

IOS – Instituto de
Oportunidade Social

GIT e Github - Comandos no Git



Objetivos da aula



- Regiões Virtuais dos Diretórios
- Primeiros comandos do Git
- Usando o arquivo .gitignore

IOS – Instituto de
Oportunidade Social

Regiões Virtuais dos Diretórios



Antes de revisar as regiões de diretórios é importante reforçar a importância de **digitar os comandos** para facilitar a **memorização e aprendizado**:

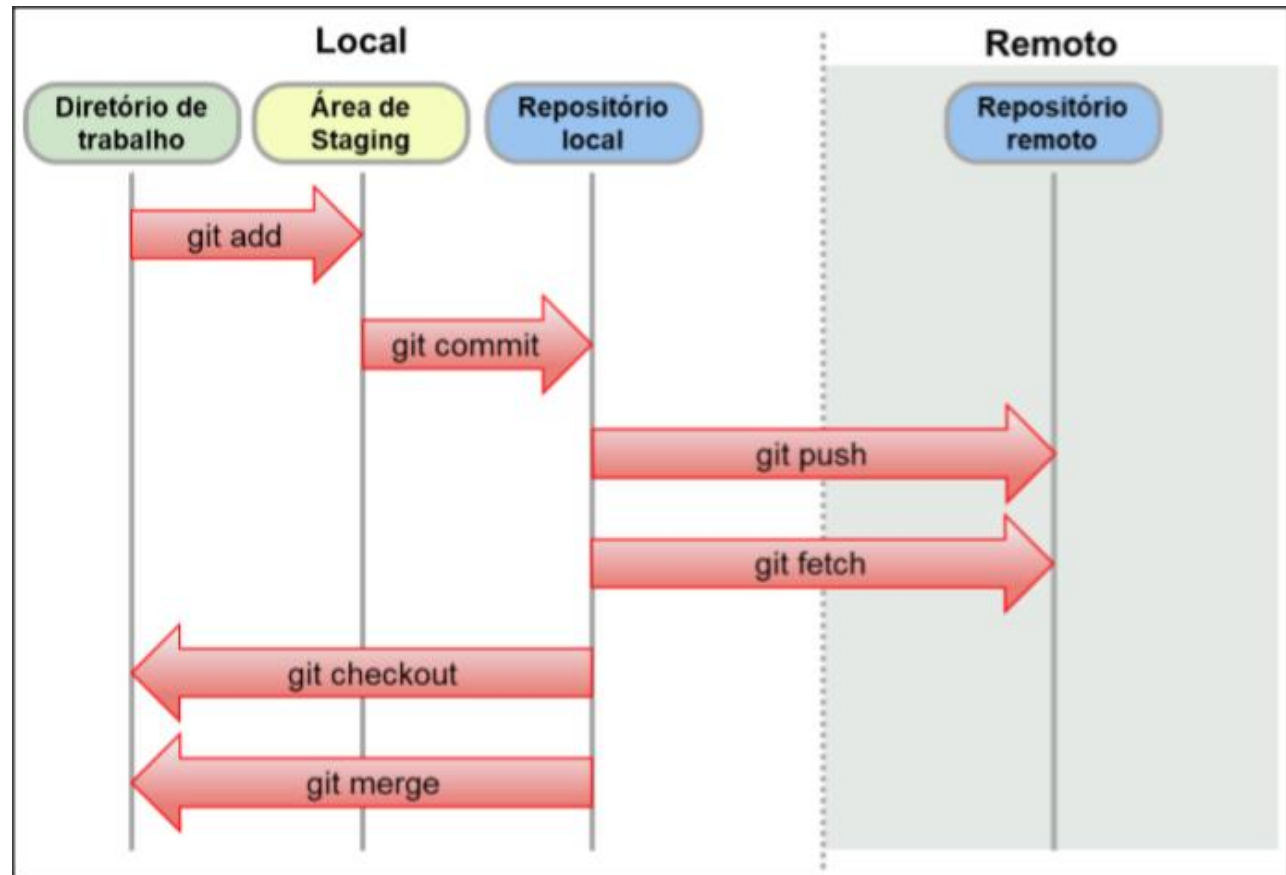


Dica!

Procure digitar todos os comandos mostrados aqui para não dar nenhum erro ao executar um determinado comando. Se você apenas copiar de um arquivo word ou de pdf o comando no terminal de comando, geralmente o terminal interpreta alguns símbolos de forma diferente, mesmo que visualmente esteja muito parecido.

Regiões Virtuais Diretórios

E principalmente, lembre-se o **alcance** de cada **comando**, por exemplo o comando **git add** envia um arquivo do **Diretório de trabalho** para a **Staging area**.



IOS – Instituto de
Oportunidade Social

Primeiros comandos no Git



Primeiros comandos no Git

Para mostrar a **versão do Git instalada** no seu computador:

git --version



Importante!

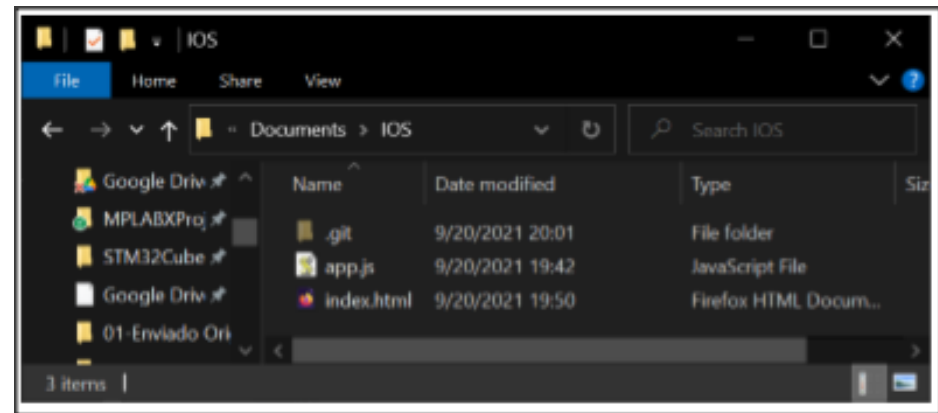
Não copie e cole o código, pois pode acontecer de nessa transição algum caractere seja colado errado, principalmente os dois sinais de subtração (--). Mesmo que esteja muito parecido, digite o código manualmente, para ter certeza de que o comando está correto.

Para criar **novo arquivo** no diretório atual:

New-Item nomeDoArquivo.ext (nome do arquivo e sua extensão)

Para criar **novo Repositório Local**:

git init



Primeiros comandos no Git



Para **vincular** o **Usuário** do GitHub:

git config --global user.name 'Coloque aqui o seu nome'

Para vincular o E-mail do GitHub:

git config --global user.email 'Coloque aqui o seu email'

```
PROBLEMAS    SAÍDA    TERMINAL    CONSOLE DE DEPURACÃO  
..\IOS> git config --global user.name 'Fernando'  
..\IOS> git config --global user.email 'fernando@esquirio.com.br'  
..\IOS> 
```


Primeiros comandos no Git

Para verificar as **configurações** atuais:

git config --list

```
PROBLEMAS  SAÍDA  TERMINAL  CONSÓLE DE DEPURACÃO

Try the new cross-platform PowerShell https://aka.ms/powershell

..\IOS> git config --list
diff.astextplain.textconv=astextplain
filter.lfs.clean=git-lfs clean -- %f
filter.lfs.smudge=git-lfs smudge -- %f
filter.lfs.process=git-lfs filter-process
filter.lfs.required=true
http.sslbackend=openssl
http.sslcainfo=C:/Program Files/Git/mingw64/ssl/certs/ca-bundle.crt
core.autocrlf=true
core.fscache=true
core.symlinks=true
pull.rebase=false
credential.helper=manager-core
credential.https://dev.azure.com.usehttppath=true
init.defaultbranch=master
filter.lfs.clean=git-lfs clean -- %f
filter.lfs.smudge=git-lfs smudge -- %f
filter.lfs.process=git-lfs filter-process
filter.lfs.required=true
user.name=Fernando
user.email=fernando@esquirio.com.br
core.editor="C:\Users\Bigode\AppData\Local\Programs\Microsoft VS Code\Code.exe" --wait
core.repositoryformatversion=0
core.filemode=false
core.bare=false
core.logallrefupdates=true
core.ignorecase=true
..\IOS> |
```

Primeiros comandos no Git

Para **verificar o Status** das alterações:

git status

```
..\IOS> git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
       app.js
       index.html

nothing added to commit but untracked files present (use "git add" to track)
..\IOS> █
```

```
..\IOS> git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
       new file:   index.html

Untracked files:
  (use "git add <file>..." to include in what will be committed)
       app.js

..\IOS> █
```

```
..\IOS> git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
       new file:   app.js
       new file:   index.html

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
       modified:   index.html

..\IOS> █
```

→ staging area

→ Diretório local

Primeiros comandos no Git



Para adicionar arquivo no staging área:

git add NomeDoArquivo.ext (adiciona arquivo específico)

git add *.ext (adiciona todos os arquivos com a extensão)

git add . (adiciona todos os arquivos alterados)

Para remover arquivo do staging área:

git rm --cached NomeDoArquivo.ext

(também podem ser utilizadas variantes de extensão e geral)

Primeiros comandos no Git



Para realizar o **commit** de alterações pendentes no **Staging Área** para o **Repositório Local**:

git commit -m 'Mensagem'

```
..\IOS> git commit -m 'Fazendo o commit inicial'
[master (root-commit) da44491] Fazendo o commit inicial
 2 files changed, 9 insertions(+)
 create mode 100644 app.js
 create mode 100644 index.html
..\IOS>
```

Primeiros comandos no Git

Para visualizar o **histórico de commits** do repositório local:

git log

```
..\IOS> git log
commit f0eb54e7a405b7711b60dca8252af224502e46e5 (HEAD -> master)
Author: Fernando <fernando@esquirio.com.br>
Date: Mon Sep 20 23:19:30 2021 +0200

    Arquivo .gitignore e novo diretorio

commit 736eac207023ceb6cc5ed5ec6fe6cfd519b1421f
Author: Fernando <fernando@esquirio.com.br>
Date: Mon Sep 20 22:16:02 2021 +0200

    Atualização do arquivo app.js

commit da44491d7aca72f6f810c9ccc552ee9ddd50c991
Author: Fernando <fernando@esquirio.com.br>
Date: Mon Sep 20 21:51:15 2021 +0200
: ...skipping...
commit f0eb54e7a405b7711b60dca8252af224502e46e5 (HEAD -> master)
Author: Fernando <fernando@esquirio.com.br>
Date: Mon Sep 20 23:19:30 2021 +0200

    Arquivo .gitignore e novo diretorio

commit 736eac207023ceb6cc5ed5ec6fe6cfd519b1421f
Author: Fernando <fernando@esquirio.com.br>
Date: Mon Sep 20 22:16:02 2021 +0200

    Atualização do arquivo app.js

commit da44491d7aca72f6f810c9ccc552ee9ddd50c991
Author: Fernando <fernando@esquirio.com.br>
Date: Mon Sep 20 21:51:15 2021 +0200
```

Usuário e e-mail
de quem fez o
commit

Fazendo o commit inicial → Mensagem do commit

IOS – Instituto de
Oportunidade Social

Vamos Praticar 1



Vamos Praticar 1



Apostila de Git e GitHub:

- Primeiros comandos no Git

Páginas 30 a 40

OBS: Acompanhar o passo a passo com o instrutor

IOS – Instituto de
Oportunidade Social

Usando o arquivo .gitignore



Usando o arquivo .gitignore



O arquivo **.gitignore** é um arquivo de texto onde cada linha contém um **padrão/regra** para **arquivos** ou **diretórios** que **serão ignorados**, ou seja, que **não serão rastreados** pelo git. Isso é **interessante**, quando você estiver trabalhando com **pacotes de bibliotecas** de terceiros ou com **frameworks** e não quer armazenar no seu git esses arquivos, apenas a sua aplicação.

IOS – Instituto de
Oportunidade Social

Vamos Praticar 2



Vamos Praticar 2



Apostila de Git e GitHub:

- Usar o arquivo .gitignore

Páginas 41 a 46

OBS: Acompanhar o passo a passo com o instrutor

IOS – Instituto de
Oportunidade Social

Exercícios



- Realizar a subida de todos os fontes para o Repositório Local
- Deixar Repositório pronto para Vincular com o Repositório Remoto na próxima aula
- Verificar se todos os passos do Git e GitHub foram executados até o momento