



Swift

O que é Git?

Git é um sistema de controle de versão de arquivos.




O que é GitHub, GitLab, Bitbucket?

São servidores de repositórios

Vamos criar nosso repositório?

github.com



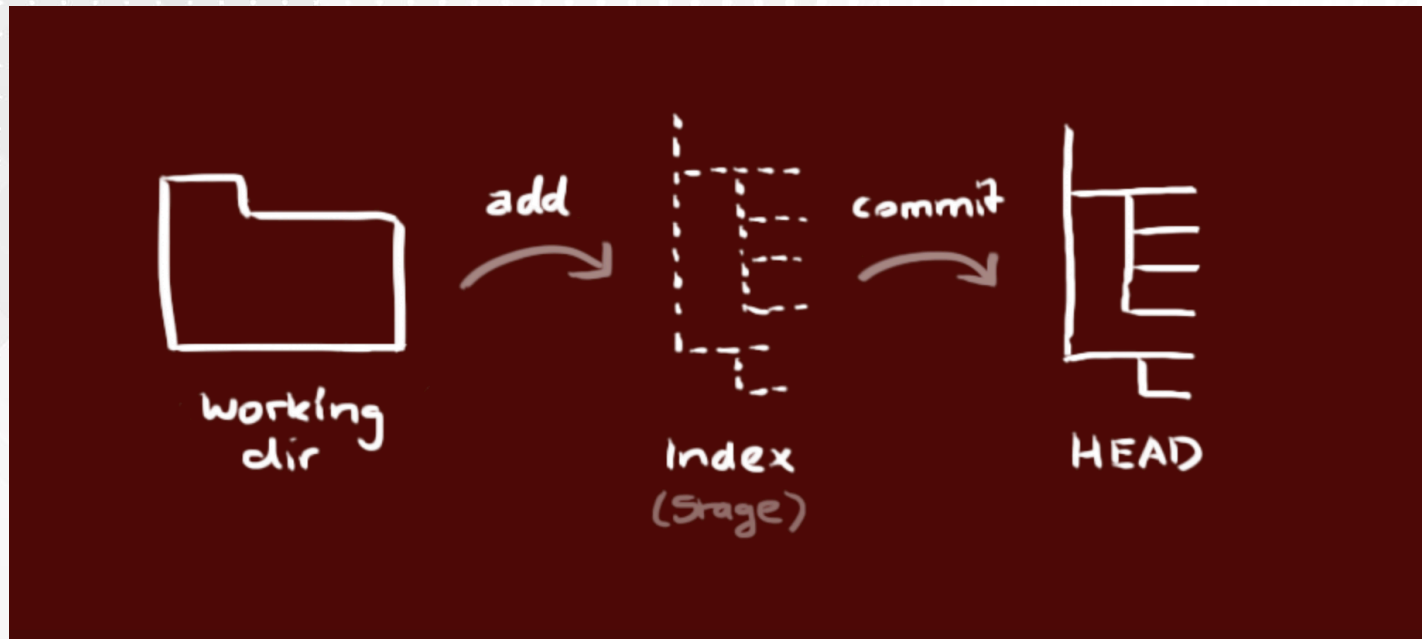
Antes de começar,
vamos adicionar o
.gitignore

Obtenha um repositório

```
git clone
```

```
Exemplo: git clone (url)
```

Fluxo de trabalho





Adicionando

Adicionando mudanças para o índice

```
git add <arquivo>
```

```
git add .
```


Commitando

Enviando o arquivo para o HEAD

```
git commit -m "comentários das alterações"
```



Enviando para o servidor

Para enviar estas alterações ao seu repositório remoto, execute

```
git push origin master
```

Branches

Branches ("ramos") são utilizados para desenvolver funcionalidades isoladas umas das outras. O branch *master* é o branch "padrão" quando você cria um repositório. Use outros branches para desenvolver e mescle-os (*merge*) ao branch master após a conclusão.



Cria um novo branch chamado
"funcionalidade_x" e selecione-o
usando


```
git checkout -b funcionalidade_xc
```






Para retorne para o master usamos

```
git checkout master
```




E para remover o branch usaremos o
seguinte comando

```
git branch -d funcionalidade_x
```



Seu branch ainda não foi
disponibilizado no servidor, para isso,
use o seguinte comando

```
git push origin funcionalidade_x
```

Para atualizar seu repositório local
com a mais nova versão, execute

```
git pull
```

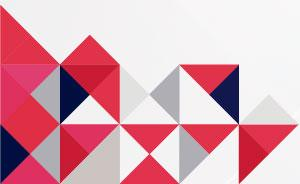



Para fazer merge de um outro branch ao seu branch ativo

```
git merge <branch>
```

Exemplo master

```
git merge master
```



Outros comandos

```
git status
```

```
git branch
```

```
git diff
```

Links

[Tudo que você precisa saber sobre Git](#)

[Git para preguiçosos](#)

[Git scm](#)