

Git e Github

Entendendo o que é Git e Github sua importância

O Git já existe um software pronto e atualizado para não ficarmos naquela dúvida de versão antiga ou versão nova ou versão final. Ele foi criado em 2005 que é um sistema de versionamento de código distribuído e foi criado pelo Linus Torvalds que também é criador do sistema Linux. O grupo que criou o sistema Linux estava espalhado pelo mundo todo, então Linus precisava de um sistema que versionasse, criasse versões do trabalho que ele estava criando ali e que fosse complexo o suficiente para que suportasse pessoas do mundo inteiro trabalhando no mesmo arquivo, e é isso que o Git soluciona.

Outras empresas surgiram criando o Github que é onde é guardado o versionamento desses códigos que é uma empresa da Microsoft. São duas tecnologias complementares, mas diferentes e vamos conhecer cada uma delas mais a fundo. Embora sejam muito bem utilizadas no mercado, o Git mesmo foi uma invenção tão bem-feita pelo Linus e sua equipe que ele é quase impossível você achar outra pessoa utilizando outra ferramenta. O Github que é o repositório online armazenaremos o nosso código e que é onde teremos uma competição mais acirrada envolvendo preços onde o Git é OpenSource (Aberto) e o Github pode ter planos pagos.

Comandos básicos para um bom desempenho no terminal

A maioria dos sistemas operacionais possuem programas que tem uma interface gráfica, ou seja, a forma que o usuário interage com esses programas é de forma gráfica podendo clicar, arrastar etc., sendo tudo responsivo ao usuário. O Git é voltado para outro tipo de utilidade, ele é um CLI (Command Line Interface) não tendo uma interface gráfica. A forma de interação nele é por linha de comando. Temos programas hoje que automatizam isso pegando o Git e acrescentando uma Gráfico User Interface tornando-o mais simples. Para isso, precisamos aprender primeiro alguns comandos que teremos que usar para utilizar o Git.

O que vamos aprender?

- Mudar de pastas
- Listar as pastas
- Criar pastas/arquivos
- Deletar pastas/arquivos

São comandos básicos para que possamos nos situar e usar a interface de CLI do terminal para conseguir interagir da melhor forma aprendendo os conceitos da forma mais certas. No Windows irão interagir com o terminal utilizando alguns comandos diferentes dos usuários que tem sistemas operacionais Unix (Sistemas derivados do Linux)

Primeiramente, vamos abrir o terminal clicando no botão do Windows e escrevendo CMD. Nosso primeiro comando a ser aprendido será para listar e nos situarmos em qual local estamos dentro do sistema operacional. No Windows esse comando se chama *dir* e ao digitarmos isso no CMD irá nos aparecer uma lista dos diretórios contido na pasta que estamos situados. Todos os comandos possuem variâncias, são “Flags” que passamos para esses comandos que acrescentam, modificam ou formatam a forma que esses comandos retornam para nós. Como eu subo de nível, desço ou vou para uma pasta específica no sistema operacional? Para isso, utilizamos o “Change Directory” (CD) que irá nos permitir navegar pelas pastas e é igual para todos os sistemas. Para isso, vamos digitar “cd /” no CMD que irá nos levar para a base do diretório C

```

Prompt de Comando
20/04/2021 15:20 <DIR> Favorites
20/04/2021 15:20 <DIR> Links
20/04/2021 15:20 <DIR> Music
07/12/2021 00:24 <DIR> OneDrive
09/10/2021 20:13 <DIR> Pictures
20/04/2021 15:20 <DIR> Saved Games
20/04/2021 15:22 <DIR> Searches
20/08/2021 23:42 <DIR> source
10/10/2021 22:01 <DIR> Videos
                    5 arquivo(s)          11.622 bytes
                   25 pasta(s) 134.715.420.672 bytes disponíveis

C:\Users\Igor Carrilho>cd /

C:\>
```

Se agora utilizarmos o comando dir que aprendemos, ele vai listar as pastas presentes dentro da base do diretório C.

```

Prompt de Comando

Pasta de C:\

13/09/2021 14:27 <DIR> Curso Udemy Perrini
10/09/2021 14:41 <DIR> eclipse
20/04/2021 16:03      35 END
31/10/2021 18:53 <DIR> Program Files
15/09/2021 15:28 <DIR> Program Files (x86)
31/03/2021 10:57 <DIR> TempPatchCD
20/04/2021 15:27 <DIR> Users
28/11/2021 23:25 <DIR> Windows
                    1 arquivo(s)          35 bytes
                    7 pasta(s) 134.860.824.576 bytes disponíveis

C:\>
```

Mas se quisermos utilizar uma entrada numa outra pasta do Windows por exemplo? Iremos digitar “cd Windows” e ai entraremos dentro da pasta Windows e se utilizarmos o dir, irá listar todos os arquivos e programas que temos dentro do diretório do Windows. Se quisermos voltar para fazer o caminho inverso? Utilizaremos o comando “cd ..”

```
Prompt de Comando

10/09/2021 14:41 <DIR> eclipse
20/04/2021 16:03 35 END
31/10/2021 18:53 <DIR> Program Files
15/09/2021 15:28 <DIR> Program Files (x86)
31/03/2021 10:57 <DIR> TempPatchCD
20/04/2021 15:27 <DIR> Users
28/11/2021 23:25 <DIR> Windows
1 arquivo(s) 35 bytes
7 pasta(s) 134.860.824.576 bytes disponíveis

C:\>cd Windows

C:\Windows>cd ..

C:\>_
```

Podemos reparar que a tela do CMD está muito “poluída” e outro comando que utilizamos muito para poder limpar isso é o “cls” que irá limpar o terminal. Um atalho muito útil no teclado é o TAB que tem uma função de auto completar. Até o momento aprendemos a listar os diretórios, caminhar para uma pasta específica, sair da pasta e agora vamos aprender de fato como criar uma pasta. O comando utilizado para isso é o “**mkdir**” (make directory). Vamos utilizar esse programa para criar uma pasta com o nome de “WorkSpace”. Não irá acontecer nada, que chamamos Silence on Success que é outro conceito que diz que se der tudo certo o terminal não irá falar nada.

```
Prompt de Comando

C:\>mkdir workspace

C:\>_
```

Se utilizarmos o dir para listar ele irá nos trazer a listra de diretórios contidos nessa pasta e no final teremos essa nova pasta que criamos.

```
Prompt de Comando

Pasta de C:\

13/09/2021 14:27 <DIR> Curso Udemy Perrini
10/09/2021 14:41 <DIR> eclipse
20/04/2021 16:03 35 END
31/10/2021 18:53 <DIR> Program Files
15/09/2021 15:28 <DIR> Program Files (x86)
31/03/2021 10:57 <DIR> TempPatchCD
20/04/2021 15:27 <DIR> Users
28/11/2021 23:25 <DIR> Windows
25/12/2021 19:36 <DIR> workspace
1 arquivo(s) 35 bytes
8 pasta(s) 134.854.508.544 bytes disponíveis

C:\>
```

Agora usaremos o comando “cd” para navegarmos para essa nova pasta criada. Agora vamos criar alguns arquivos nessa pasta para que possamos entender a diferença entre deletar arquivos e deletar repositórios. Então iremos criar o arquivo direto pelo terminal com o comando “**echo**” que simplesmente printa de volta no terminal o que digitamos. Vamos utilizar “echo hello >” onde esse sinal de maior é um redirecionador de fluxo, pegando o output que é a saída dessa função echo e vai jogar em um arquivo. O diretório ira procurar se no diretório já tem um arquivo com o nome que escolhemos e caso não tenha ele irá criar para nós.

```
C:\workspace>echo hello > hello.txt

C:\workspace>dir
O volume na unidade C é Acer
O Número de Série do Volume é 0EEB-3FF2

Pasta de C:\workspace

25/12/2021  19:43    <DIR>          .
25/12/2021  19:43    <DIR>          ..
25/12/2021  19:43                8 hello.txt
                1 arquivo(s)                8 bytes
                2 pasta(s) 134.607.294.464 bytes disponíveis

C:\workspace>
```

Vamos sair dessa pasta utilizando o “cd ..” e vamos listar para ficar didático. Então sabemos que dentro do diretório **workspace** temos o arquivo chamado **hello.txt** e vamos utilizar o comando “del workspace” e o CMD irá nos retornar uma pergunta se temos certeza que queremos deletar tudo que tem dentro dessa pasta.

```
C:\>dir
10/09/2021  14:41    <DIR>          eclipse
20/04/2021  16:03                35 END
31/10/2021  18:53    <DIR>          Program Files
15/09/2021  15:28    <DIR>          Program Files (x86)
31/03/2021  10:57    <DIR>          TempPatchCD
20/04/2021  15:27    <DIR>          Users
28/11/2021  23:25    <DIR>          Windows
25/12/2021  19:43    <DIR>          workspace
                1 arquivo(s)                35 bytes
                8 pasta(s) 134.661.783.552 bytes disponíveis

C:\>del workspace
C:\workspace\*, Tem certeza (S/N)? s

C:\>
```

Teoricamente a pasta foi deletada, mas se listarmos novamente com o dir, a pasta ainda estará aparecendo listada. Isso pois no Windows tem uma diferença entre deletar arquivos e repositórios. Esse comando ele é restritivo apenas para deletar arquivos, então ele deletou tudo o que havia dentro da pasta workspace e não era exatamente o que queríamos. Para deletarmos uma pasta iremos utilizar um atalho que é a seta pra cima para navegar no histórico de comandos que já executamos nesse terminal e vamos utilizar ele procurar o comando que utilizamos para criar o arquivo hello.txt para cria-lo novamente e termos ele novamente dentro da nossa pasta workspace. Agora temos novamente a pasta workspace com

o arquivo hello.txt lá dentro. Agora vamos aprender o comando para realmente remover o repositório com todos os arquivos que ele possui dentro dele. Esse comando é o rmdir e para ficar mais usual iremos utilizar duas Flags, sendo elas o /S e o /Q

```
C:\> Prompt de Comando
13/09/2021 14:27 <DIR> Curso Udemy Perrini
10/09/2021 14:41 <DIR> eclipse
20/04/2021 16:03 35 END
31/10/2021 18:53 <DIR> Program Files
15/09/2021 15:28 <DIR> Program Files (x86)
31/03/2021 10:57 <DIR> TempPatchCD
20/04/2021 15:27 <DIR> Users
28/11/2021 23:25 <DIR> Windows
25/12/2021 19:52 <DIR> workspace
1 arquivo(s) 35 bytes
8 pasta(s) 134.655.733.760 bytes disponíveis

C:\>rmdir workspace /S /Q

C:\>
```

Como não trouxe nenhum erro, provavelmente deu certo e se listarmos novamente veremos que o repositório foi removido.

Esses são os comandos básicos que iremos utilizar para poder navegar no terminal.

Windows	Unix
- cd	- cd
- dir	- ls
- mkdir	- mkdir
- del / rmdir	- rm -rf