

The Linux Command Line 6th, Internet Edition -> <http://sourceforge.net/projects/linuxcommand/>

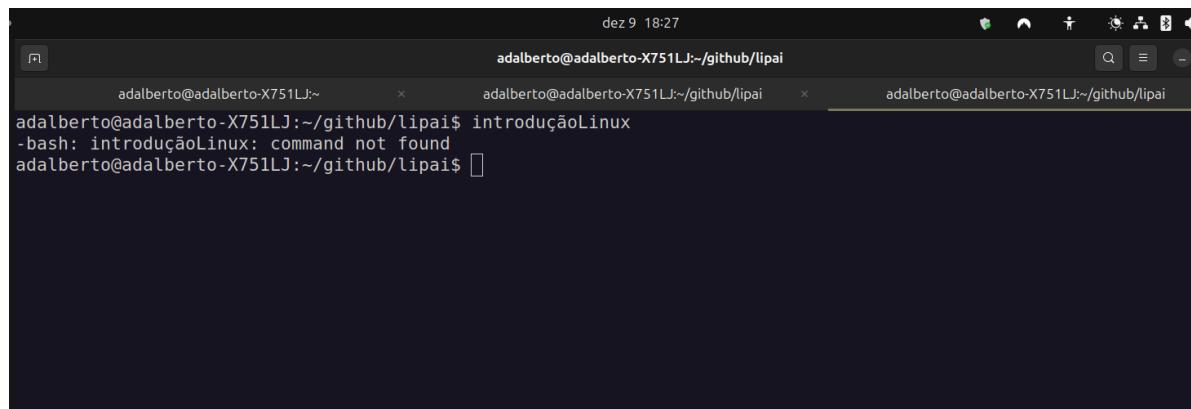
Capítulo 1

Primeiro contato com o terminal

```
# escrevendo qualquer coisa no terminal e observando a resposta  
comandoDigitado  
-bash: comandoDigitado: command not found
```

. Anotação

Aparece uma mensagem de erro pois não é localizada a informação no sistema



Alguns comandos simples no terminal linux

Comando	Função	Observações
date	Mostra a data no terminal	a data que aparece é completa e depende do que foi configurado no sistema
uptime	Mostra o tempo que o pc está em atividade	um dado interessante é que o uptime mostra a carga que está no pc sendo 1.0 para carga de 100% como meu pc tem 4 cores a carga está ok, no uptime também consigo ver o que estava carregado nos tempos de 1 minutos, 5 minutos e 15 minutos
df	Mostra a quantidade de espaço livre no hd	gosto de usar a opção df -hT que me mostra de maneira mais fácil de entender

Comando	Função	Observações
free	Mostra a quantidade de memória livre no sistema	algumas vezes o sistema trava devido ao google-chrome e o free me ajuda a ver esses dados

```

dez 9 18:37
adalberto@adalberto-X751LJ:~/github/lipai
adalberto@adalberto-X751LJ:~/github/lipai$ free
              total        used      free    shared   buff/cache  available
Mem:       11657980     8138264     571156    967740      4236852     3519716
Swap:      12582908      42232   12540676
adalberto@adalberto-X751LJ:~/github/lipai$ uptime
  ter 09 dez 2025 18:36:28 -03
adalberto@adalberto-X751LJ:~/github/lipai$ df -h
Filesystem      Type  Size  Used Avail Use% Mounted on
tmpfs           tmpfs  1,2G  28M  1,1G  3% /run
/dev/sda2        ext4  439G  384G  33G  93% /
tmpfs           tmpfs  5,6G  407M  5,2G  8% /dev/shm
tmpfs           tmpfs  5,0M  16K  5,0M  1% /run/lock
efivarfs        efivarfs 128K  104K  20K  85% /sys/firmware/efi/efivars
tmpfs           tmpfs  5,6G  0  5,6G  0% /run/qemu
/dev/sdal        vfat   511M  6,2M  505M  2% /boot/efi
tmpfs           tmpfs  1,2G  184K  1,2G  1% /run/user/1000
adalberto@adalberto-X751LJ:~/github/lipai$ df -k
Filesystem      1K-blocks  Used Available Use% Mounted on
tmpfs           1165800    28248   1137552  3% /run
/dev/sda2        459850824 402631884 33786288  93% /
tmpfs           5828988    413476   5415512  8% /dev/shm
tmpfs           5120        16      5104  1% /run/lock
efivarfs        128         104     20  85% /sys/firmware/efi/efivars
tmpfs           5828988     0      5828988  0% /run/qemu
/dev/sdal        523248    6288   516960  2% /boot/efi
tmpfs           1165796    184    1165612  1% /run/user/1000
adalberto@adalberto-X751LJ:~/github/lipai$ exit

```

Saindo do terminal

. Para sair do terminal ou encerrar o mesmo usamos o comando exit

```
# saída do terminal linux
exit
```

Curiosidades

. É importante notar que podemos abrir diversos terminais no linux mesmo que não estando em modo gráfico através da combinação de teclas Ctrl + Alt + tecla de função F1 até F6, em quase todas as distribuições.

. Para mudar de ambiente e ir para outro pode ser usar a combinação da tecla Alt + tecla do ambiente original que foi inicialmente usado , se usou Ctrl + Alt + F1 e depois Ctrl + Alt + F2 , ao apertar Alt + F1, iremos para o ambiente gerado no F1.

. Notar que cada distribuição linux pode ter um detalhe diferente nessa parte, na versão Ubuntu 24.04 no meu pc ao fazer Ctrl + Alt + F1 eu fecho a área de trabalho

Capítulo 2

Navegação

. Os comandos básicos de navegação no terminal linux são:

```

# mostra o diretório que estamos atualmente
pwd

# Muda o diretório atual
cd novo_diretório

# Lista o diretório atual, com todas as 'coisas' dentro.

ls -alsht

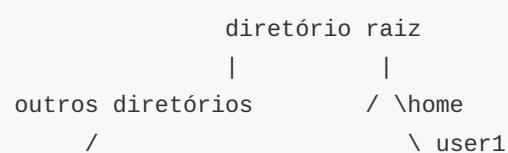
```

```

adalberto@adalberto-X751LJ:~$ pwd
/home/adalberto/github/lipai
adalberto@adalberto-X751LJ:~/github/lipai$ cd imgs
adalberto@adalberto-X751LJ:~/github/lipai/imgs$ ls -alsht
total 316K
4,0K drwxrwxr-x 3 adalberto adalberto 4,0K dez  9 19:04 ..
4,0K drwxrwxr-x 2 adalberto adalberto 4,0K dez  9 18:38 .
264K -rw-rw-r-- 1 adalberto adalberto 262K dez  9 18:37 comandos_date_uptime_df_free.png
44K -rw-rw-r-- 1 adalberto adalberto 44K dez  9 18:28 comando_zero.png
adalberto@adalberto-X751LJ:~/github/lipai/imgs$ 

```

. A árvore de diretórios dos sistemas baseados em Unix , considera apenas um diretório central e embaixo dele cria diferentes ramos para cada necessidade.



```

adalberto@adalberto-X751LJ:~/github/lipai$ tree ~/github/lipai
/home/adalberto/github/lipai
└── atividade_S1_A2.md
    └── imgs
        ├── comandos_date_uptime_df_free.png
        ├── comandos_pwd_cd_ls.png
        └── comando_zero.png
2 directories, 4 files
adalberto@adalberto-X751LJ:~/github/lipai/imgs$ 
```

. Ao fazermos uma navegação podemos usar o cd caminho do diretório que queremos ir usando o caminho absoluto que é o caminho desde o diretório raiz até o diretório destino ou então podemos usar um caminho relativo usando ' .' ou " .. " , dependendo de onde quer ir.

```

# caminho absoluto
cd /caminho absoluto

# caminho relativo
cd ./bin 
```

```

T adalberto@adalberto-X751LJ:~/github/lipai/imgs$ pwd
/home/adalberto/github/lipai/imgs
adalberto@adalberto-X751LJ:~/github/lipai/imgs$ cd /home/adalberto/github/lipai
adalberto@adalberto-X751LJ:~/github/lipai$ pwd
/home/adalberto/github/lipai
adalberto@adalberto-X751LJ:~/github/lipai$ cd ./imgs
adalberto@adalberto-X751LJ:~/github/lipai/imgs$ 
```

Capítulo 3

Comando	Função, Observações
ls -a	Lista todos os arquivos, mostra inclusive os arquivos ocultos que começam com
ls -l	Formato longo,"Exibe permissões, dono, tamanho e data. É o que mais uso para ver detalhes"
ls -h	Tamanho legível,"Mostra o tamanho em K, M, G (humano) em vez de bytes. Gosto de usar junto com o -l (ls -lh)"

Comando	Função, Observações
ls -r	Inverte a ordem, Útil para ver os últimos arquivos se combinado com ordenação por tempo
ls -S	Ordena por tamanho, Ajuda a achar arquivos grandes
ls -t	Ordena por tempo, Mostra os modificados recentemente primeiro

```
# listando arquivos com detalhes, tamanho legível e ordenado por tempo (mais recentes no fim)
ls -lhtr

# a opção r é que mostra a reversão
```

```
adalberto@adalberto-X751LJ:~/github/lipai/imgs$ ls -lhtr
total 640K
-rw-rw-r-- 1 adalberto adalberto 44K dez  9 18:28 comando_zero.png
-rw-rw-r-- 1 adalberto adalberto 262K dez  9 18:37 comandos_date_uptime_df_free.png
-rw-rw-r-- 1 adalberto adalberto 137K dez  9 19:05 comandos_pwd_cd_ls.png
-rw-rw-r-- 1 adalberto adalberto 108K dez  9 19:17 comando_tree.png
-rw-rw-r-- 1 adalberto adalberto 81K dez  9 19:24 comando_caminhos_absoluto_relativo.png
adalberto@adalberto-X751LJ:~/github/lipai/imgs$
```

Descobrindo o tipo de arquivo e visualizando conteúdo

No Linux, a extensão do arquivo (como .txt ou .jpg) não importa tanto. Para saber o que o arquivo realmente é, usamos o comando file. Para ler arquivos de texto longos sem abrir um editor, usamos o less.

```
# mostra os dados do arquivo comando_ls_arquivos.png

file comando_ls_arquivos.png
# mostra os dados do arquivo , como é um arquivo de imagem mostra a dimensão, o tipo de cor, o padrão de cor no caso RGBA de 8 bits
```

```
adalberto@adalberto-X751LJ:~/github/lipai/imgs$ ls
comando_caminhos_absoluto_relativo.png  comandos_date_uptime_df_free.png  comando_tree.png
comando_ls_arquivos.png                 comandos_pwd_cd_ls.png        comando_zero.png
adalberto@adalberto-X751LJ:~/github/lipai/imgs$ file comando_ls_arquivos.png
comando_ls_arquivos.png: PNG image data, 1595 x 541, 8-bit/color RGBA, non-interlaced
adalberto@adalberto-X751LJ:~/github/lipai/imgs$
```

Comando less

. faz a listagem do conteúdo de um arquivo que é muito grande para se ver de uma vez só no terminal.

Comando	Função
q	faz a saída do less
/texto	procura um texto que está no arquivo
G	vai para o final do arquivo
1G	vai para o início do arquivo

Estrutura de diretórios do linux

Diretório	Função	Observações
/	Raiz do sistema	Tudo começa aqui
/bin	Binários essenciais	"Onde ficam comandos como ls, cp, mv"
/etc	Configurações	Onde ficam os arquivos de configuração do sistema (quase tudo texto)
/home	Arquivos dos usuários	Onde fica a minha pasta /home/adalberto e meus documentos
/root	Home do superusuário	Área restrita do administrador (root)
/var	Arquivos variáveis	Onde ficam logs (/var/log) e arquivos que mudam constantemente
/usr	Recursos do sistema	Onde ficam os programas instalados para usuários (/usr/bin)

Capítulo 4

Manipulando Arquivos e Diretórios.

Usando Curingas (Wildcards)

. O shell usa caracteres especiais para selecionar grupos de arquivos rapidamente.

Curinga	Significado	Observações
*	Qualquer caractere	O mais usado. <code>rm *.html</code> apaga todos os html. Cuidado com o espaço! 444
?	Um único caractere	Útil quando os nomes variam apenas por uma letra ou número 5

Curinga	Significado	Observações
[]	Conjunto de caracteres	[abc]* pega arquivos começando com a, b ou c 6
[:digit:]	Apenas números	Uso para filtrar arquivos que têm números no nome 7

Copia todos os arquivos que terminam com .txt para a pasta Documentos

```
cp *.txt Documentos/
```

Criar um diretório

```
mkdir novo_diretório  
# ou  
mkdir ~/github/lipai/novo_diretorio
```

Copia arquivos de um local

```
cp /home/user/adalberto/texto.txt ~/github/lipai/novo_diretorio
```

Move arquivos e renomeia se for no mesmo diretório

```
# renomeia arquivo  
mv arquivo.txt novo_nome_arquivo.txt  
# move arquivo para outro diretório  
mv arquivo.txt /diretório_novo/arquivo.txt
```

Apaga um arquivo pode ser recursivo (r)

```
# apaga os arquivos txt que estão no diretório e nos subdiretórios  
rm -rf *.txt  
  
# apaga um arquivo  
rm arquivo.txt
```

Links (Hard e Simbólicos).

- . O comando `ln` cria links.
- . Existem dois tipos: Hard Links (antigos, não podem cruzar partições) e Links Simbólicos (mais modernos, parecem atalhos do Windows)

Tipo	Comando	Observações
Hard Link	<code>ln arquivo link</code>	Os dois arquivos são idênticos e apontam para o mesmo lugar no disco
Symbolic Link	<code>ln -s arquivo link</code>	Se apagar o link, o arquivo original fica lá

Capítulo 5

Trabalhando com Comandos

- . Até agora usamos uma série de comandos "misteriosos". [cite_start]Neste capítulo, vamos tirar esse mistério e descobrir o que eles realmente são. [cite: 3, 4]
- . [cite_start]Um comando pode ser: um programa executável (binário), um comando interno do shell (builtin), uma função do shell ou um alias. [cite: 18]

Identificando Comandos

- . É útil saber exatamente que tipo de comando estamos usando. O Linux nos dá ferramentas para descobrir isso.

Comando	Função	Observações
<code>type</code>	Mostra o tipo do comando	[cite_start]Descobri que o <code>ls</code> na verdade é um alias para <code>ls --color=auto</code> no meu sistema [cite: 31, 40]
<code>which</code>	Mostra onde está o executável	Ele mostra o caminho completo (ex: <code>/usr/bin/cp</code>). [cite_start]Só funciona para programas reais, não para comandos internos como o <code>cd</code> [cite: 42, 47]

```
# Verificando o que é o comando ls
type ls
# Saída: ls is aliased to `ls --color=auto'

# Verificando onde está o executável do navegador
which firefox

## Conseguindo Ajuda (Documentação)
. O Linux possui muita documentação embutida. Não precisamos decorar tudo, apenas saber onde procurar.
```

Comando	Função	Observações
help	Ajuda para builtins	Uso para comandos internos do shell como <code>cd</code> ou <code>exit</code> , que não têm manual comum
--help	Resumo de uso	A maioria dos programas aceita isso (ex: <code>mkdir --help</code>) para mostrar as opções rapidamente
man	Manual completo	O <code>man ls</code> mostra tudo. Uso as setas para ler e a tecla <code>/</code> para pesquisar um texto lá dentro
apropos	Pesquisa no manual	Útil quando não sei o nome do comando, mas sei o que quero fazer. Ex: <code>apropos partition</code>
whatis	Descrição de uma linha	Mostra um resumo muito breve do que o comando faz

Criando meus próprios comandos (Alias)

. Podemos criar nossos próprios comandos usando o `alias`. . Um truque legal antes de criar o alias é saber que podemos usar `;` para colocar vários comandos numa linha só.

```
# Criando um comando chamado 'foo' que entra em /usr, lista o conteúdo e volta
para onde eu estava
alias foo='cd /usr; ls; cd -'

# Para remover o alias criado
unalias foo
```

. Aliases criados diretamente no terminal somem quando fecho a janela (sessão). Depois aprenderei a salvar eles para sempre nos arquivos de inicialização.