting Commits

SEND FEEDBACK

QUESTION 3 OF 4

What will happen to the changes from the `3772ab1` commit if `git reset --hard HEAD~3` is run? Will the changes be in the Staging Index, in the Working Directory, or complete erased?

☐ Staging Index

☐ Working Directory

☒ erased

ENVIAR

QUESTION 4 OF 4

What will happen to the changes from the `209752a` commit if

Introducción x Git Home x TareaGit x TareaGit x POO/Ejer x TareaGit x POO/Tare x Version C x Using the x

https://classroom.udacity.com/courses/ud123/lessons/02167ad-3ba7-40e0-a157-e5320a5b0dc8/co...

Clase 6: Undoing Changes

BUSCAR

RECURSOS

CONTENIDO

- 1. Intro
- 2. Modifying The Last Commit
- 3. Reverting A Commit
- 4. Resetting Commits
- 5. Lesson Outro
- 6. Course Outro

Resetting Commits

SEND FEEDBACK

QUESTION 4 OF 4

What will happen to the changes from the `209752a` commit if `git reset --soft HEAD^^` is run? Will the changes be in the Staging Index, in the Working Directory, or complete erased?

☒ Staging Index

☐ Working Directory

☐ erased

ENVIAR

Reset Recap

To recap, the `git reset` command is used erase commits:

```
$ git reset <reference-to-commit>
```


- ✓ Video: What is a remote? 4 min
- ✓ Video: Working with Remotes 3 min
- ✓ Video: Fetching New Changes 3 min
- ✓ Video: Updating the Local Repository 3 min
- ✓ Lectura: Git Remotes Cheat Sheet 10 min
- ✓ Cuestionario práctico: Practice Quiz: Using a Remote Repository 5 preguntas

Solving Conflicts

Module Review

Practice Quiz: Using a Remote Repository

Cuestionario Práctico • 10 min

✓ Envía tu tarea

Reanudar tarea

✓ Recibe la calificación

Para Aprobar 80 % o más

Tu calificación

100 %

Ver comentarios

Conservaremos tu puntaje más alto.

👍 Like

💬 Dislike

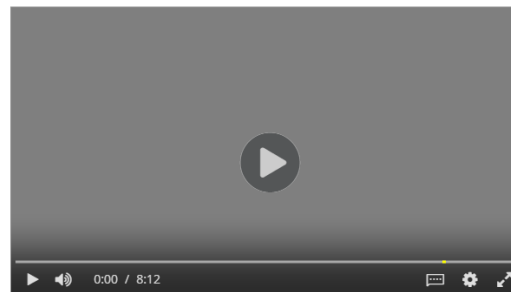
📄 Informar de un problema

- ✓ Video: Pushing Remote Branches 8 min
- ✓ Video: Rebasing Your Changes 4 min
- ✓ Video: Another Rebasing Example 8 min
- ✓ Video: Best Practices for Collaboration 4 min
- ✓ Lectura: Conflict Resolution Cheat Sheet 10 min
- ✓ Cuestionario práctico: Practice Quiz: Solving Conflicts 5 preguntas

Module Review

Pushing Remote Branches

📄 Notas 💬 Discutir



coursera

Bienvenido al curso de Introducción a Git y GitHub

Introducción a Git y GitHub > Semana 3 > Discussion Prompt

Introduction to GitHub

Using a Remote Repository

Solving Conflicts

Module Review

- ☒ Video: Module 3 Wrap Up:
Working with Remotes
1 min
- Herramienta externa con calificación: Qwiklabs
Assessment: Introduction to Github
1 h 30 m
- Cuadro de aviso de la discusión: Discussion Prompt
10 min

Luis Felipe Martinez · hace 6 horas

nothing special

0 Votos a favor Responder

JM Garcia · hace 11 horas

Good

0 Votos a favor Responder

Kanharaju Pandurangappa · hace 11 horas

0 Votos a favor Responder

Jamal Najeeb Mohsen Mahdi · hace 12 horas

this is nice ...