Git Documentation

Alejandro Leon Marin

25/02/2024

Introduction

In this document we will be seeing the basics of git and how to use it. To promete the use and the knowledge of this tool. ALso it will function as a guide for the use of git.

What is Git?

First let's start by defening git. So Git is a tool that helps us to keep track of the changes that we do in our code. So as it is. Git is a control version system tool. Most of the developers use it because it make more easy to work in team. Imagine the situacion that you are working in a team and have a very big project. So you start the basics structure of the project, imagine the situation that now you have to share it with your team. You have 3 options, tell your team how to create the project with the same structure that you have. Second option, compress your project and send the project to your team or the third option is to upload the project to github or gitlab and share the link with your team and they clone the project using git. It seams that the third option is the best. Becuase is easier and quickly. Also it helps that the work in team be more organized. So this is git and we will learn how to use it. But first let's see what is github

Whats is Github?

Let's define Github as cloud service that helps us to store our projects. It is a web page that helps us to store our projects. It's also like a social network for developers. It helps us to share our projects with others developers. Also it helps a portfolio for the developers and it helps to find a job. So it is a very useful tool and this is where we will store our projects. So Github is not very difficult to use. We just need to understand the basics of how GitHub stores the projects. So let's see how to use it.

Reporsiories

GitHub use a structure for a project and let's say that this structure is like a box and into this box you will put all your code and all the files that you need in your project. So we have this box and this box is called repository. So in this repository we will have all the project. It's safe but we will have our project in a branch called main. So every time we change our code and the project isn's working, as we usually called it and unstable version, we don't like to have this version to the public. So we will learn about brancher and how useful they are.

Branches

So let's define branch. So see it as an computer science engineer. Let's see it as a tree. So as we know from Data structures a tree is a data structure that has a root and from the root we have branches that we can create and access. Also can see it as an arraylist where we have nodes and we can access to the nodes. So in this concept Github use the same concept. But in every branch we will have a different version of the project. So in the main branch we will have the stable version of the project and in the dev branch we will have the unstable version of the project. We can create the branches directly from the web page of github or we can create it with git. So know lets see how to use git.

How to use Git

So git is a tool that we can use in the terminal. It functions differently in windows and in linux. I use linux so I will show you how to use it in linux.But for windows when you install git, it will install a terminal called git bash, that is a terminal thet works as a linux terminal so with that terminal is with the terminal that you will work with git. So first we need to config git in our computer. So we will configure our name and our email, this to let know git who is the person that is making the changes. So let's see how to do it.

```
git config ——global user.name ”Your Name”
git config ——global user.email ”Your Email”
```

So as you can see we use the command git config and then we use the flag –global to let know git that we are configuring the global configuration of git. After we indicated the flag we use the flag user.name and user.email to indicate the name and the email of the person that is using git. That information have to be in double quotes. So know we have configured git. Also is important to check if git config is correctly configured. So we can use the command git config –list to see the information that we have configured. So know lets see how to create a repository in our computer.

To create a repository in our computer we need to use the command git init. This command will create a repository in the folder that we are, so it means that our repository will we created in the folder that we are. So let's see how to do it.

```
git init
```

When you use this command you will see that a folder called .git will be created. This folder is the folder that git use to store the information of the repository. So know we have created a repository. So let's see how to add files to the repository. To add the files to the repository we need to use the command git add. With git add you can add all the file using the next command.

```
git add .
```

If you want to add and specific file you can use the next command.

```
git add <filename.extension>
```

And if you want to add all the files that have a specific extension you can use the next command.

```
git add <.extension>
```

So those are the three ways to add files to the repository. So know we add the files to the repository. But we need to comment what is on those files or what we change in those files. To do that we use a message called commit message. This message has to be short and specific. To be readable to devs and to understant what we change. So let's see how to do it.

```
git commit –m ”Your message”
```

So with this command we add to the files a commit message. And this will we stored in the repository. So now we have our repository in our computer. But we need to store it in github. So let's see how to do it.

How to use Github

So to use Github we need to access the we page of github `https://github.com` and create an account. After we create the account we need to create a repository. So to create a repository we need to go to our profile and click on over the button that says Reporsiories. After we click on the button, we need to find a button that says new and after that we need to fill the information of the repository. So we need to put the name of the repository and a description. Also we can put the repository public or private. So after we fill the information we need to click on the button that says create repository. After we click on the button we will see the information of the repository. So we need to copy the link of the repository. So we can use the command git remote add origin ¡link¿ to add the repository to our computer. So let's see how to do it.

```
git remote add origin <link>
```

So with this command we add the remote repository to our computer. So know we need to push the repository to github. To do that we use the command git push. So let's see how to do it.

```
git push –u origin main
```

So with this command we push the repository to github. So know we have our repository in github. So know we can work in team. With this we can share our projects with others developers. So this is the basics of git and github.

So let's see other commands that we can use in git.

Other commands

To create new branches we use the git checkout command. So let's see how to do it.

```
git checkout –b <branchname>
```

So with this command we create a new branch. So know we can work in the new branch. So to change the branch we use the next command.

```
git checkout <branchname>
```

So with this command we change the branch. So know we can work in the new branch. So as we create a new branch we need to push it to github. To do that we use the next command.

```
git push origin <branchname>
```

So with this command we push the branch to github. So know we can work in team with this branch. And now if we have a stable version of the project in the dev branch we can merge the branch to the main branch. To do that we use the next command.

```
git merge <branchname>
```

So with this command we merge the branch to the main branch. So know we have the stable version of the project in the main branch. Also we can delete the branch. To do that we use the next command.

```
git branch –d <branchname>
```

So with this command we delete the branch. We also have git status, that will helps us to see the status of the repository on our local repository. So let's see how to use it.

```
git status
```

And this will list all of the files that we have in the repository that have changes and need to be update. Another command that we have is git log. This command will show us the history of the repository. To see the history of the repository we use the next command.

```
git log
```

Also we have the git clone command. This command will help us to clone a repository from github. So let's see how to do it.

```
git clone <link of the repository>
```

So with this command we clone the repository to our computer. And now to finish we have git pull command. This command will help us to update the repository in our computer. So let's see how to do it.

```
git pull
```

So with this command we update the repository in our computer.

Conclusion

So this are the basics of git and github. I hope that this document helps you to understand the basics of git and github. If you have any question you can contact me.