



Bantha Inc.

September 25, 2023

Git Guide

How to Git

Author: Die Go

Contents

Table of Content	ii
Revision History	iii
List of Figures	iv
List of Table	v
List of Listings	vi
1 Introduction	1
1.1 Initializing	1
1.2 Cloning	2
2 Results	3

Revision History

Date	Version	Author	Description
08SEP2023	0.0.1	Diego F.	Template Created
09SEP2023	0.0.2	Diego F.	Executive Summary

List of Figures

List of Tables

Listings

1.1	Commands for init first repo	1
1.2	First Push	1
1.3	Branch Name Change	2

Chapter 1

Introduction

This is a guide of how to do Git. It is a really beginner thing as I will go through from how start a repo to some really cool stuff. I try to explain what stuffs does.

1.1 Initializing

First on creating a repo, create a gitignore for it. Google the right gitignore for respective program. Like Latex and KiCad stuff. Gitignore is a file that specifies files to not be pushed to the repo.

```
1 touch .gitignore
2
3 git init
4
5 git add .
6
7 git commit -m "Msg"
8
9 git remote add origin hithubURL
10
11 git branch -M master
```

Listing 1.1: Commands for init first repo

On the first push do:

```
1 git push -u origin master
```

Listing 1.2: First Push

1. git push: This is the main Git command used for pushing changes to a remote repository. It takes the changes you've committed locally and sends them to the specified remote repository.
2. -u (or --set-upstream): This option is used to set up a tracking relationship between the local branch (in this case, "master") and the remote branch (in this case, "origin/master"). Setting up a tracking relationship means that in the future, when you use git push or git pull without specifying a branch name, Git will know which remote branch to push to or pull from. It's a convenient way to streamline your workflow.

3. origin: This is the name of the remote repository you are pushing your changes to. "Origin" is a commonly used default name for the remote repository from which you initially cloned your local repository. You can have multiple remotes if your project requires it, and you'd use the appropriate remote name here.
4. master: This is the name of the local branch that you want to push to the remote repository. In Git, "master" is a default branch name, but your project may use a different branch name as its main branch.

So, when you run `git push -u origin master`, you are pushing the changes from your local "master" branch to the "master" branch in the remote repository called "origin", and you are setting up tracking so that future pushes and pulls can be done without specifying branch names if desired. If the remote branch is named differently then use:

```
1  git branch -m master main
2  or
3  git branch -M main
```

Listing 1.3: Branch Name Change

1.2 Cloning

Chapter 2

Results