

LINUX

MESTRE DOS DIRETÓRIOS



DIOGO LOUVO PIEVAN


INTRODUÇÃO

No universo Linux, tudo começa e termina nos diretórios. Eles são o esqueleto da organização do sistema — desde o kernel até seus scripts mais simples.

Se você é programador, sysadmin ou apenas curioso, entender como manipular diretórios no terminal é essencial. Aqui, você vai aprender os comandos mais importantes para criar, mover, renomear, copiar, deletar e explorar diretórios no Linux.

Este eBook foi feito para ser direto: cada comando é um capítulo, com aplicação prática. Prepare o terminal. Vamos explorar o Linux um diretório por vez.





CAPÍTULO

01

\$ mkdir

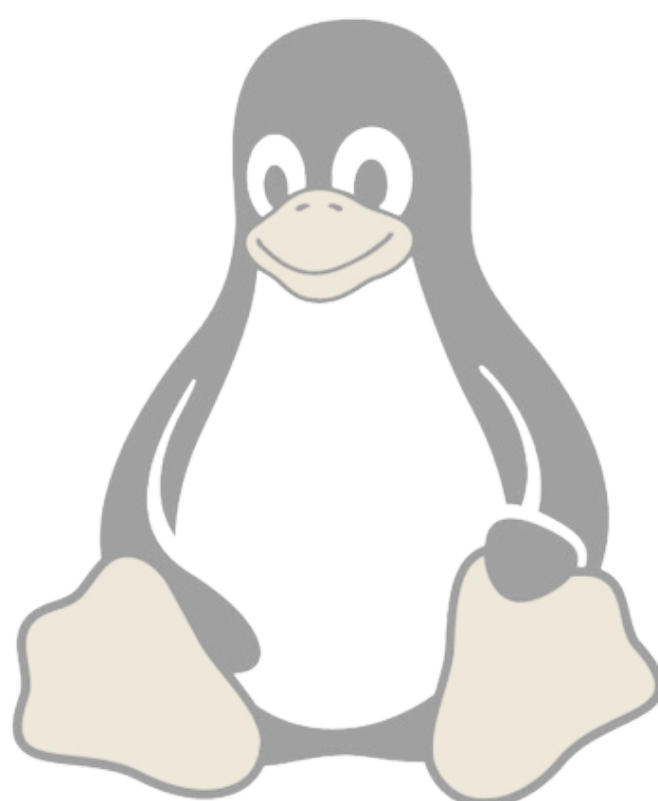
criação de diretórios

CRIAR DIRETÓRIOS

Criar diretórios no Linux é o primeiro passo para organizar seus arquivos de forma lógica. O comando **mkdir** (de make directory) permite criar pastas no sistema de maneira simples e direta, seja uma única pasta ou uma hierarquia completa.

```
○ ○ ○  
$ mkdir projects
```

Cria uma pasta chamada **projetos** no diretório atual.

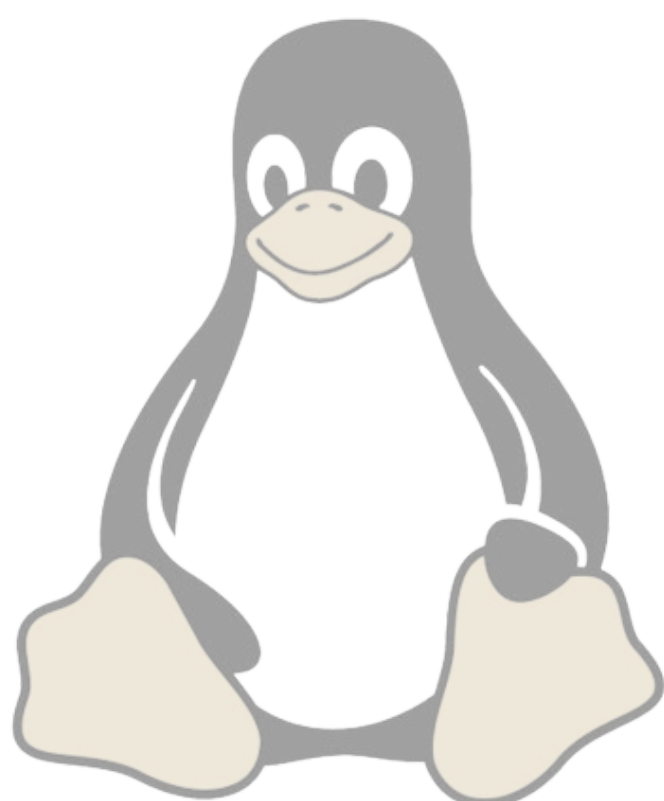



CRIAR DIRETÓRIOS



```
$ mkdir -p projects/frontend/src
```

Cria toda a estrutura de pastas **projects** → **frontend** → **src** de uma só vez, mesmo que as pastas intermediárias ainda não existam.





CAPÍTULO

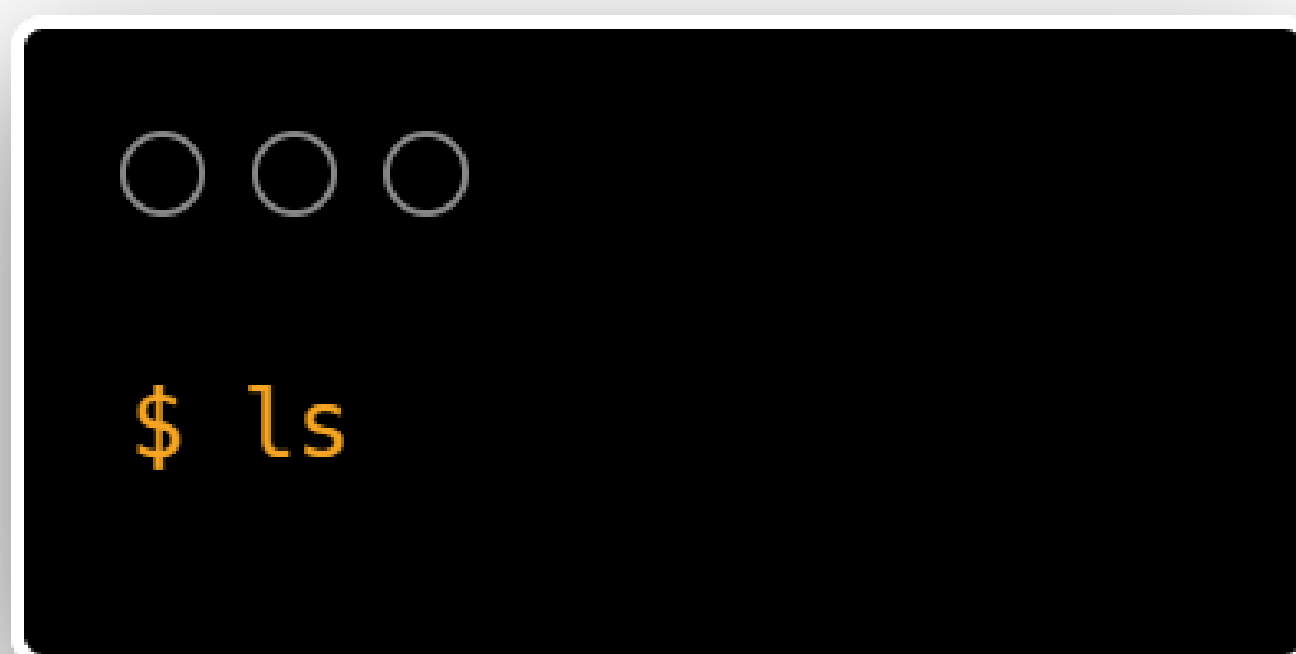
02

\$ Is

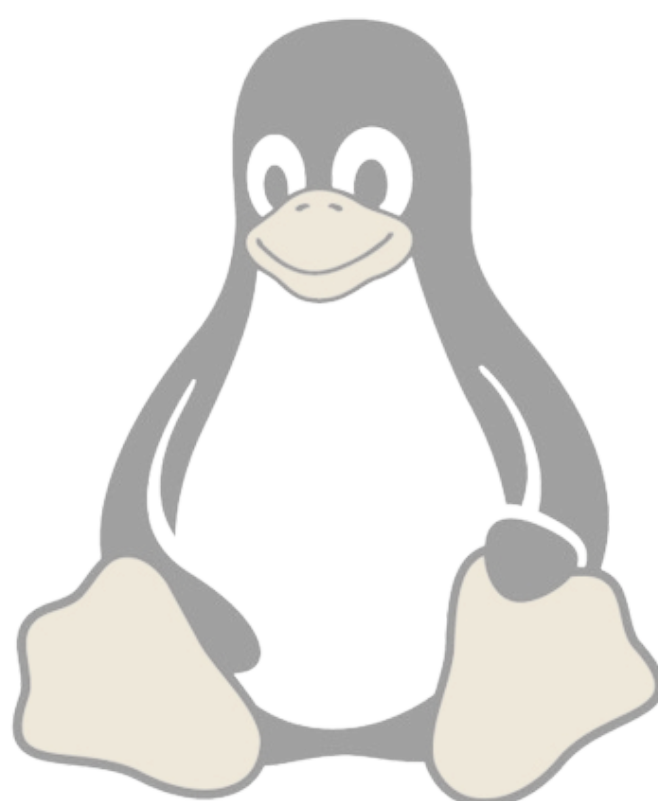
EXPLORANDO DIRETÓRIOS

LISTAR ELEMENTOS

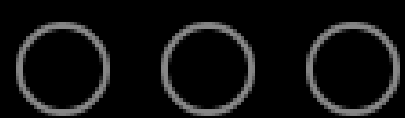
Depois de criar diretórios, o próximo passo é explorar o que há dentro deles. O comando **ls** (list) exibe os arquivos e pastas presentes em um diretório, permitindo que você visualize rapidamente o que está ali — com diferentes níveis de detalhe.



Lista os arquivos e diretórios no diretório atual, de forma simples.

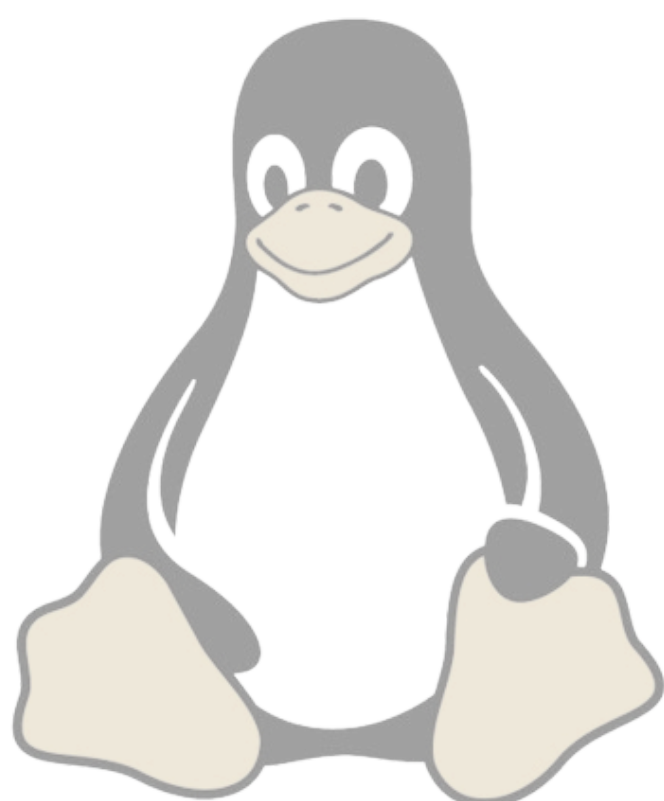



LISTAR ELEMENTOS



```
$ ls -lha
```

Mostra uma lista completa com permissões, tamanhos, datas e arquivos ocultos (os que começam com .). Ideal para quando você precisa entender a estrutura e os detalhes de um diretório.





CAPÍTULO

03

\$ cd

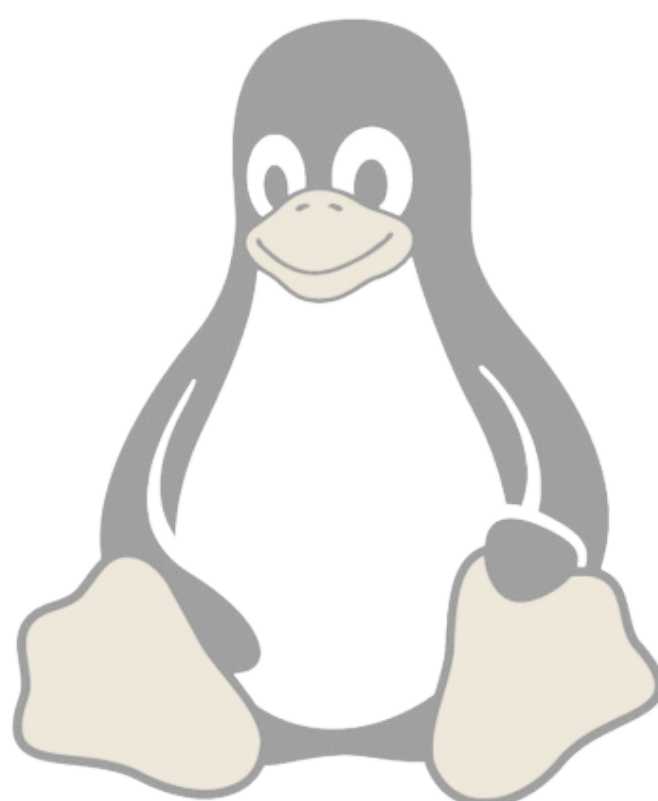
NAVEGANDO ENTRE DIRETÓRIOS

MUDAR DIRETÓRIO

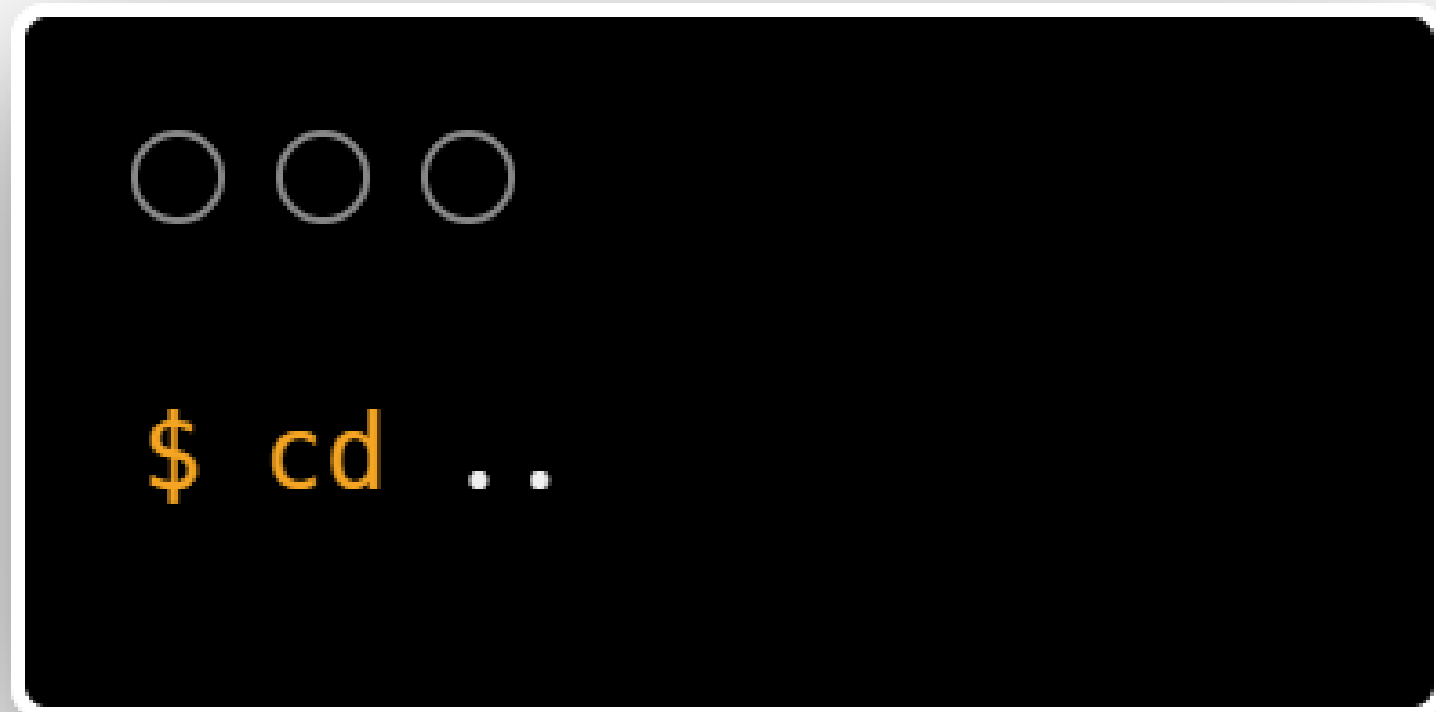
O comando **cd** (change directory) permite que você se mova entre diretórios no Linux. Ele é a forma mais rápida de ir de uma pasta para outra, navegar por estruturas complexas e acessar qualquer lugar do sistema.

```
○ ○ ○  
  
$ cd projects
```

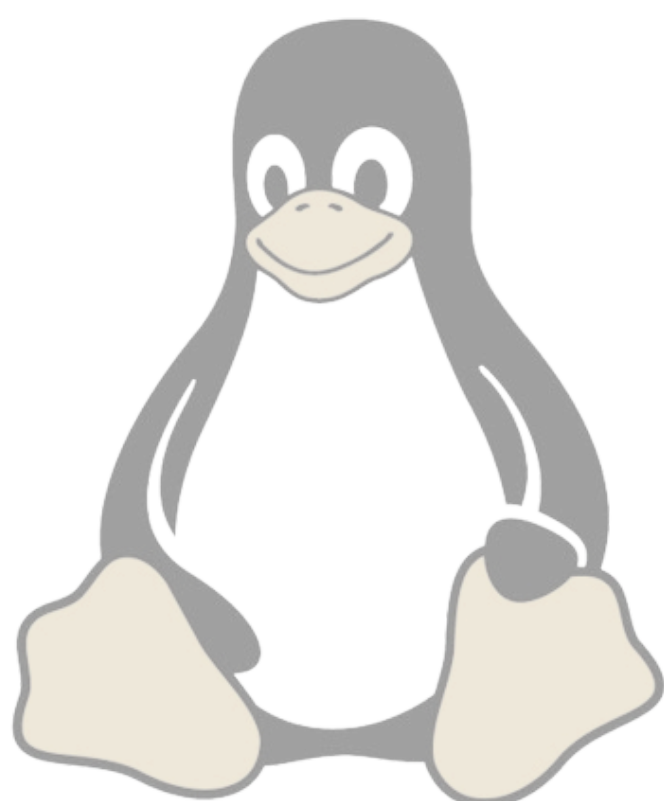
Entra no diretório **projects** a partir da pasta atual.

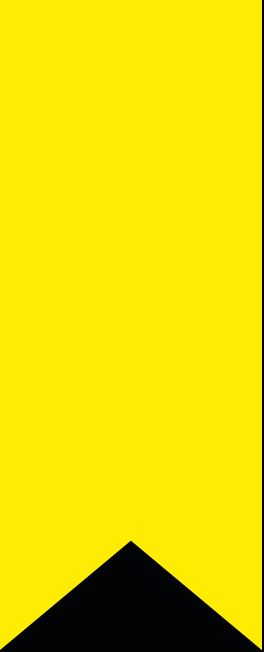


VOLTAR DIRETÓRIO



Retorna para o diretório anterior (pai), útil para subir na estrutura de diretórios sem precisar digitar o caminho completo.





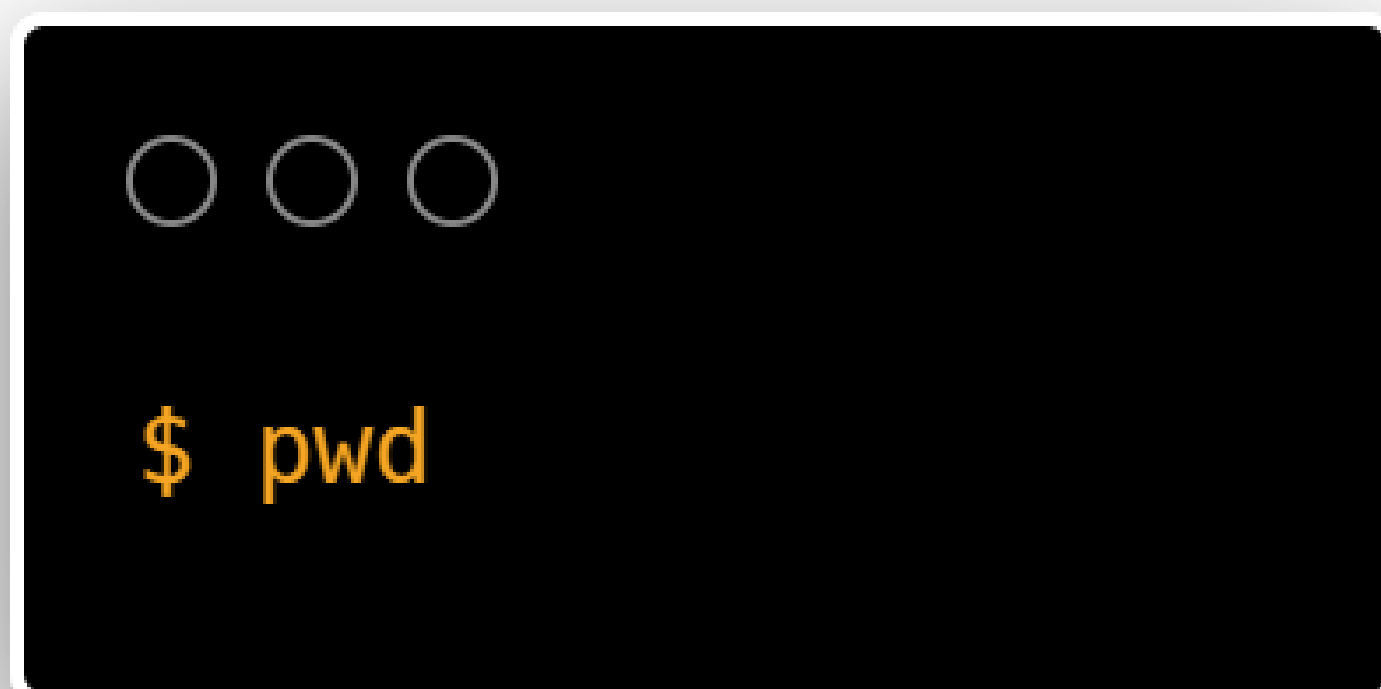
CAPÍTULO

04

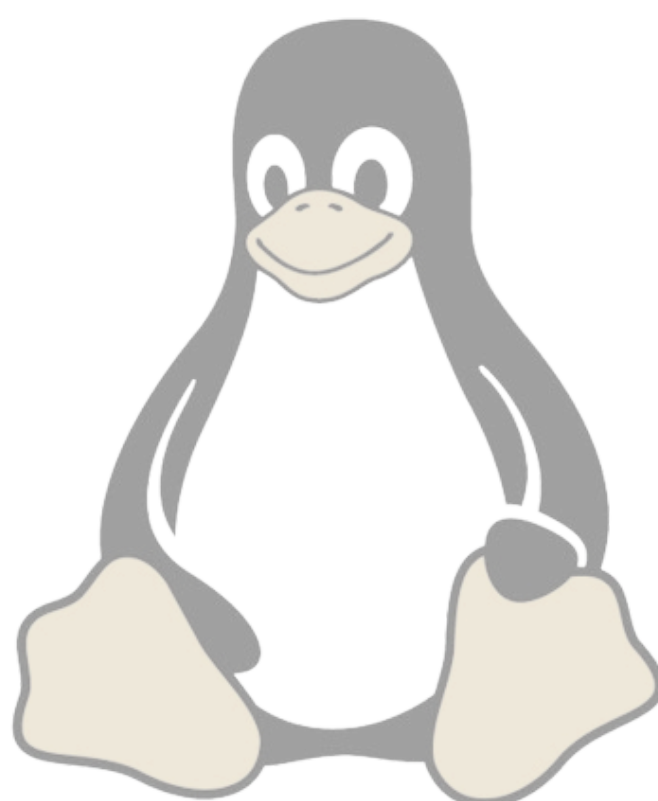
\$ pwd

CAMINHO ATUAL

O comando **pwd** (print working directory) mostra exatamente onde você está no sistema de arquivos. Isso é essencial quando você está navegando por múltiplos diretórios e precisa saber seu caminho completo.



Exibe o caminho absoluto do diretório atual.
Exemplo de saída: **/home/usuario/projetos/frontend**





CAPÍTULO

05

\$ rm -r

\$ rm -r

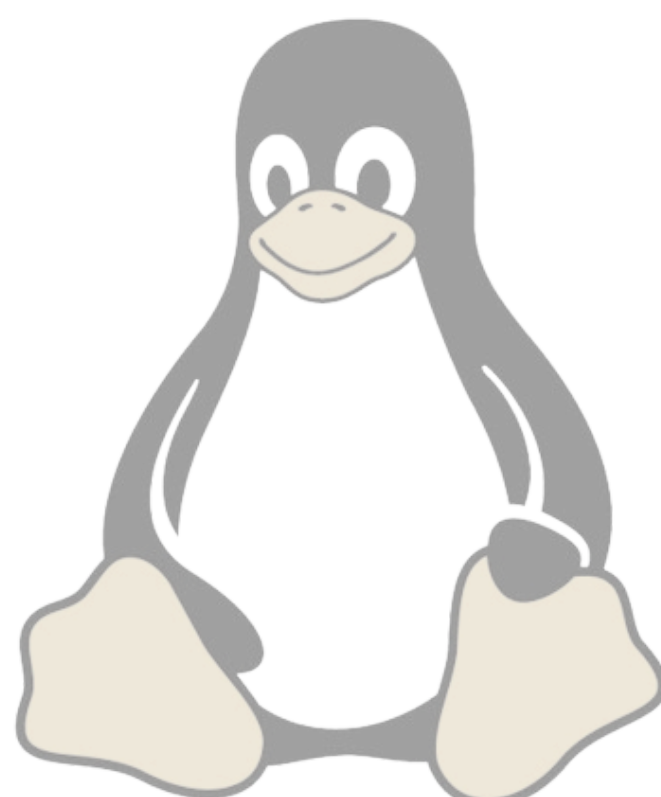
REMOVENDO DIRETÓRIOS

REMOVER DIRETÓRIO

Em algum momento, você vai precisar remover diretórios. Para isso, o Linux oferece duas opções: `rmdir` para pastas vazias e `rm -r` para remover diretórios com conteúdo. A diferença entre elas é crucial — e saber usá-las evita dor de cabeça.

```
○ ○ ○  
  
$ rmdir testes
```

Remove o diretório `testes`, mas somente se estiver vazio.



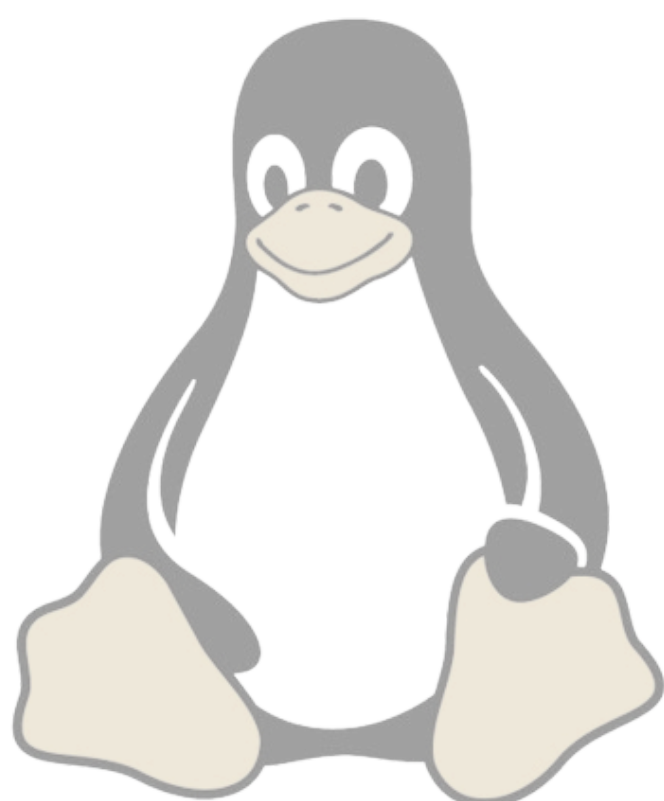
REMOVER DIRETÓRIO

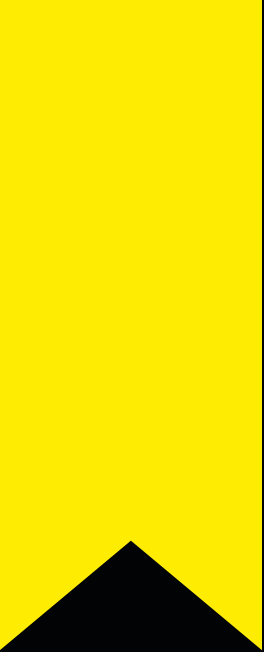


```
$ rm -r backups
```

Remove a pasta **backups** e tudo o que estiver dentro dela — subpastas, arquivos e tudo mais.

Use com atenção: o **rm -r** não pede confirmação por padrão e pode apagar muitos dados de uma vez.





CAPÍTULO

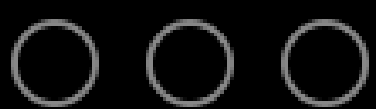
06

\$ mv

MOVENDO E RENOMEANDO

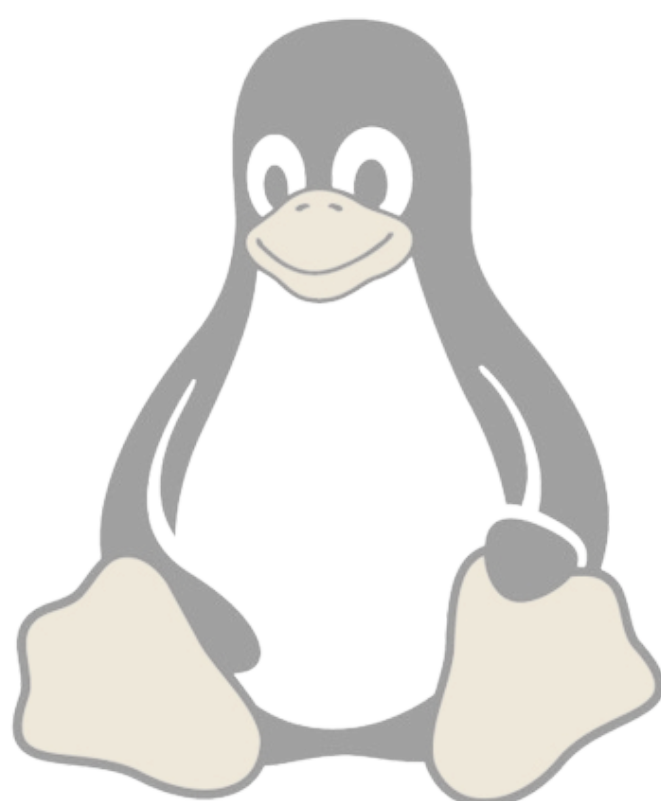
RENOMEAR DIRETÓRIO

O comando **mv** é versátil: ele move arquivos e diretórios para outro local, mas também é usado para renomeá-los. É uma ferramenta poderosa para reorganizar sua estrutura de pastas com rapidez e eficiência.



```
$ mv projects projeto-final
```

Renomeia a pasta **projects** para **projeto-final** no mesmo local.



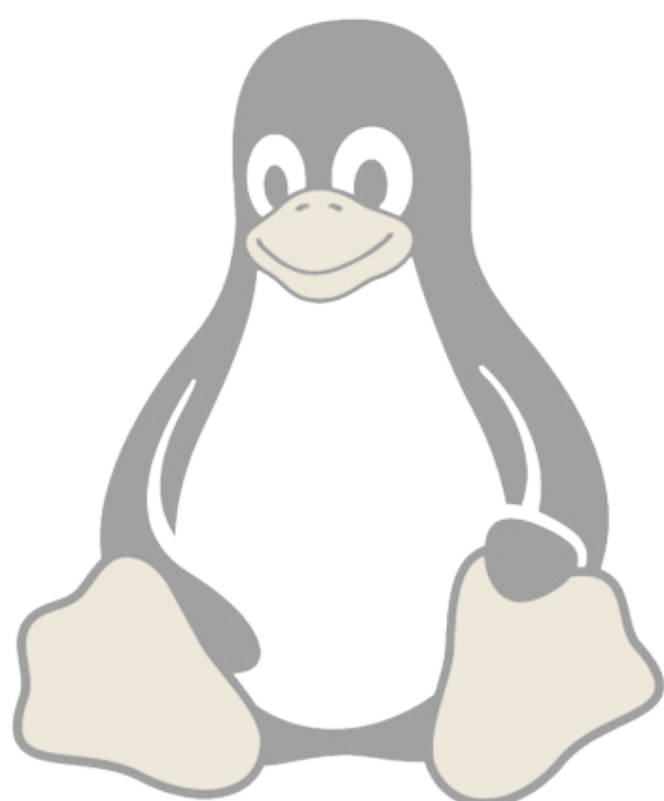
MOVER DIRETÓRIO




```
$ mv projeto-final ~/Documentos/
```

Move a pasta **projeto-final** para dentro do diretório **Documentos** do usuário.

Dica: ao mover algo para outra pasta mantendo o mesmo nome, o Linux entende que você quer transferir, não renomear.





CAPÍTULO

07

\$ cp -r

COPIANDO DIRETÓRIOS

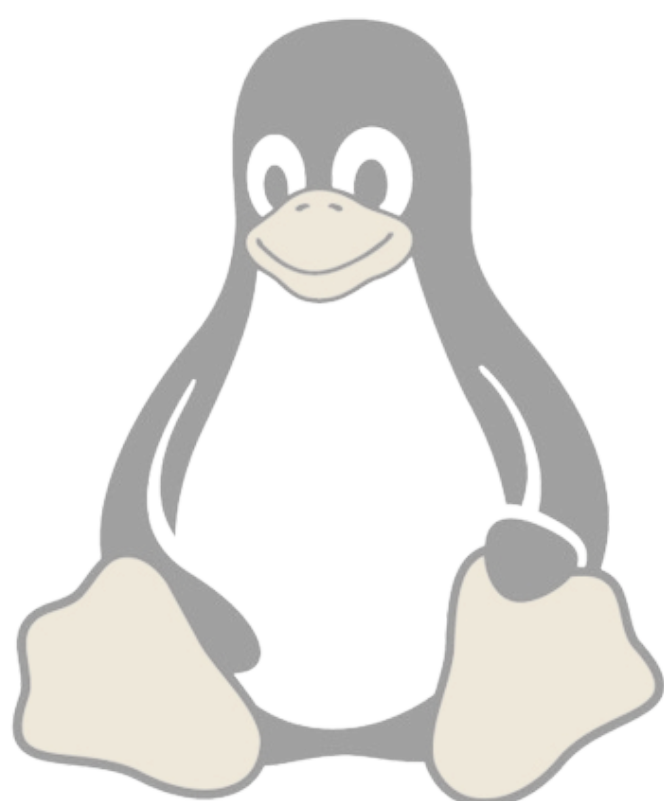
COPIAR DIRETÓRIO


Quando você precisa duplicar uma pasta inteira, o comando **cp -r** entra em cena. O **-r** (ou **--recursive**) é o que permite copiar diretórios inteiros, incluindo todos os seus arquivos e subpastas.



```
$ mv -r projeto-final projeto-copia
```

Cria uma cópia exata da pasta **projeto-final**, com o nome **projeto-copia**. Todo o conteúdo interno será replicado, mantendo a estrutura original.





CAPÍTULO

08

\$ find

BUSCANDO DIRETÓRIOS

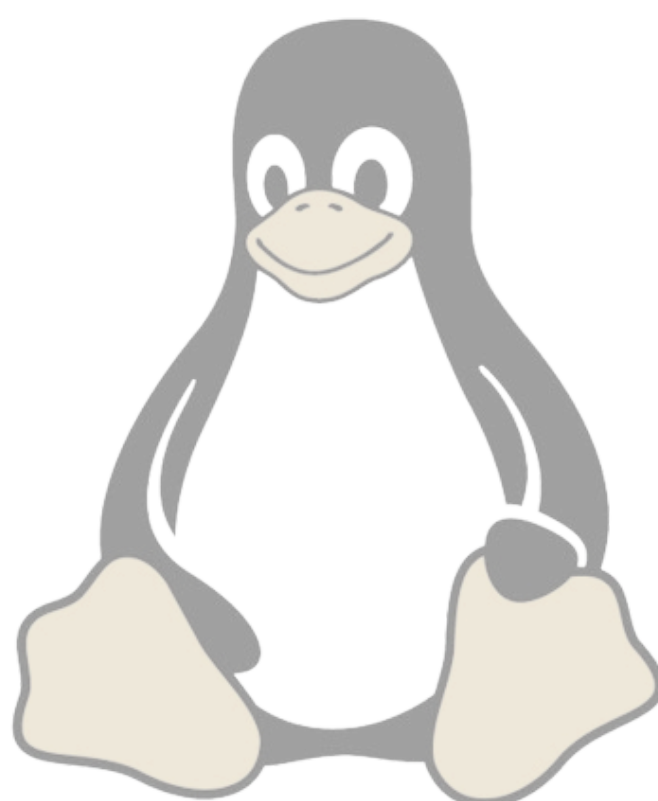
BUSCAR DIRETÓRIO

Em sistemas com muitas pastas e arquivos, localizar um diretório específico pode ser difícil. O comando **find** resolve isso, permitindo buscas precisas com base em nome, tipo e localização.



```
$ find . -type d -name "backup*"
```

A partir do diretório atual (.), procura por diretórios (-type d) cujo nome começa com “backup”. O asterisco (*) funciona como curinga.





AGRADECIMENTOS

FINAIS

AGRADECIMENTOS

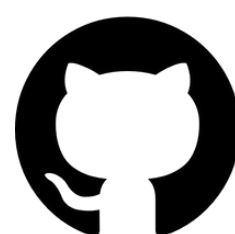
OBRIGADO POR LER ATÉ AQUI

Esse eBook foi gerado com o apoio de inteligência artificial, mas estruturado com cuidado humano. Toda a organização, capítulos e exemplos foram pensados para tornar o aprendizado de Linux mais direto e acessível para programadores.

Este conteúdo tem fins didáticos e foi criado como material de apoio ao estudo de terminal e sistemas Linux.

<https://github.com/diogolpievan/prompts-recipe-to-create-a-ebook/>

Autor: Diogo Louvo Pievan



github.com/diogolpievan



linkedin.com/in/diogolpievan