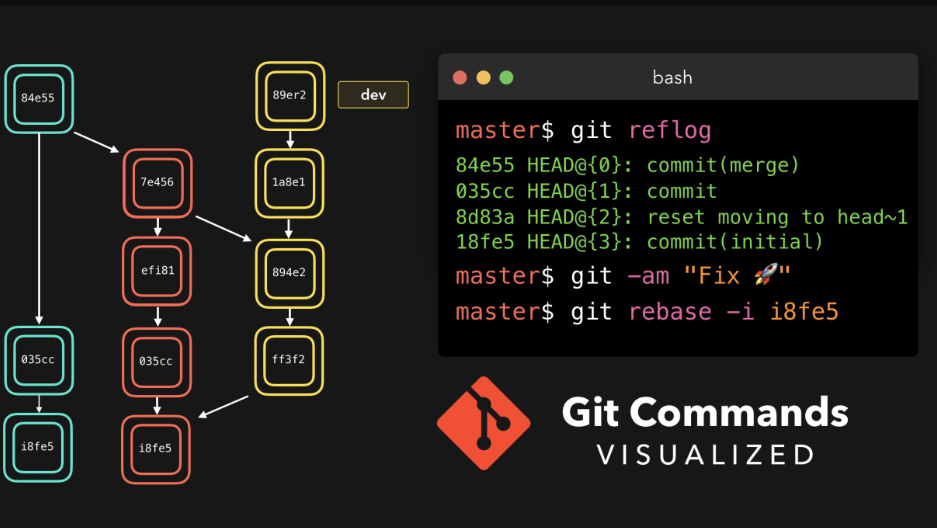
**StarLabs 2022 - Documentation**

**Git (Version Control)**

GitHub is a code hosting platform for version control and collaboration. It lets you and others work together on projects from anywhere.



***Supported languages:***

Most GitHub features work regardless of which languages your code is written in. You can search for code or enable syntax highlighting based on any language known to GitHub.

***Core languages supported by GitHub features:***

Core languages for GitHub features include C, C++, C#, Go, Java, JavaScript, PHP, Python, Ruby, Scala, and TypeScript. For features that support package managers, the currently supported package managers are included in the table with their relevant languages. [See the table (Ctril+Click)](https://docs.github.com/en/get-started/learning-about-github/github-language-support)

***The Advantages and Disadvantages of Using GitHub:***

***Advantages:***

1. **Markdown:** Basically, Markdown allows you to use a simple text editor to write formatted documents. GitHub, like many online repo services, supports Markdown for the issue tracker, user comments, wikis everything. With so many other programming languages to learn for setting up projects, it’s really a big benefit to have your content inputted in a format without having to learn yet another system. In addition, there is also what is known as the GitHub flavored markdown a feature that adds changes to the usual markdown in order to make it more useful in programming environments
2. **GitHub has some of the best documentation around:** You won’t run out of content when you use GitHub, thanks to a well-padded guide and help section for articles that you can pull up for practically any topic on earth, for as long as it is related to a git. It’s got content for helping you learn about generating SSH keys. A guide for the best git workflow is available. Samples on gitignore (and more) are abound for your next planned project, among other things. You would not need to look elsewhere for all the information that you need**.**
3. **GitHub has Gists and GitHub Pages, too:** A while back, GitHub rolled out a feature called Gists, which lets you convert one or several files into a working git repository. This new feature converted sharing and tracking changes made to configuration files and even simple scripts into a whole new level of easy. While they aren’t as rich in features like a full-blown GitHub repository, they really work well even if you are without a paid account. GitHub pages, on the other hand, lets you host static websites by simple assigning HTML pages onto another, separate repository – the way you would any other type of git repository. With this, blogging can be done off the bat as well as updating with additional documentation or bumping up its web presence
4. **Collaboration:** For those who are not in the same physical location, an online Git is an easy solution requiring no setup for new users. With no need to connect to the company’s VPN, it may be easier to dump everything on a private repository on GitHub.

This benefit is much greater to those working collaboratively on a project that are not part of a professional environment – particularly open source projects. Most programmers are already familiar with how to use GitHub, and it’s easy to point people to a GitHub page if they want to make contributions. Online repositories are essential for open source projects, and the only reason some may avoid GitHub was acquired by Microsoft some time ago, which resulted in many switching to alternative like [GitLab](https://about.gitlab.com/). While there’s no current reason in particular for this switch, many do not trust Microsoft’s track record of attacking the open source community in the past.

1. **Backup:** Using an online repository should never be considered infallible, but it provides a nice and simple way to have their code and version history available online, regardless of what happens to their local machine. For some people, this is enough, but we stress that a multi solution backup plan is always the best.

***Disadvantages:***

1. **Security:** GitHub does offer private repositories, but this isn’t necessarily perfect for many. For high value intellectual property, you’re putting all of this in the hands of GitHub as well as anyone who has a login, which like many sites has had security breaches before and is targeted constantly. It is often better than nothing, but it’s not perfect. In addition, some clients/employers will only allow code on their own secure internal Git as a matter of policy.
2. **Pricing:** Some of GitHub features, as well as features on other online repositories, are locked behind a SaaS paywall. If you have a large team, this can add up fast. Those who already have a dedicated IT team and their own internal servers are often better off using their own internal git for cost reasons, but for most the cost isn’t outrageous.