

Branches

ENTENDENDO O QUE É UMA BRANCH

Acessar o repositório de Maria via Git Bash

Verificar que o arquivo master é uma referência para o último commit feito nesta branch

```
$ git hist  
$ cat .git/refs/heads/master
```



HEAD

ENTENDENDO A REFERÊNCIA HEAD

Acessar o repositório de Maria via Git Bash

Verificar que o arquivo HEAD é uma referência para a branch atual master

```
$ cat .git/head  
$ git hist
```



Criando uma nova branch

ENTENDENDO OS COMANDOS BRANCH/SWITCH

Acessar o repositório de Maria via Git Bash

Verificar que neste momento a branch atual é a master

```
$ git hist
```

Nota: novamente, não esqueça, HEAD é uma referência para a branch atual. E a branch atual é uma referência para o último commit realizado nesta branch

Criar uma nova branch para funcionalidade de cadastro

```
$ git branch registerr
```

Nota: "registerr" está escrito errado de propósito. Será corrigido em breve

Verificar que foi criado uma branch registerr a partir da branch master

```
$ git hist
```

Nota: a branch master ainda permanece como branch corrente

Alternar para a branch registerr

```
$ git switch registerr
```

Verificar que a branch registerr se tornou a branch corrente

```
$ git hist
```

Criar um arquivo register.js

```
$ echo "function register(){...}" >> register.js
```



SEÇÃO 3: REFERÊNCIAS NO GIT

Adicionar register.js para a área de preparo

```
$ git add register.js
```

Realizar o commit

```
$ git commit -m "add register"
```

Verificar que a branch registerr está a um commit a frente da branch master

```
$ git hist  
$ git hist master
```



Branches de rastreamento

ENTENDENDO BRANCHES DE RASTREAMENTO

Listar as branches com todos os tipos

```
$ git branch -a
```

Nota: em vermelho é exibido as branches de rastreamento, em branco as branches locais e o * em verde indica a branch corrente

Nota: a branch `registerr` foi criada mas ainda não existe uma de rastreamento correspondente

Enviar ao repositório remoto

```
$ git push
```

Nota: ocorreu um erro ao realizar o push novamente. Isso ocorreu porque a nova branch `registerr` não possui uma branch de rastreamento correspondente

Para o primeiro envio nesta branch seguir a recomendação do Git

```
$ git push --set-upstream origin registerr
```

Nota: a opção `--set-upstream` é equivalente a `-u`

Verificar que agora existe uma branch de rastreamento para a branch local `registerr`

```
$ git branch -a
```

Verificar que a branch local `"registerr"` rastreia `"origin/registerr"`

```
$ git branch -vv
```

Visualizar a branch criada no Bitbucket



BRANCHES DE RASTREAMENTO - GIT INTERNALS

Visualizar que a branch de rastreamento, neste momento, também é uma referência para o commit atual

```
$ cat .git/refs/remotes/origin/registerr  
$ git hist
```



Renomear uma branch

RENOMEAR BRANCH LOCAL

Acessar o repositório de Maria via Git Bash

Renomear branch local "registerr" para "register"

```
$ git branch -m register
```

Nota: caso a branch a qual o nome que será alterado não seja a branch atual, poderia ser executado o seguinte comando: `git branch -m "<NOME_ANTIGO>" "<NOME_NOVO>"`

RENOMEAR BRANCH NO REPOSITÓRIO REMOTO

Após ajustar o nome da branch localmente, é necessário remover a branch com o nome antigo do Bitbucket. Para isso há 2 opções:

1. Remover manualmente através do Bitbucket
2. No terminal enviar uma ordem de remoção através do comando a seguir:
 - `git push origin --delete registerr` OU
 - `git push origin :registerr`

Enviar ao repositório remoto a branch com o nome correto

```
$ git push -u origin register
```



Remover uma branch

REMOVER BRANCH LOCAL

Acessar o repositório de Maria via Git Bash

Tentar remover a branch register com o seguinte comando

```
$ git branch -d register
```

Nota: o Git informa que não é possível remover a branch corrente

Alternar para a branch master

```
$ git switch master
```

Remover a branch register

```
$ git branch -d register
```

Verificar que a branch local "register" foi removida

```
$ git branch -a
```

Nota: a branch de rastreamento "remotes/origin/register" permanece

REMOVER BRANCH NO REPOSITÓRIO REMOTO

Remover a branch do repositório remoto

```
$ git push origin :register
```

Verificar que a branch de rastreamento "remotes/origin/register" foi removida

```
$ git branch -a
```



Tags anotadas

ENTENDENDO UMA TAG ANOTADA

Acessar o repositório de Maria via Git Bash

Verificar o histórico

```
$ git hist
```

Criar uma tag anotada

```
$ git tag -a "v1.0" -m "first version"
```

- -a: indica que é uma tag anotada
- -m: indica que o próximo parâmetro será a descrição da tag
- Por padrão a tag será adicionado no commit corrente. Mas você pode adicionar tags a qualquer outro commit do passado, basta adicionar a hash do commit

Listar as tags

```
$ git tag
```

Visualizar as informações de uma tag anotada

```
$ git show v1.0
```

TAG ANOTADA – GIT INTERNALS

Verificar que o arquivo da tag não é uma referência para um objeto de commit

```
$ cat .git/refs/tags/v1.0
```

Copiar a hash apresentada



Verificar que o objeto trata-se de uma tag

```
$ git cat-file -t <HASH_TAG>
```

Verificar que a tag está relacionada ao objeto do commit atual

```
$ git cat-file -p <HASH_TAG>  
$ git hist
```



Tags leves

ENTENDENDO UMA TAG LEVE

Acessar o repositório de Maria via Git Bash

Criar uma tag leve

```
$ git tag v1.0-lw
```

Verificar no histórico que a tag foi criada

```
$ git hist
```

Visualizar as informações de uma tag leve e comparar as informações disponíveis de uma tag anotada que não existem em uma tag leve

```
$ git show v1.0-lw  
$ git show v1.0
```

TAG LEVE – GIT INTERNALS

Verificar que o arquivo da tag é uma referência para o objeto do último commit feito nesta branch

```
$ git hist.  
$ cat .git/refs/tags/v1.0-lw
```

REMOVER UMA TAG

Remover a tag temporária

```
$ git tag -d v1.0-lw
```



Enviando tags para o repositório remoto

ENTENDENDO O ENVIO DE TAGS PARA O REPOSITÓRIO REMOTO

Acessar o repositório de Maria via Git Bash

Enviar os objetos para o repositório remoto

```
$ git push
```

Nota: push sem argumentos não envia as tags para o repositório remoto

Para enviar as tags junto com os commits que ainda não existem no repositório remoto utilize a opção "--tags"

```
$ git push --tags
```

Acessar o Bitbucket e validar criação da tag anotada



Está gostando deste curso?

*Compartilhe sua experiência nas redes sociais com a tag
#rsantanatech para que eu possa interagir com a sua postagem.*

**Acompanhe nas redes sociais
e fique por dentro de todos os conteúdos.**

