

# Curso de Linux - Primeiros Passos

Autor:

Canal Curso em Vídeo  
(Gustavo Guanabara,  
Ricardo Pinheiro)

Link:

[https://www.youtube.com/playlist?list=PLHz\\_AreHm4dIIxleu20uwPWFOSSwqLYbV](https://www.youtube.com/playlist?list=PLHz_AreHm4dIIxleu20uwPWFOSSwqLYbV)

## NOTA SOBRE OS COMANDOS:

Os vídeo foram realizados em ambiente Linux, na distro Mint.

Todos os comandos aqui apresentados deveriam funcionar em todas as distribuições derivadas do Debian.

## NOTA SOBRE O GUIA:

Este guia está dividido em duas partes, a primeira apresenta os comandos e uma descrição básica de cada um.

A segunda parte é basicamente um resumo do que foi falado nos vídeos, algumas sessões abordam de forma mais detalhada algum assunto.

*Alguns comandos funcionam de várias formas, porém, eles decidiram por apresentar somente algumas formas.*

**Recomenda-se verificar as fontes e veracidade das informações deste guia antes de considerar seriamente um estudo sobre esse documento.**

NOTA:

DIRETÓRIO e PASTA significam a mesma coisa neste guia.

## Listar Arquivos/Diretórios

\$ls		/*	Ambos mostram os arquivos da que estão dentro da pasta atual	*/
\$ls	.			
\$ls	-l	/*	Executa a visualização detalhada de todos os arquivos dentro da pasta atual	*/
\$ls	-a	/*	Mostra arquivos, pastas e elementos ocultos dentro do diretório	*/
\$ls	-all			
\$ls	-h	/*	Converte a visualização de bits para ordens de grandeza da computação (Kb,Mb,Gb etc)	*/
\$ls	-R	/*	É a listagem recursiva, mostra todas as pastas contidas	*/
\$ls	/localizacao/*nome	/*	Função listar tudo o que se encaixa com a palavra "nome", dentro da pasta	*
\$ls	*parte_da_palavra*	/*	Executa uma busca atrás daquela parte da palavra, na pasta atual	*/
O Parâmetro	*		é usado para substituir uma identificação.	

```

=====
$ls      /localizacao/[a-i]                /*      Parecido com ' * '. Esse
                                                abrange de um caractere
                                                até outro                                */

$ls      /localizacao/[a-c,e]              /*      Desde a letra ' a ' até ' c '
                                                junto com o caractere ' e '          */

$ls      /localizacao/f[a-c,e]*            /*      Inclui agora o ' f ' no começo
                                                da palavra e o asterisco inclui
                                                qualquer coisa no final dessa
                                                mesma palavra                        */

=====
$ls      /localizacao/{string}              /*      Executa busca, por meio
                                                de string (sentença)                */

$ls      /localizacao/{string1,string2}
$ls      /localizacao/*{string1,string2}*   /*      Pode usar referência global    */
=====

```

+++++

É possível combinar parâmetros com outros comandos:

```

$ls      -lh                                /*      Saída listada formatada
$ls      -l -h                              para a leitura de
                                                humanos                                */

$ls      ls      -lah                       /*      Vai usar o parâmetro “ -l “ que
                                                mostra a listagem detalhada.
                                                ”-a” Mostra mostra TODOS os
                                                arquivos dentro da pasta
                                                ”-h” Converte os bits de
                                                saída para mb/gb/kb etc            */

```

+++++

Parâmetro para a pasta atual        '        '        (<----ponto, só o ponto)

```

$ls      .                                  /*      Faz a função de listar os
                                                arquivos da pasta atual            */

```

+++++

## Acessar Diretórios

\$cd	/nome/nome	/*	Acessa uma pasta (caminho relativo)	*/
\$cd	.	/*	Acessa o diretório atual	*/
\$cd	..	/*	Retorna diretório anterior	*/
\$cd	~	/*	Acessa o diretório do usuário	*/
\$cd	/	/*	Acessa a pasta raiz	*/

## Criar Diretórios

\$mkdir	NomePasta	/*	Criação de uma pasta	*/
\$mkdir	Nome1 Nome2 Nome3	/*	Criação de várias pastas	*/
\$mkdir	PastaPai/PastaFilho/PastaNeto	/*	Cria uma dentro da outra	*/
\$mkdir -p	PastaPai/Pasta\Filho/Pasta\Neto	/*	-p Para nome separado	*/
\$rmdir	NomeDaPasta	/*	Apaga diretório	*/

## Manipular Arquivos

\$rm	NomeArquivo	/*	Remove um arquivo	*/	
\$rm	-rf	/*	Remove recursivamente, " f " é para forçar	*/	
\$rm	-i	/*	Pergunta se deseja apagar, de um em um	*/	
\$mv	nome_arquivo	/pasta_destino	/*	Move o arquivo	*/
\$mv	nome_arquivo	novo_nome_do_arquivo	/*	Cria um novo nome para a pasta	*/
\$cp	nome_do_arquivo.txt	/pasta_destino	/*	Realiza a copia arquivo para a pasta desejada	*/
\$cp	/etc/*.conf	localização_da_pasta	/*	Usa o parâmetro global asterísco	*/

## Apelido Para Comandos

\$alias	/*	Coloca um apelido temporário no comando que permanece enquanto o terminal não for fechado	*/
\$alias nome_do_comando='apelido'	/*	Cria um apelido para o para o comando	*/
\$alias rm='beijola'	/*	beijola é o comando remover	*/
\$alias rm='/bin/rm -i'	/*	Mais de 3 arquivos para apagar gera uma mensagem de confirmação antes de executar o comando de remoção	*/

## Outros Comandos

\$nano nome.txt	/*	Usa o editor de texto nano, padrão no Linux	*/
\$sudo su	/*	Faz login como root	*/
\$sudo su nome_do_usuario	/*	Faz o login como modo usuário	*/
\$nautilus	*/	Interface gráfica. Abre o dir do usuário	*/
\$nautilus /pasta/SubPasta	/*	Acessa pelo caminho	*/

\$cat	/*	Mostra as informações do arquivo na tela	*/
\$cat /Desktop/nome_do_.txt	/*	Também o elemento acessa pelo caminho	*/
\$tac	/*	Faz a mesma coisa que o cat, mas, de trás pra frente.	*/
\$less nome_do_arquivo	/*	Faz a paginação do arquivo	*/
\$lscpu	/*	Mostra informações do processador	*/
\$pwd	/*	Mostra o caminho até o arquivo atual, desde o dir raíz (caminho absoluto)	*/
\$touch Nome.txt	/*	Cria um arquivo de texto comum	*/
\$man nome_do_comando	/*	Dá informações sobre o comando É um help mais avançado	*/
\$history	/*	Mostra todos os comandos usado até o momento, mas tem limite.	*/
\$!numero_comando	/*	Ele acessa o comando, referenciado por um número.	*/



## Segunda Parte - Trechos retirados dos vídeos

### INFORMAÇÃO GERAL:

- > O Linux tem vários interpretadores de comando, um dos mais usados é o BASH, que é o terminal padrão.
- > NEM TODOS os comandos são escritos em minúsculo.
- > Quase tudo em UNIX é arquivo.

nomedousuario@nomedousuario:~\$	//Isso se chama Prompt

\*Demonstração da interface do bash (Figura acima)

Um parâmetro modifica a saída do comando (existem vários..só o comando ls...ultrapassa 30).  
Um parâmetro é passado com um tracinho e uma letra, é o padrão da norma POSIX.  
Ou pode usar DOIS tracinhos e UMA palavra, por exemplo “ --help “.

Arquivos começam com um ponto “ . “ viram arquivo ocultos.  
+ A vantagem disso é que não vai aparecer na listagem  
+ É muito usado para guardar configuração  
Para visualizar arquivos ocultos, use o comando \$ls -a

-----

Veja o comando \$ls -a sendo usado abaixo  
Diretórios ponto-ponto e ponto único

drwxr-xr-x	4	usuario usuario	4,0K	set	4 21:31	.	//ponto
drwxr-xr-x	29	usuario usuario	4,0K	ago	31 20:28	..	//ponto-ponto
drwxrwxr-x	5	usuario usuario	4,0K	ago	27 21:47		Estudos
-rw-rw-r--	1	usuario usuario	173K	set	3 22:20		Linux_Guanabara
-rw-rw-r--	1	usuario usuario	362K	set	3 22:17		Linux_Guanabara.xcf
drwxrwxr-x	2	usuario usuario	4,0K	set	4 21:31		.ponto

Nessas duas primeiras linhas abaixo, a pasta ponto único aponta pra pasta que vc está.  
E a pasta ponto-ponto aponta para a pasta antecessor à ela.

-----

Os comandos em linux são executáveis que estão salvos na pasta bin (ex: ls, mkdir etc)  
Alguns como cd não aparece como arquivo.

-----

INFO:

\$ls	--all	//Esse é parâmetro, tem menos-menos na frente
\$ls	-a	//Também é parâmetro
\$cd	..	//ESSE é um ARGUMENTO

-----

Na pasta raiz “ / “ existe o diretório dev  
Lá estão todos os dispositivos da máquina:

Quando abrir essa região, verás o elemento “sda”, que é o HD da máquina (sda1, sds2 etc são as partições dele)

Os dispositivos neste diretório estão coloridos em amarelo.

As pastas estão em azul

Se colocar um pendrive na máquina, ele poderá ser acessado em /dev/usb, ele será criado lá dentro, não se esqueça de ejetar o pendrive antes de retirá-lo da máquina.

-----

-----

Os programas estarão na pasta        /usr

/proc é um sistema de arquivos que tem vários arquivos com informações sobre o sistema

A pasta boot é onde fica o Kernel, vmlinuz é um link para o kernel

-----

O Linux tem alguns editores de texto, como o VI (víáí), e o nano

-----

LINK ---->Site do Aurelio (apelidado de verde, por causa do cabelo que foi pintado de verde),  
tutorial para o shell do linux ---> <https://aurelio.net/shell/>

-----

>A distro Mint é muito focada nos computadores comuns

>A distro DEBIAN tem seu uso mais genérico

>Os outros interpretadores de comando possuem características diferentes, um deles podem  
apresentar o comando inserido errado.

>A maioria das distro linux usam os mesmo comandos.

>O Linux salva até 256 caracteres por nome, e ele aceita tudo em teoria, e separa nomes com  
letras maiúsculas e minúsculas

>Caminho absoluto é o pwd que mostra onde você encontra desde a pasta raíz, caminho  
relativo é o caminho que você pode escolher fazer à partir da sua pasta atual..o cd por ex;

>O Linux tem lixeira, mas quando o elemento é apagado por comandos, eles não vão pra  
lixeira

-----

Terminal Linux - Referência Global - Curso Linux #08

Nem tudo pelo terminal é mais rápido de usar, em comparação com a interface gráfica.

Tanto o Ubuntu quanto o Mint tornaram o linux mais fácil de ser usado pelo usuário comum.

Em 2004 a galera do Ubuntu distribuía o sistema de graça em CD.

Nem a interface gráfica e nem o terminal é de fato mais amigável à todos, essa denominação  
cabe ao quê o usuário está mais acostumado a usar.

Para um estudo mais aprofundado -> [estudonauta.com](http://estudonauta.com) (segundo Gustavo, tem mais de 60  
aulas, desde o básico até criação de um servidor “diferenciado”, tudo feito no Debian).

-----

-----

Referência Global: Ricardo “o que a gente chama “caracteres de referência global são recursos para especificar um ou mais arquivos/diretórios de uma só vez.(Também existe no Ambiente Windows, com o prompt de comando ou powershell)”.

Ex: Ricardo “vamos dizer que você quer especificar vários arquivos que tem alguma coisa em comum. Tipo, como o nome em comum/ formato comum (.pdf, .docx, .iso etc.).”

-----

Ricardo “No ambiente Linux, o sistema NÃO exige a presença de extensões de arquivos, sendo usado apenas por conveniência”.

“Um exemplo do uso da referência global: copiar e colar vários arquivos de uma pasta pra outra”.

“Referência global não é um comando em si, ele é usado JUNTO com um comando”.

-----

Ricardo “O /etc/ guarda configurações do computador como um todo”.

-----

Ricardo “tudo em Linux é arquivo (ele se refere à diretório e arquivo como sendo ambos a mesma coisa.)”

“Quando você colocar um pendrive no computador, ele se torna um dispositivo montado na máquina, assim como o teclado, o sistema cria um arquivo relacionado ao dispositivo. Lá na pasta de device, o pendrive vai tá lá montado certinho, e vai ter um arquivo relacionado ao pendrive.”

-----

Ricardo: “Uma coisa que a gente precisa definir bem é que, qualquer sistema tem uma entrada padrão e uma saída padrão, entrada padrão: teclado/saída padrão: tela”.

“Isso também vale pro DOS, ele tem duas saídas e ambas apontam pro mesmo lugar, pra tela: Saída do comando é a saída padrão e também a saída de erro”.

-----

Ricardo: “vamos usar agora comandos para manipulação de arquivo. Para isso vamos usar a o arquivo /tmp. O tmp é o lugar onde ficam guardados os arquivos temporários. Então, como são temporários...se você não acessar eles com uma certa frequência, uma hora eles somem”.

-----

-----

Ricardo: “Uma coisa que é importante falar, é sobre a questão de pagar arquivos”  
É difícil recuperar arquivos apagados, “boa sorte se for usar ferramenta” pra isso.  
Guanabara: “Nenhuma ferramenta de recuperação de arquivos funciona 100%”.

Ricardo: “quando você apaga o arquivo, ele desaloca os setores, você até pode conseguir recuperar o conteúdo mas perde o nome original do arquivo, depende da ferramenta também”.  
NOTA: Dá pra recuperar arquivos depois de deletados, até talvez com uso de programas? A resposta é talvez.

-----

Usando o comando cd, é possível acessar qualquer caminho, independente da localização você está.

Uso de pendrives e HDs externos no Linux - Curso Linux #10

Guanabara: “O Mac não permite gravação do pendrive formatado em NTFS (até 2019, o momento que foi gravado o vídeo, essa regra persiste)”.

Ricardo Pinheiro: “Isso também acontecia antigamente no Linux, por causa da patente da microsoft sobre o NTFS”.

“””Os desenvolvedores do Linux fizeram engenharia reversa para conseguir a compatibilidade”””. -Carece de fonte...como todas as outras declarações dele e do Guanabara.

-----

Ricardo: “Não vale a pena formatar um pendrive para o ext4 (formato jornalizado do Linux) porque você fica limitado ao sistema”

“Um sistema jornalizado grava as informações temporariamente para que você possa recuperá-las se ocorrer algum problema, mas normalmente não usam esse sistema nos pendrives”

“Os vírus de pendrive são comuns nos outros formatos como o fat32, um vírus conhecido são aqueles arquivos autorun.inf”

“USB, o S de USB significa serial. Isso também significa que a transmissão é assíncrona, então quando pedimos para ejetar o dispositivo pendrive, ele vai parar o seu recebimento de arquivos, vai sincronizar os dados e vai salvar tudo, só assim que o sistema operacional permitir que retire, será então o momento de remover o pendrive, qualquer S.O é assim”.

Guanabara: “A microsoft planeja implementar a opção que permite a remoção do pendrive sem a necessidade de requisição da ejeção do dispositivo, mas em teoria o sistema deverá ficar mais lento”.

Ricardo: “Um dos pinos do pendrive passa corrente elétrica”

-----

Guanabara: “Apaguei um arquivo do pendrive..mas não apareceu na lixeira..por quê?”.

Ricardo: “Tanto o Windows/Mac/Linux trabalha com uma pasta LIXEIRA escondida dentro do dispositivo. No Windows esse arquivo é o ‘recycle...alguma coisa’ ”

Guanabara: “Ok, eu tenho um arquivo aqui e vou mover DIRETAMENTE para a lixeira do Linux...ok...tá ok, fiz isso, mas porque o tamanho do armazenamento não aumentou..na real, ta até parecendo que aumentou o consumo de espaço...por quê isso aconteceu?”.

Ricardo: “É porque ele agora salvou as informações necessárias para você recuperar o arquivo dentro da lixeira do S.O, caso você requisite isso. Essa é uma conveniência”.

Gustavo: “Ok..mas a pasta não foi transferida para minha lixeira dentro meu computador como eu pedi..ou foi?”

Ricardo: “Ela aparece lá, mas não está lá..e no pendrive agora tem um arquivo oculto. ”

**\*\*tira o pendrive e colocar de volta**

Guanabara: “Ó, tiramos o pendrive e colocamos de volta, a lixeira ainda do S.O ainda está com o arquivo do pendrive que nós movemos pra cá, aqui”

“A lixeira não mostra só o lixo que foi apagado no S.O, mas também do pendrive”

Ricardo: “em todos os dispositivos tem uma lixeira, quando você esvazia a principal do S.Om ela apaga todos os arquivos que já foram exigidos para serem apagados a partir dos outros dispositivos. Quando você pede pra esvaziar a principal, todas as outras lixeiras também esvaziam”.

Guanabara: “Segurando a tecla shift e del, você apaga definitivamente o arquivo, sem mandar pra lixeira. APAGA DE VEZ. APAGA DIRETAMENTE”. ISSO SERVE PRA TODOS OS SISTEMAS OPERACIONAIS

NOTA GUANABARA:

“Vai sair o novo explorer (EDGE) vai sair pra Linux”

## COMANDOS:

```
=====
|-----|
| $ls      -l    //Listagem detalhada de todos os arquivos dentro da pasta atual |
|-----|
| total 32 |
| drwxr-xr-x 2 cursoemvideo cursoemvideo 4096 fev 14 09:35 Desktop |
| drwxr-xr-x 2 cursoemvideo cursoemvideo 4096 fev 14 09:35 Documents |
| drwxr-xr-x 2 cursoemvideo cursoemvideo 4096 fev 14 09:35 Downloads |
|-----|
```

```
-      -      -      -      -      -      -
-      Leitura de um comando $ ls -l      -
-      -      -      -      -      -      -
```

```
d r w x r-xr-x 2 usuario usuario 4096          set 1 19:29          Desktop
|              |          |          |          (Data criação/modificação)
|              |          |          |
|              |          |          |
|              |          |          |
|              |          |          |=> Tamanho da pasta (aqui é 4kb)
|              |          |          |
|              |          |          |==> Grupo Proprietário
|              |          |          |
|              |          |          |==> Usuário Proprietário
|              |          |          |
|              |          |          |==> Número de pastas que tem dentro
```

```
|=====> Essa posição marca o tipo de arquivo. Nesse caso é um diretório (" d ")
|      " - " Se for um arquivo normal
|      " b " Se fosse um dispositivo de block
|      " l " Se fosse um link (Link é um atalho, um arquivo tipo atalho)
```

=====

-----

```
$ls /nome_do_diretório //acessa o conteúdo de dentro do diretório, mas não entra nele.
```

```
$ls .. //acessa o conteúdo de dentro do diretório anterior, mas não entra nele.
```

-----

```
$touch Nome.txt //Cria um arquivo de texto comum. Por esse comando é possível  
fazer mais coisas como alterar data de criação, etc
```

#### DICAS:

Durante a entrada de texto, pode-se utilizar a tecla TAB para autocompletar a palavra

#1 vez a tecla TAB completa a palavra

#2 vezes a tecla TAB é usado quando a primeira vez não funciona, isso ocorre quando existe mais de uma palavra que começa da mesma forma. O resultado disso é a exibição no console todas as palavras que estão começando da mesma forma

-----

```
|-----|  
| $history //Mostra o histórico de todos os comandos, em teoria o linux salva até |  
| //2 mil comandos no histórico (achei comando usado à vários dias atrás)|  
| $!92 //Selecionou o comando que foi usado na nonagésima segunda vez |  
|-----|
```

```
|-----|  
| $ nano nome.txt //Usa o editor de texto nano, padrão no Linux |  
| //Dá pra salvar os documentos em formatos para outros S.O |  
|-----|
```

```
|-----|  
$rm -rf /*~ //Apaga todos os arquivos com "~"  
$rm -rf /* ~ //Apaga.....o Linux, e espaço em branco MATA  
|-----|
```

----

Ricardo “Vamos dizer que estamos na pasta aqui abaixo”



/etc/

“Aqui damos um ‘ls’ e vemos todas essas pastas”

adduser.conf	guest-session	polkit-1
alsa	hddtemp.db	popularity-contest.conf
alternatives	hdparm.conf	ppp
anacrontab	host.conf	printcap
apache2	hostname	profile
Zapg.conf	hosts	profile.d
apm	hosts.allow	protocols
apparmor	hosts.deny	pulse
apparmor.d	hp	python
appport	ifplugd	python2.7
//A lista original é muito maior que essa		

“Mas ae é que tá, eu só quero os arquivos .conf”

“E agora o que faço?”

COMANDO:

\$	ls	/etc/*.conf
/etc/adduser.conf	/etc/insserv.conf	/etc/pnm2ppa.conf
/etc/appstream.conf	/etc/kernel-img.conf	/etc/resolv.conf
/etc/brltty.conf	/etc/kerneloops.conf	/etc/rsyslog.conf
/etc/ca-certificates.conf	/etc/ld.so.conf	/etc/rygel.conf
/etc/debconf.conf	/etc/libao.conf	/etc/sensors3.conf
//A lista original é muito maior que essa		

Ricardo “Ta ae ó”

“O ‘ \* ‘ substitui qualquer coisa, porque ele é um caractere reservado”.

O...asterisco...substitui qualquer coisa.

Vamos supor que estamos na pasta tal

```
|-----|
| usuario@usuario:~Pasta_tal$ ls |
|                                     |
| blenderplayer      icons          Python-license.txt |
| blender-softwaregl jemalloc-license.txt  readme.html |
| blender.svg        lib |
| blender-thumbnailer.py LICENSE-bfont.ttf.txt |
|-----|
```

Ricardo: “Mas você quer ver os arquivos que se chamem ‘blend’ ” ;

```
|-----|
| usuario@usuario:~Pasta_tal$ ls *blend* |
|                                     |
| blender      blenderplayer      blender.svg |
| blender.desktop blender-softwaregl blender-thumbnailer.py |
|-----|
```

(12:50)

Ricardo “”””No DOS isso também funciona, só que é mais básico””””.

-----

(14:00)

Ricardo: “O ponto de interrogação ‘ ? ’ funciona de forma semelhante ao asterisco ‘ \* ’, mas o interrogação não substitui uma sentença inteira, ele substitui apenas UM CARACTERE”.

Esse “ ? ” não é muito usado, mas é útil quando precisamos definir os detalhes da pesquisa

-----

Perceba que estamos lidando com caracteres globais. Desde o início da aula 8 já foi falado sobre 2 caracteres globais: O asterisco e o ponto de interrogação.

Agora vamos abordar o terceiro caractere global, o colchetes ' [ ] ' que são usados para pegar FAIXA de caracteres.

Ricardo/Guanabara: ""ISSO NÃO TEM NO DOS"".

```
|-----|
| $ ls /localizacao/f[a-i]* |
|                               |
|                               |
|                               |
|                               |
|                               |
|                               |
|                               |
|                               |
|                               |
|                               |
|-----|
```

\_\_ Significa que o caractere que voce está pesquisando tem um caractere nessa posição depois do ' f ' que varia entre a LETRA ' a ' E À LETRA ' i '

```
|-----|
|$ ls /localizacao/f[a-c,e]* //busca por caracteres que sejam entre a |
|                               //até ' c ', e mais a letra ' e '. |
|-----|
```

Tá vendo que tem um tracinho ' - ' que indica o caractere inicial até o caractere final?

Ricardo: "As versões mais novas do bash conseguem entender a mesma instrução sem esse tracinho";

=====

Ricardo: “O último caractere global que vamos falar agora é a chave, as duas chaves ‘{ }’ ”.

“É natural se confundir com os dois chaves”.

“As chaves são usadas para fazer PADRÕES de strings, que são padrões de caracteres”

```
-----|
|          $      ls      /localizacao/{string_aleatória,string_aleatória2} |
|-----|
```

INFO: Parece com o colchetes, mas esse busca por uma string inteira

```
-----|
|          $      ls      /localizacao/*{string_aleatória,string_aleatória2}* |
|-----|
```

INFO: O asterisco também pode ser usado, assim como nos colchetes

```
-----|
|          $      ls      /localizacao/*.pdf}    //Busca pela extensão PDF |
|          $      ls      /localizacao/*.docs}   //Busca pela extensão DOCX |
|-----|
|          $      ls      /localizacao/*.docs,.png,.jpeg} //Busca pela extensão |
|-----|
```

Ricardo: “Eu sempre falo, no Linux a gente está sempre descobrindo coisas, até eu que uso Linux ah mais de 20 anos”;

```
-----|
$cat /etc/services      //Ele pega o conteúdo do arquivo e mostra na tela
                        //Dá até pra jogar a informação de um executável na tela....não da pra
                        entender nada, mas ele consegue jogar o conteúdo na tela do mesmo jeito.
-----|
```

\$less nome\_do\_arquivo //Ele é próprio para fazer a paginação do arquivo, muito útil quando o arquivo é muito grande, está cheio de informações, tipo um manual sobre alguma coisa

“Pode ser usado no lugar no cat para faz a paginação do arquivo”

“ ‘ h ’ abre o menu dele”

“O less é muito prático”

“Dá pra fazer buscar com ele”

‘/nome\_do\_que\_quer\_pesquisar’ “E ele busca por todo o arquivo”

‘ :Q ’ “Você sai dele”

-----  
Ricardo: “Faz parte da filosofia do Linux que cada comando faça bem uma atividade específica, o \$man usa trás informações sobre outros programas, mas ele também usa o less para formatar a visualização, para facilitar o entendimento do usuário ”.  
-----

\$tac //faz a mesma coisa que o cat....só que de trás pra frente. Ele pega o arquivo e mostra na tela, só que da última linha para a primeira.  
//Muito útil em checagem de lista, buscando pela ordem de precedência.  
-----

\$cp nome\_do\_arquivo.txt /pasta\_destino //copia arquivo para a pasta desejada

Ricardo: “ Primeiro vc escreve da onde ele vem, e depois para onde ele vai”

“Vamos copiar arquivos de uma pasta específica, mas só vamos fazer isso com arquivos que tem a extensão .conf”.

\$cp /etc/\*.conf localização\_da\_pasta  
-----

Ricardo: “apelido para o comando (alias), vamos usar ele para o comando \$rm”.  
“Enquanto o terminal que foi definido o apelido estiver aberto, o alias permanece configurado para aquele comando que foi definido”.

\$alias rm='/bin/rm -l' //Ricardo: “agora toda vez que usar o \$rm para apagar mais De 3 arquivos, irá retornar um aviso de confirmação antes de executar a ação ”

Ricardo: “É um comando bom para iniciante. Ele evita com que você faça besteira”.

-----

## DÚVIDAS COMUNS ENTRE USUÁRIOS:

> Meu pendrive recebeu uma letra associada à ele, porque essa letra não aparece no Linux?

Resposta:

> Porque o Linux atribui letras aos dispositivos.

Ricardo: “Muitos que vinham do DOS me perguntavam como fazia eles para renomear um arquivo”.

“Para isso também usamos o comando para mover o arquivo \$mv porque você move ele pro mesmo local onde ele se encontra”.

NOTA: estamos aqui na pasta e temos as seguintes pasta:

Pasta1

Vamos renomeá-la agora

```
$mv nome_arquivo novo_nome_do_arquivo //a pasta foi re-nomeada.
```

-----

Gustavo: “Seja paranoico”, faça sempre BACKUP. -> “Se você tem 1, você não tem nenhum!  
Se você tem 2, você tem 1”.

-----