

Merge: fast-forward

ENTENDENDO MERGE COM A ESTRATÉGIA FAST-FORWARD

Acessar o repositório de Maria via Git Bash

Comparar o histórico da branch atual(login) e a master

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

Nota: a branch login está commits à frente em relação a master

Retornar a branch master

```
$ git switch master
```

Realizar o merge com a branch login

```
$ git merge login
```

Nota: nesta situação a estratégia de merge utilizada é fast-forward

Verificar o histórico da branch master (branch atual)

```
$ git hist
```

Nota: após o merge o histórico permaneceu linear

Enviar esse histórico da branch master para o repositório remoto

```
$ git push
```

Verificar a existência da função validate do arquivo login.js no Bitbucket



Merge: three-way

ENTENDENDO MERGE COM A ESTRATÉGIA THREE-WAY

Acessar o repositório de Maria via Git Bash

Criar uma nova branch e alternar para a mesma

```
$ git switch -c users
```

Gerar um novo commit nesta nova branch que adiciona um novo arquivo users.js

```
$ echo "function save(){...}" >> users.js  
$ git add users.js  
$ git commit -m "add save users"
```

Alternar novamente para a branch master

```
$ git switch master
```

Gerar um novo commit nesta branch

```
$ echo "function setCookie(){...}" >> login.js  
$ git add login.js  
$ git commit -m "add setCookie on login"
```

Comparar o histórico da branch atual(master) e users

```
$ git hist  
$ git hist users
```

Nota: o último commit em comum entre as duas branches é o penúltimo commit em ambas as branches



SEÇÃO 5: BIFURCAÇÕES NO HISTÓRICO

Realizar o merge com a branch users

```
$ git merge users
```

Nota: nesta situação a estratégia de merge utilizada é ort

Verificar o histórico da branch master (branch atual)

```
$ git hist
```

Nota: após o merge o histórico não é mais linear, possuindo uma bifurcação

Enviar esse histórico da branch master para o repositório remoto

```
$ git push
```

Verificar o novo arquivo users.js e a nova função setCookie do arquivo login.js no Bitbucket



DAG

NAVEGAÇÃO EM UM HISTÓRICO BIFURCADO

Acessar o repositório de Maria via Git Bash

Abrir dois terminais

No terminal do lado esquerdo, executar:

```
$ git hist
```

Nota: último commit possui dois pais, por ser um commit de merge

No terminal do lado direito, executar:

```
$ git show head~
```

Nota: A opção ~ considera o commit anterior, como o commit atual possui dois pais, foi considerado o pai da esquerda no histórico

```
$ git show head^
```

Nota: a opção ^ considera o primeiro pai (sempre da esquerda pra direita)

```
$ git show head^2
```

Nota: a opção ^2 considera o segundo pai (sempre da esquerda pra direita)



Reset de merge

ENTENDENDO COMO DESFAZER UM MERGE

Acessar o repositório de Maria via Git Bash

Verificar o histórico da branch atual

```
$ git hist
```

Nota: o último commit é um commit de merge

Copiar a hash do commit de merge

Remover o commit de merge com git reset

```
$ git reset --hard head~
```

Verificar o histórico da branch atual

```
$ git hist
```

Retornar com o commit de merge com git reset

```
$ git reset --hard <hash do commit de merge>
```

Verificar o histórico da branch atual

```
$ git hist
```



No fast-forward

ENTENDENDO MERGE COM A OPÇÃO NO FAST-FORWARD

Acessar o repositório de Maria

Criar uma nova branch e alternar para a mesma

```
$ git switch -c groups
```

Gerar um novo commit nesta nova branch

```
$ echo "function save(){...}" >> groups.js  
$ git add groups.js  
$ git commit -m "add save groups"
```

Verificar o histórico da branch groups

```
$ git hist
```

Alternar novamente para a branch master

```
$ git switch master
```

Realizar o merge com a branch groups

```
$ git merge groups
```

Nota: a estratégia de merge foi fast-forward como esperado

Verificar o histórico da branch atual

```
$ git hist
```

Nota: ambas as branches são referências para o commit "add groups"



SEÇÃO 5: BIFURCAÇÕES NO HISTÓRICO

Retornar para o commit anterior, que nesse caso era o commit referenciado antes do merge

```
$ git reset --hard head~
```

Realizar o merge com a branch groups utilizando a opção --no-ff

```
$ git merge --no-ff groups
```

Nota: com a opção --no-ff não é utilizado a estratégia fast forward, forçando a estratégia ort

Verificar o histórico da branch atual

```
$ git hist
```

Nota: foi gerado um commit de merge para mesclar com a branch groups

Enviar esse histórico da branch master para o repositório remoto

```
$ git push
```

Verificar o novo arquivo groups.js no Bitbucket



Conflitos

RESOLVENDO CONFLITOS DE MESCLAGEM DURANTE O PROCESSO DE MERGE

Acessar o repositório de Maria via Git Bash

Criar uma branch login-webservice e alternar para a mesma

```
$ git switch -c login-webservice
```

Verificar o conteúdo do arquivo login.js

```
$ cat login.js
```

Alterar o arquivo login.js adicionando uma quarta linha e realizar o commit desta alteração

```
$ echo "function validateOnWebservice(){...}" >> login.js  
$ git add login.js  
$ git commit -m "add validate webservice on login"
```

Retornar para a branch master

```
$ git switch master
```

Verificar o conteúdo do arquivo login.js

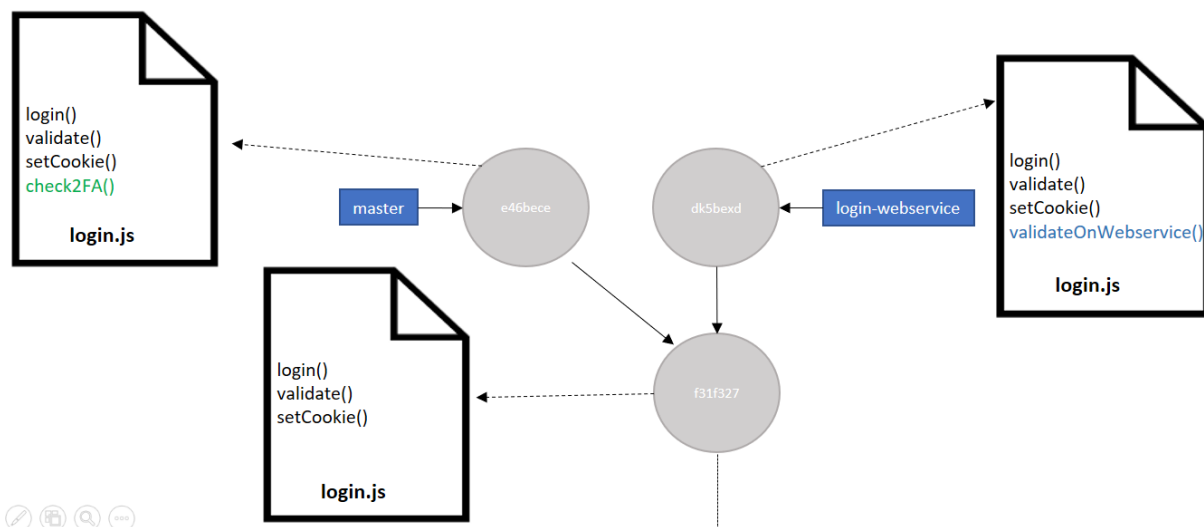
```
$ cat login.js
```



SEÇÃO 5: BIFURCAÇÕES NO HISTÓRICO

Alterar o arquivo login.js adicionando uma quarta linha e realizar o commit desta alteração

```
$ echo "function check2FA(){...}" >> login.js  
$ git add login.js  
$ git commit -m "add check if 2fa is enable"
```



Realizar o merge com a branch login-webservice

```
$ git merge login-webservice
```

Nota: o merge automático não foi possível pois houve conflito

Resolva o conflito com o editor de preferência

Adicionar a resolução do conflito na área de preparo e fazer o commit

```
$ git add login.js  
$ git commit
```



SEÇÃO 5: BIFURCAÇÕES NO HISTÓRICO

Avaliar o histórico após a resolução de conflito

```
$ git hist
```

Enviar esse histórico da branch master para o repositório remoto

```
$ git push
```

Verificar as novas funções no arquivo login.js no Bitbucket



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.**

