Git Rebase vs Git Merge: Which is Better?

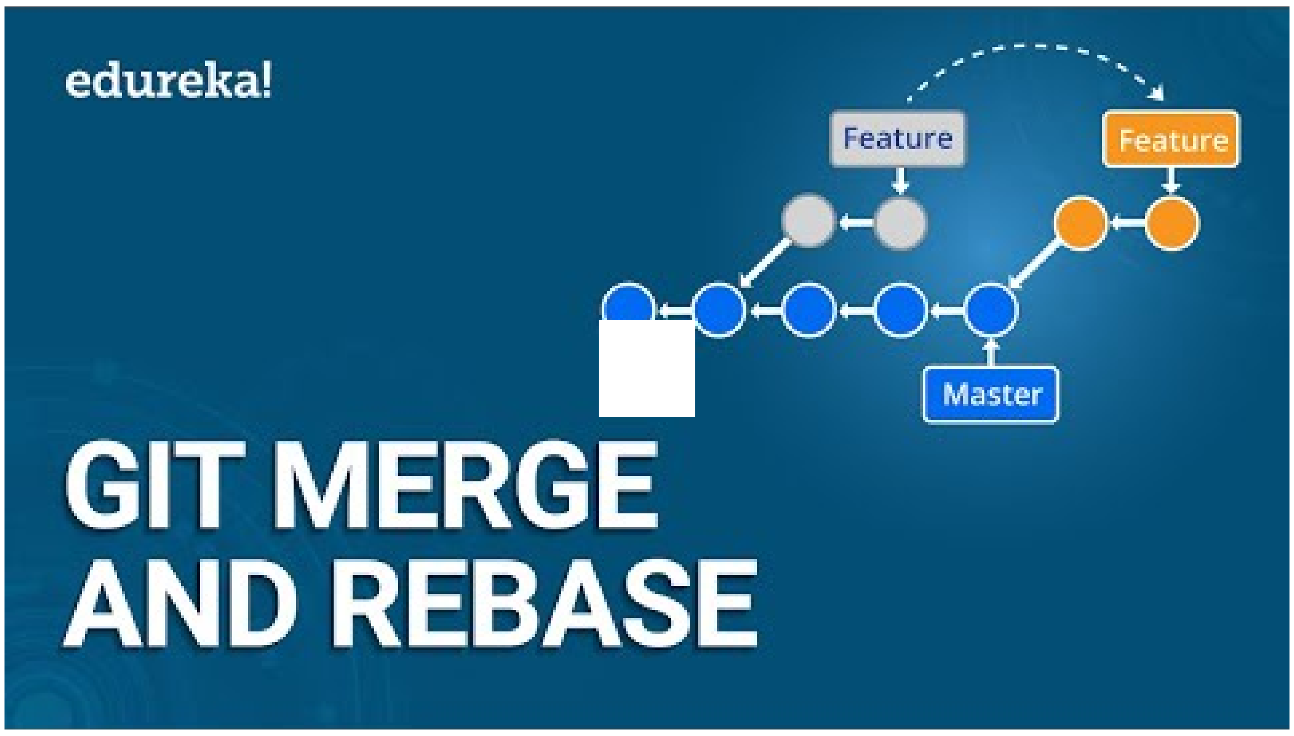
Published on Jan 18,2022 *74.1K Views* Share

Tanishqa Chaudhary
An intellectual brain with a strong urge to explore different upcoming technologies,...

There's a lot of debate about git merge and git rebase about which is better. Today in the blog on Git Rebase vs Merge we will clear all your doubts about Git Rebase and Git Merge. Both the techniques are used for the same purpose, so it is a bit tricky to understand them because of their similarities. By the end of the blog, you will get to know when to use Git Rebase vs Merge.

Git Merge and Git Rebase commands are used to combine the work of multiple developers in one code. The end objective for both these commands is same, but their usage varies. Today in this blog, we will try to understand Git Merge vs Git Rebase.

If you're not into reading, here's a video on Git Rebase vs Merging which will help you understand which is written in this blog.



So, the following are the topics covered in this blog:

How does Git Work?

For understanding the working of git, we need to understand the two fundamental concepts in git which is git commit and git branch. Let's understand these two terms respectively.

What is a Commit?

Commit is defined as the location where the code and its changes are stored. Let us take an example and discuss in brief from the diagram shown below:

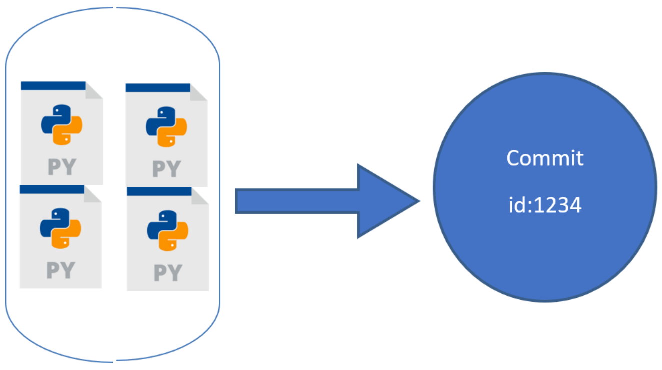


Fig 1: The changes made are saved in a commit

In Fig: 1, Let us assume that we have four python files. We saved them on git. These four python files will be saved inside a commit. Each commit has a commit-id, let it be 1234 in our case. Now let us say we have made some changes to the code by adding another python file. These changes in git will be saved as another commit with another commit-id 14343. This can be seen

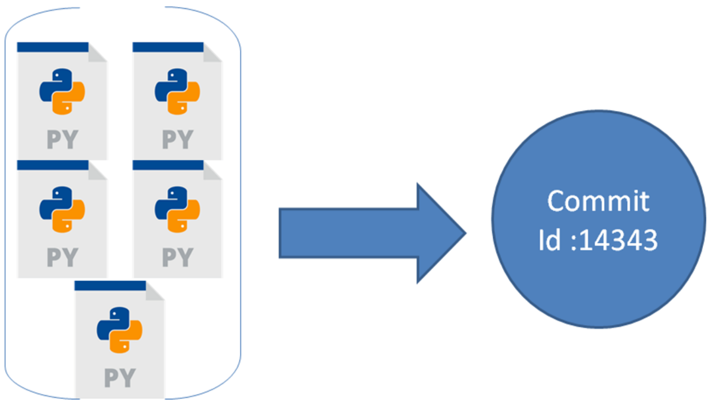
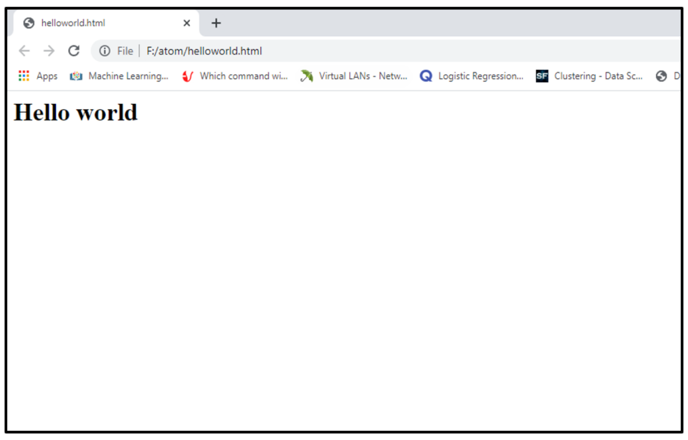


Fig 2: The changes made in the repository will be saved as a new commit with a new commit id

What is a Branch?

A Branch is a representation of different isolated versions of code. Let us take an example to understand this, Let's say you have a website that is currently running. The website looks something like this.



Hello world

Fig 3: Sample example of the website before changes

You want to add more features to this website. For this, you will have to change the code of the website. But if you are changing the code, you do not want the changes to be reflected in the main website which is deployed. So, what do you do? Ideally, you will copy the code of this website in a new folder. Make changes in the code, and then once the changes are finalised, you will replace the code in the main folder right? Let us understand how we can do the above thing in git.

So, in git to isolate different versions of code we have branches. By default, all your code is stored on a master branch. So, the website we showed you above is the master branch. Now, we don't want to touch the master branch code, we want to copy the code of the master branch to a new place, where we can experiment or change the code. Hence, not affecting the master branch. So, we create a new branch from the master branch, let's call it a 'feature branch. Now, any new changes that you will do on the feature branch, will not affect the code on the master branch.

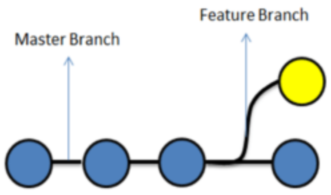


Fig 4: Master branch and the Feature branch

Once you are done with the changes, we simply 'merge' the changes of the 'feature' branch to the 'master branch'. And now, the changes which were made in the feature branch will exist on the master branch as well.

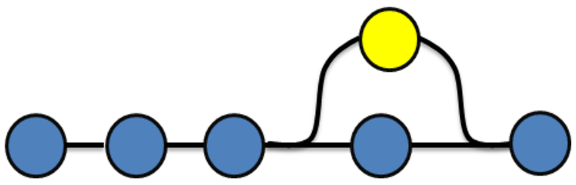
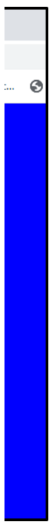


Fig 5: Merging of feature branch with the master branch

If we consider the above website example after merging changes can be seen in the diagram below:



ɔn the website.

ꞔ of two branches, feature
branch.

l a feature branch, after
ıd B. We also did some

ıanges of Commit A and B

ꞩ from Commit 1,2, A, and B,

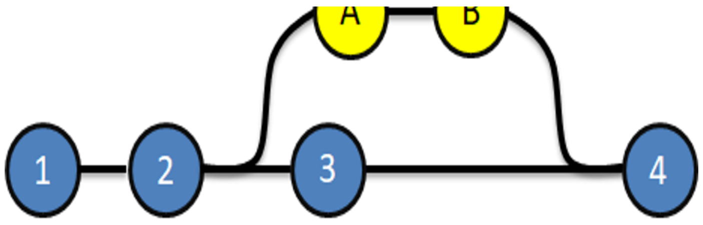
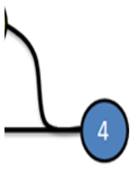


Fig 8: After merging

We merged the feature branch on master, resulting in commit 4. Commit 4 on the master, has all the changes of the code i.e Commit 1,2, 3, A, and B. Now, that we understand merging, let's understand the different types of merging that we can perform in git. In Git, merging is of two types:

- Git Merge
- Git Rebase

Let's understand both of them in detail



hen each merge happened

using git-merge

was introduced to overcome

nd feature branch commits as
commit A and B. If we perform a git rebase operation then the commits A and B will be rebased on to the master branch as commit 4 and 5 and there will be no logs of the feature branches. This is depicted in Fig 12.

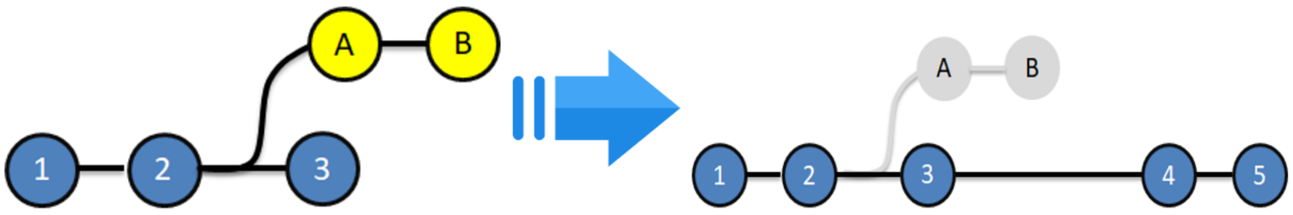


Fig 12: Before and After git rebase

Advantages:

- The logs are linear
- It's easy to move through the project.

Disadvantages:

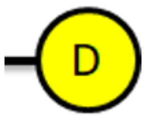
: target branch is private

ebase

ebase

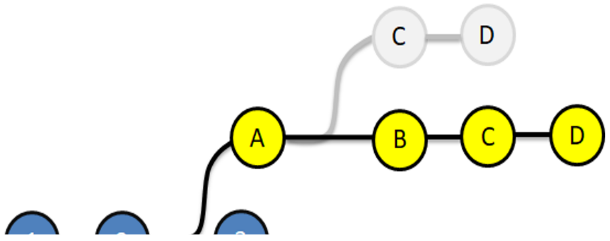
branch. However, in the git-
: merge when we want our
se Git Rebase when the logs
e are working on branches,
h can be viewed by other

,3, and 2 feature branches
g 14.



To experiment with the code, he creates another branch Feature branch 2, does some changes in it, and finalizes it with Commits C and D.

He does not want anyone to know about his private branch, because it's unnecessary. So, he can rebase Feature 2 on Feature 1.



ıde on feature branch 1,

out Git Merge vs Git Rebase
Git Tutorial blog for more

’s Devops Certification

What is DevOps – A Beginners Guide To DevOps

Watch Now

What is Docker – DevOps Tool For Containerization

Watch Now

Continuous Integration With Jenkins

Watch Now

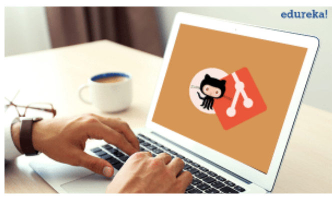
DevOps-Redefining y Strategy

Watch Now

‹›

Top Jenkins Interview Questions You Must Prepare In 2023

Read Article

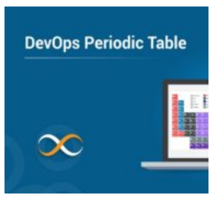


‘Git’ting Ahead: Hacking Git and GitHub Part 2

Read Article

Exploring Ansible Tower With A Hands-On

Read Article



DevOps Periodic Tab Ultimate Cheat Shee

Read Article

‹›

1 Comment

Reply

Trending Courses in DevOps

Git Rebase vs Git Merge: Which is B

×

Subscribe to our Newsletter, and get personalized recommendations.

Already have an account? Sign in.

FREE WEBINAR
Learn How To implement Kubernet...

ig | Cyber Security

lopment

ict Management

re Testing

n

ice & Machine Learning

ing

sting |

| Artificial Intelligence |

Digital Marketing

JavaScript interview questions |

rs in C | Linux commands |

l | Python tutorial |

What Is cloud computing |

ig in java |

Python Programming Language | Python interview questions | Multithreading in java | ReactJS Tutorial | Data Science vs Big Data vs Data Analyt... |

Software Testing Interview Questions | R Tutorial | Java Programs | JavaScript Reserved Words and Keywor... | Implement thread.yield() in Java: Exam... |

Implement Optical Character Recogniti... | All you Need to Know About Implement...

Terms & Conditions

Legal & Privacy

Git Rebase vs Git Merge: Which is Better?

×

Subscribe to our Newsletter, and get personalized recommendations.

Already have an account? Sign in.