



Capitulo 6: Redirecionadores, Pipes, AND e OR

☒ Reviewed



REDIRECIONADORES

```
> #Redireciona a saída padrão - stdout (standard output)

<

>>

<<
```

>

- Manda o comando para um determinado lugar
- Se colocar outro comando além do colocado, ele vai **sobrepor**

```
ls > saida_ls.txt
uptime > saida_uptime.txt
```

>>

- Redireciona a saída padrão e concatena no arquivo/disp indicado
- **Ele adiciona na linha abaixo**

```
ls >> saida_ls.txt
```

<

- Joga para o **comando interpretar** - Stdin

```
cat < opa.txt
```

```
#saida: GIROPOPS STRIGUS
```

<<

- Ele **para o modo de edição** quando você coloca uma palavra chave

```
cat << giropops
> eu fui na feira
> ontem de manha
> eh mentira
> pois na holanda nao tem feira
> giropops
```

REDIRECIONAMENTO DE ERROS

- stdin = 0
- stdout = 1
- stderr = 2

```
2> #Redireciona a saida de erro padrão substitui - stderr
```

```
2>> #Redireciona a saida de erro padrão concatenando - stderr
```

```
lsblablablabla 2> erro.txt
```

```
#saida: lsblablabla: command not found
```

&

- tanto a saída de erro quanto a saída padrão serão direcionadas para o mesmo destino.

```
ls foguete opa.txt > saida_marota.txt 2>&1
#saida: ls: cannot access 'foguete': No such file or director
#      opa.txt
```

PIPES (|)

- Faz várias coisas ao mesmo tempo
- **Pega o conteúdo do comando anterior e joga para o outro comando processar/executar**

```
exemplo 1: ls | tee arquivo_novo.txt
```

```
exemplo 2: cat /etc/passwd | grep root | nl
```

OPERADORES (AND && e OR ||)

- **AND (&&):** Se o primeiro funcionar, execute o segundo comando — somente se for com sucesso. Se o segundo comando não funcionar, o terceiro ele não funciona
- **OR (||):** Quando ele encontra um arquivo válido, ele descarta os outros

AND:

```
ls opa.txt && mv opa.txt novo_opa.txt
```

OR:

```
ls novo_opa.txt || ls opa.txt || ls arquivo_novo.txt
```

```
ls kskfdks || ls kdeodk || ls novo_opa.txt
```