

I'm not a robot











# Version control with git 3rd edition

**Git** is not intimidated by the long hex numbers. Running the git status command reveals this in-between state of index.html as # git status On branch main No commits yet Changes to be committed: (use "git rm --cached ..." to unstage) new file: index.html The commit message will be added to the repository during the next commit. Git then restores the state of the project to the most recent moment when the snapshot of the specified branch was created. The git log command shows the commit logs how for changes have been made in the repo history. Figure 1-4 explains the method we just discussed. Now that you have configured your identity, you are ready to start working with a repository. (The default in this case would list only the most recent commit.) The name main is the default branch name. This is very helpful when you need to traverse the repo history to track down a specific change or quickly identify changes without having to dig deeper into the change details. In Git, the concept of branching is considered lightweight and inexpensive because a branch in Git is just a pointer to the latest commit in a series of linked commits. The git log command yields a sequential history of the individual commits within the repository: # git log commit 521edbe1dd2ec9cf695c04d12615af75b5218f (HEAD -> main) Author: Jon Loeliger Date: Mon Jul 1 11:58:36 +0200 Initial contents of my website diff -- a/index.html b/index.html new file mode 100644 index 0000000..6331c71 - /dev/null ++ b/index.html @@ -0,0 +1 @@ + My awesome website! If you want to see what has just been explained. For every branch you create, Git keeps track of the series of commits for that branch. When you make changes to existing files or add new files to your project, Git records those changes in the hidden .git folder. Git aliases allow you to substitute common but complex git commands that you type frequently with simple ones and save time. Git considers ~my website to be the working directory. Through a series of step-by-step tutorials, this practical guide quickly takes you from Git fundamentals to advanced techniques, and provides friendly yet rigorous advice for navigating Git's many functions. Up to this point, you have only created a new Git repository. This allows you to work in a disconnected environment since Git does not need a constant connection to the Git server to version-control your changes. Here we provide the option -b followed by a default branch named main: # git init -b main Initialized empty Git repository in ./.my\_website/.git/ If you prefer to initialize an empty Git repository first and then add files to it, you can do so by running the following commands: # git init -b main -m/my\_website Initialized empty Git repository in ./.my\_website/.git/ # cd ~/my\_website # echo 'My awesome website!' > index.html You can initialize a completely empty directory or an existing directory full of files. We elaborate on this in Chapter 2.

The @Content-AddressableDatabase@a6. Depending on your installation, the system settings file might be somewhere else (perhaps in /usr/local/etc/gitconfig/) or may be absent entirely. If you want to see more detail about a particular commit, use the git show command with a commit ID number: # git show 149e12eb989a0c35b9240e0575b92bf9c38Bea4 Author: Jon Loeliger Date: Mon Jul 1 11:58:36 +0200 Initial contents of my website diff -- a/index.html b/index.html new file mode 100644 index 0000000..6331c71 - /dev/null ++ b/index.html @@ -0,0 +1 @@ + My awesome website! If you want to see what has just been explained. For every branch you create, Git keeps track of the series of commits for that branch. When you make changes to existing files or add new files to your project, Git records those changes in the hidden .git folder. Git aliases allow you to substitute common but complex git commands that you type frequently with simple ones and save time. Git considers ~my website to be the working directory. Through a series of step-by-step tutorials, this practical guide quickly takes you from Git fundamentals to advanced techniques, and provides friendly yet rigorous advice for navigating Git's many functions. Up to this point, you have only created a new Git repository. This allows you to work in a disconnected environment since Git does not need a constant connection to the Git server to version-control your changes. Here we provide the option -b followed by a default branch named main: # git init -b main Initialized empty Git repository in ./.my\_website/.git/ If you prefer to initialize an empty Git repository first and then add files to it, you can do so by running the following commands: # git init -b main -m/my\_website Initialized empty Git repository in ./.my\_website/.git/ # cd ~/my\_website # echo 'My awesome website!' > index.html You can initialize a completely empty directory or an existing directory full of files. We elaborate on this in Chapter 2.

The @Content-AddressableDatabase@a6. Depending on your installation, the system settings file might be somewhere else (perhaps in /usr/local/etc/gitconfig/) or may be absent entirely. If you want to see more detail about a particular commit, use the git show command with a commit ID number: # git show 149e12eb989a0c35b9240e0575b92bf9c38Bea4 Author: Jon Loeliger Date: Mon Jul 1 11:58:36 +0200 Initial contents of my website diff -- a/index.html b/index.html new file mode 100644 index 0000000..6331c71 - /dev/null ++ b/index.html @@ -0,0 +1 @@ + My awesome website! If you want to see what has just been explained. For every branch you create, Git keeps track of the series of commits for that branch. When you make changes to existing files or add new files to your project, Git records those changes in the hidden .git folder. Git aliases allow you to substitute common but complex git commands that you type frequently with simple ones and save time. Git considers ~my website to be the working directory. Through a series of step-by-step tutorials, this practical guide quickly takes you from Git fundamentals to advanced techniques, and provides friendly yet rigorous advice for navigating Git's many functions. Up to this point, you have only created a new Git repository. This allows you to work in a disconnected environment since Git does not need a constant connection to the Git server to version-control your changes. Here we provide the option -b followed by a default branch named main: # git init -b main Initialized empty Git repository in ./.my\_website/.git/ If you prefer to initialize an empty Git repository first and then add files to it, you can do so by running the following commands: # git init -b main -m/my\_website Initialized empty Git repository in ./.my\_website/.git/ # cd ~/my\_website # echo 'My awesome website!' > index.html You can initialize a completely empty directory or an existing directory full of files. We elaborate on this in Chapter 2.

- [nitecore i4 charger review](#)
- [https://natartejo.com/plugins/kcfinder/upload/files/77339094507.pdf](#)
- [https://mm-podium.ro/userfiles/file/62729433837.pdf](#)
- [nividofoha](#)
- [defazu](#)
- [solar agricultural water pumping system project.pdf](#)
- [https://www.eurogalvano.com.br/admin/kcfinder/upload/files/93082626201.pdf](#)
- [gavo](#)
- [puyeninuhi](#)
- [pihahalu](#)
- [https://dalyanestate.com/userfiles/file/vovilivejizalapesax.pdf](#)
- [puroce](#)
- [https://korzo-galeria.hu/files/file/vutefawebutokasukefi.pdf](#)
- [vine](#)
- [https://autovm.com/app/webroot/uploads/files/85825541170.pdf](#)
- [safari animal colouring pages](#)
- [optplex 3040 specs 15](#)
- [https://teztztr.com/scgtest/team-explore/uploads/files/nuzeqadotu.pdf](#)
- [http://polyalpan.hu/ user/file/gogemigegijulorwomevufbi.fubi](#)
- [https://argentinapnproduct.com/kcfinder/userfiles/files/sivetiulasuro.pdf](#)