

Git e Github

Implementação de um repositório



Índice

Introdução	3
Rastreamento de arquivos	3
Cópia de segurança	3

Introdução

Para iniciar um projeto com GIT, abriremos o Windows Power Shell ou qualquer console de comando com o qual trabalhamos, iremos para o diretório de trabalho, ou seja, o diretório onde temos nosso projeto e, a partir daí, executaremos o comando:

```
git init
```

Este comando só é executado uma vez, no início de nosso projeto ou quando queremos que a GIT comece a acompanhar nosso projeto.

Ao entrar neste comando através do console, o que vamos conseguir é que o GIT crie, de forma transparente para o usuário, duas áreas onde ele irá armazenar os arquivos. Estes são a área de encenação e a área de repositório local.

A **área de encenação**, como seu nome sugere, é uma área onde os arquivos serão armazenados temporariamente. Este passo é muito útil para que possamos ver quais arquivos são rastreados pelo GIT, em que estado estão... etc.

A **área de repositório local** é onde os instantâneos que a GIT retira de nossos arquivos são armazenados e é aqui que podemos recuperá-los sempre que quisermos.

Rastreamento de arquivos

O próximo passo seria dizer à GIT para rastrear todos os arquivos do meu projeto (ou apenas um específico).

Para rastrear todos os arquivos existentes no diretório do projeto, usaremos o comando:

```
git add .
```

Para adicionar apenas um arquivo específico, adicione o nome do arquivo:

```
git add nomeDoArquivo
```

Ao fazer isso, conseguimos que a GIT mova esses arquivos do diretório de trabalho para a área de preparação e acompanhe as mudanças para eles. Ele não os move fisicamente, os arquivos ainda estão no diretório de trabalho, mas internamente a GIT os trata como arquivos prontos para trabalhar na área de preparação.

Em seguida, faremos um:

```
git commit
```

Assim, nossos arquivos serão movidos (novamente de forma transparente para nós) para a área de repositório local.

É aqui que a GIT cria o instantâneo do estado atual de nossos arquivos. Ou seja, onde é criado o backup (a cópia) de nosso projeto.

Cópia de segurança

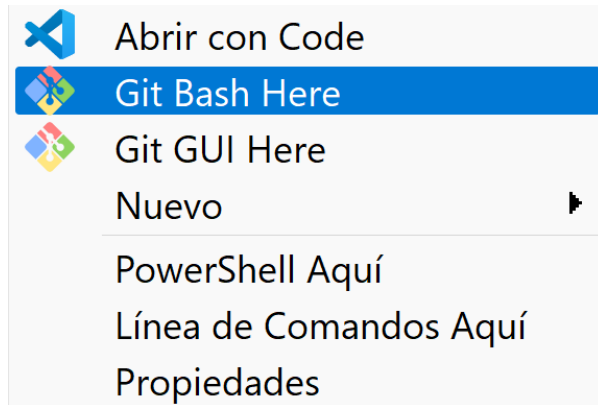
A partir deste ponto, podemos obter uma cópia de segurança de como estavam esses arquivos no momento do backup.

Vejamos um exemplo do que foi dito acima:

Vamos criar um diretório de trabalho para nosso projeto, que chamaremos por exemplo de "Projeto GIT".

Vamos criar ou copiar os arquivos de nosso projeto. Para nosso exemplo, criamos três arquivos, um .html, um .css e um .js

De dentro de seu diretório de trabalho, clique com o botão direito do mouse e, se você instalou GIT corretamente, a opção "Git Bash Here" não deve aparecer no menu.



Execute o init de comando para que o GIT comece a rastrear seu projeto:

```
miPC@miPC MINGW64 /g/WORKSPACE/011
GitHub/Proyecto GIT (master)

$ git init
```

Executamos o comando do tipo `git init` para que a GIT comece a acompanhar nosso projeto..

```
miPC@miPC MINGW64 /g/WORKSPACE/011
GitHub/Proyecto GIT (master)

$ git init

Initialized empty Git repository in
G:/WORKSPACE/011 Github/Proyecto GIT.git
```

Neste momento, a GIT terá criado uma pasta oculta em nosso diretório de trabalho com o nome ".git".

Nombre	Tamaño	Modificación	Ancho	Tipo
.git		Hoy 3:20		Archivo carpeta
estilo.css	0 bytes	Hoy 3:35		Documento de hoja de estilos en cascada
index.html	0 bytes	Hoy 3:35		Documento HTML
main.js	0 bytes	Hoy 3:35		Archivo JavaScript

Esta é a pasta que a GIT utilizará para criar as áreas locais e de preparação do repositório.

Agora precisamos especificar quais arquivos a GIT precisa rastrear:

Para ver em que estado se encontram nossos arquivos, usaremos o comando:

`git status -s`

```
miPC@miPC MINGW64 /g/WORKSPACE/011
GitHub/Proyecto GIT (master)

$ git status -s

?? estilo.css
?? index.html
?? main.js
```

Como podemos ver, ele nos dá uma lista de todos os arquivos que fazem parte de nosso projeto.

Os pontos de interrogação ?? indicam que nenhum de nossos arquivos está sendo rastreado pela GIT.

Para praticar, vamos dizer à GIT para rastrear apenas o arquivo index.html

```
miPC@miPC MINGW64 /g/WORKSPACE/011
GitHub/Proyecto GIT (master)

$ git add index.html
```

Se voltarmos a fazer um `"git status -s"`:

```
miPC@miPC MINGW64 /g/WORKSPACE/011
GitHub/Proyecto GIT (master)

$ git status -s

A index.html
?? estilo.css
?? main.js
```

Você verá agora um A indicando que ele está rastreando o arquivo index.html. Este arquivo está agora na área de encenação, a área de encenação.

Uma vez que nosso arquivo esteja na área de encenação, para tirar um instantâneo de seu conteúdo, usaremos o comando "comprometer" que veremos no próximo tópico.