

Atalhos e Comandos no Terminal Essenciais para Iniciantes.

Aqui estão alguns dos principais atalhos de teclado para Linux e Windows, que podem variar um pouco dependendo do ambiente de desktop que você está usando (como GNOME, KDE, etc.). Aqui estão alguns atalhos gerais e comuns :

Índice.

ATALHOS DO TECLADO (Linux) :

- Para abrir uma nova aba ;
- Para abrir uma nova janela ;
- Para fechar a aba atual ;
- Para fechar a janela atual ;
- Para minimizar a janela atual ;
- Seleção do botão esquerdo do mouse ;
- Simulação do botão direito do mouse ;
- Navegação entre itens ;
- Navegação e Gerenciamento de Janelas ;
- Navegação de Arquivos ;
- Outros Atalhos Úteis ;
- Terminal (Linux) : **Ctrl + Alt + T.**

TERMINAL (Linux, Windows) :

- Atalhos no terminal;
- Comandos no Terminal;
- Navegação e Listagem;
- Busca;
- Criação e Remoção;
- Cópia e Movimentação;
- **Abrir arquivos de texto;**
- **Abrir qualquer arquivo com o programa padrão;**
- **Gerenciamento de Pacotes;**
- **Outros Comandos Úteis;**
- **Exemplo Prático.**

ATALHOS DO TECLADO :

Para abrir uma nova aba :

- **Windows/Linux** : Ctrl + T
- **Mac** : Command (⌘) + T

Para abrir uma nova janela :

- **Windows/Linux** : Ctrl + N
- **Mac** : Command (⌘) + N

Esses atalhos funcionam na maioria dos navegadores, como Google Chrome, Firefox, Edge e outros.

Para fechar a aba atual :

- **Windows/Linux** : Ctrl + W
- **Mac** : Command (⌘) + W

Para fechar a janela atual :

- **Windows/Linux** : Ctrl + Shift + W ou Alt + F4
- **Mac** : Command (⌘) + Shift + W ou Command (⌘) + Q (para fechar completamente o navegador)

Para minimizar a janela atual :

- **Windows/Linux** : Windows + M ou Alt + Espaço, seguido de N
- **Mac** : Command (⌘) + M

Esses atalhos funcionam na maioria dos navegadores e sistemas operacionais.

Para simular a seleção do botão esquerdo e direito do mouse via atalhos de teclado, você pode usar os seguintes métodos, dependendo do que deseja fazer :

Seleção do botão esquerdo do mouse :

- **Selecionar um item ou texto :**
 - **Shift + Setas** : Para selecionar texto ou itens um por um.
 - **Shift + Clique** : Para selecionar um intervalo de itens ou texto.

Simulação do botão direito do mouse :

- **Abrir o menu de contexto (botão direito) :**
 - **Shift + F10** : Isso geralmente abre o menu de contexto no Windows, permitindo acessar opções como copiar, colar, etc.
 - **Tecla de menu** (se disponível) : Algumas teclados têm uma tecla dedicada para abrir o menu de contexto. Ela é geralmente localizada entre a tecla Alt direita e a Ctrl direita.

Navegação entre itens :

- **Tab** : Move o foco para o próximo item interativo (como botões e links).
- **Shift + Tab** : Move o foco para o item anterior.

Esses métodos permitem interagir com elementos em uma interface gráfica sem o uso do mouse.

Navegação e Gerenciamento de Janelas:

- **Alt + Tab** : Alterna entre janelas abertas.
- **Super (Tecla Windows) + D** : Minimiza todas as janelas e mostra a área de trabalho.
- **Super + L** : Bloqueia a tela.
- **Ctrl + Alt + T** : Abre o terminal (Linux).

Navegação de Arquivos

- **Ctrl + N** : Abre uma nova janela de navegação em gerenciadores de arquivos.
- **Ctrl + W** : Fecha a janela atual.

Outros Atalhos Úteis

- **Alt + F2** : Abre o prompt de execução para iniciar aplicativos.
- **Ctrl + Shift + Esc** : Abre o gerenciador de tarefas (em alguns ambientes).

Esses atalhos podem variar um pouco com base na distribuição e no ambiente de desktop, mas são um bom ponto de partida para aumentar sua eficiência no Linux!

Linux

TERMINAL :

O terminal é uma ferramenta poderosa para interagir com o sistema de forma mais profunda e eficiente. Para te auxiliar nessa jornada, vou apresentar alguns dos comandos mais importantes que você deve conhecer :

Atalhos Comuns (Bash e PowerShell)

Estes atalhos funcionam de forma semelhante tanto no Terminal Bash quanto no PowerShell, tornando-os universais para operações básicas de edição e controle:

- **Ctrl + C:** Interrompe o comando ou processo atual em execução.
- **Ctrl + A:** Move o cursor para o início da linha (Bash) / Seleciona todo o texto na linha de comando ou painel de script (PowerShell).
- **Ctrl + E:** Move o cursor para o final da linha.
- **Ctrl + X:** Recorta o texto selecionado (principalmente em interfaces com capacidade de seleção, como ISE no PowerShell).
- **Ctrl + V:** Cola o texto copiado ou recortado.
- **Ctrl + Z:** Desfaz a última ação.
- **Tab:** Auto-completa comandos, nomes de arquivos e diretórios. Pressione repetidamente para ciclar pelas opções.
- **Ctrl + Shift + T:** abre nova aba no terminal.
- **Ctrl + Shift + N:** abre nova janela no terminal.
- **Ctrl + Shift + W:** fecha aba/janela atual no terminal.

Atalhos Específicos do Terminal Bash

O Bash (Bourne Again SHell) é o shell padrão na maioria das distribuições Linux e macOS. Estes atalhos são ótimos para navegação e edição de linha de comando:

- **Edição e Controle de Linha:**
 - **Ctrl + B:** Move o cursor um caractere para trás.
 - **Alt + B:** Move o cursor uma palavra para trás.
 - **Ctrl + F:** Move o cursor um caractere para frente.
 - **Alt + F:** Move o cursor uma palavra para frente.
 - **Ctrl + W:** Apaga a palavra anterior ao cursor.

- **Ctrl + U:** Apaga tudo do cursor até o início da linha.
- **Ctrl + K:** Apaga tudo do cursor até o final da linha.
- **Ctrl + Y:** Cola o texto que foi apagado com Ctrl + U, Ctrl + K ou Ctrl + W.
- **Alt + D:** Apaga a palavra à frente do cursor.
- **Histórico e Execução de Comandos:**
 - **Ctrl + R:** Pesquisa incremental inversa no histórico de comandos. Pressione novamente Ctrl + R para navegar entre os resultados.
 - **Ctrl + G:** Sai do modo de pesquisa do histórico.
 - **Ctrl + P** ou Seta para Cima (↑): Exibe o comando anterior no histórico.
 - **Ctrl + N** ou Seta para Baixo (↓): Exibe o próximo comando no histórico.
 - **!!:** Repete o último comando executado.
 - **!\$** ou **Alt + .:** Insere o último argumento do comando anterior.
- **Controle de Processos:**
 - **Ctrl + Z:** Pausa o comando atual (coloca-o em segundo plano). Use fg para trazê-lo de volta para o primeiro plano ou bg para continuar em segundo plano.
 - **Ctrl + D:** Envia um sinal de "fim de arquivo" (EOF), que pode encerrar a sessão do terminal (similar a `exit`).
- **Gerenciamento da Tela:**
 - **Ctrl + L:** Limpa a tela do terminal, mantendo a linha de comando atual no topo.

Atalhos Específicos do PowerShell

O PowerShell é um shell de linha de comando e linguagem de script desenvolvido pela Microsoft para automação e gerenciamento de configurações.

- **Edição de Texto e Comando:**
 - **Ctrl + Y:** Refaz a última ação desfeita.
 - **Ctrl + Home:** Move o cursor para o início do buffer de entrada.
 - **Ctrl + End:** Move o cursor para o final do buffer de entrada.
 - **Ctrl + Left Arrow:** Move o cursor uma palavra para trás.
 - **Ctrl + Right Arrow:** Move o cursor uma palavra para frente.
 - **Ctrl + Backspace:** Apaga a palavra anterior ao cursor.
 - **Ctrl + Delete:** Apaga a palavra seguinte ao cursor.

- **Ctrl + Space** (No ISE/VS Code com extensão): Exibe a ajuda do IntelliSense (preenchimento automático).
- **Controle de Execução:**
 - **Enter:** Executa o comando atual.
 - **Shift + Enter:** Adiciona uma nova linha sem executar o comando (útil para comandos de várias linhas).
- **Histórico de Comandos:**
 - **Seta para Cima (↑):** Exibe o comando anterior no histórico.
 - **Seta para Baixo (↓):** Exibe o próximo comando no histórico.
 - **F7:** Exibe uma janela com o histórico de comandos, permitindo selecionar com as setas.
 - **Alt + F7:** Limpa o histórico de comandos da sessão atual.
 - **F8:** Pesquisa o histórico de comandos a partir da posição atual do cursor.
 - **F9:** Seleciona um comando do histórico pelo seu número.
- **Interface do PowerShell ISE (Integrated Scripting Environment):**
 - **F1:** Abre a ajuda para o cmdlet ou tópico atual.
 - **F5:** Executa o script completo no painel de script.
 - **F8:** Executa a seleção atual no painel de script.
 - **Ctrl + N:** Abre um novo painel de script.
 - **Ctrl + O:** Abre um arquivo existente.
 - **Ctrl + S:** Salva o arquivo atual.
 - **Ctrl + M:** Expande ou recolhe regiões de código (code folding).
 - **Ctrl + F:** Abre a caixa de diálogo "Localizar" no painel de script.
 - **Ctrl + H:** Abre a caixa de diálogo "Substituir" no painel de script.
 - **Ctrl + J:** Mostra snippets de código (modelos de código pré-definidos).
- **Atalho Universal do PowerShell:**
 -

Atalhos no terminal :

- **Ctrl + Alt + T :** Abre o terminal.
- **Ctrl + C :** Interrompe um comando em execução.

- **Ctrl + Z** : Coloca um comando em segundo plano.
- **Ctrl + A** : Move o cursor para o início da linha.
- **Ctrl + E** : Move o cursor para o final da linha.
- **Ctrl + R** : Busca por comandos anteriores.
- **Tab** : Completa automaticamente nomes de arquivos ou diretórios.
-

Comandos no Terminal :

Navegação e Listagem

- **ls** : Lista os arquivos e diretórios dentro do diretório atual.
 - **ls -l** : Mostra informações detalhadas sobre os arquivos, como permissões, tamanho e data de modificação.
- **cd** : Muda o diretório de trabalho atual.
 - **cd ..** : Volta um nível no diretório.
 - **cd /** : Vai para o diretório raiz.
 - **cd ~** : Vai para o diretório home.
- **pwd** : Mostra o caminho completo do diretório atual.

Busca

- **find** : buscar arquivos e diretórios em um sistema de arquivos, com base em critérios específicos. Ele permite filtrar por nome, tipo, tamanho, data de modificação, permissões e muito mais. Além disso, pode executar ações nos resultados encontrados (como excluir, mover ou processar arquivos).

- **1. Buscar por nome :**

- **-name** : Busca por nome exato (case-sensitive).

```
```bash
```

```
find /home -name "arquivo.txt" # Busca "arquivo.txt" em /home
```

- **-iname** : Busca por nome (case-insensitive).

```
```bash
```

```
find . -iname "*.PDF"                  # Encontra .pdf, .PDF, .Pdf, etc.
```

- **2. Buscar por tipo :**

- **-type** : Filtra por tipo de item :

f : Arquivos regulares.

d : Diretórios.

l : Links simbólicos.

```
```bash
```

```
find /var -type d # Lista todos os diretórios em /var
```

- **3. Buscar por tamanho :**

**-size** : Filtra por tamanho (ex : +10M para arquivos maiores que 10 MB).

```
```bash
```

```
find / -size +100M          # Arquivos maiores que 100 MB
```

- **4. Buscar por tempo de modificação :**

- **-mtime** : Dias desde a última modificação.

+7 : Modificados há mais de 7 dias.

- **-1** : Modificados há menos de 1 dia.

```
```bash
```

```
find /tmp -mtime +7 # Arquivos não modificados nos últimos 7 dias
```

- **5. Buscar por permissões :**

**-perm** : Filtra por permissões (ex : 644).

```
```bash
```

```
find ~ -perm 644            # Arquivos com permissões exatamente 644
```

- **6. Executar comandos nos resultados :**

-exec : Executa um comando para cada item encontrado.

```
```bash
```

```
find . -name "*.log" -exec rm {} \; # Exclui todos os .log
```

**{}** : Substitui o nome do arquivo encontrado.

**\;** : Indica o fim do comando.

- **7. Combinar critérios :**

**-and** (implícito), **-or**, **-not** :

```
```bash
```



```
find . -name "*.txt" -not -name "backup*"      # .txt que não começam com  
"backup"
```

- **8. Limitar profundidade da busca :**

-maxdepth : Define a profundidade máxima de subdiretórios.

```
```bash
```

```
find . -maxdepth 2 -name "*.js" # Busca até 2 níveis de subpastas
```

### Exemplos práticos :

- **Excluir arquivos temporários :**

```
```bash
```

```
find /tmp -type f -name "*.tmp" -exec rm {} \;
```

- **Listar arquivos modificados nas últimas 24 horas :**

```
```bash
```

```
find ~ -type f -mtime -1
```

- **Buscar arquivos de mais de 1 GB e listar detalhes :**

```
```bash
```

```
find / -size +1G -exec ls -lh {} \;
```

- **Encontrar diretórios vazios :**

```
```bash
```


```
find . -type d -empty
```

- **Alterar permissões de todos os scripts .sh :**

```
```bash
```

```
find . -name "*.sh" -exec chmod +x {} \;
```

Dica de segurança :

- Sempre teste o comando find sem -exec ou -delete primeiro para verificar os resultados.
- Use -print ou -print0 (com xargs -0) para lidar com nomes de arquivos complexos (ex : espaços).
- O find é essencial para automatizar tarefas de administração de sistemas e gerenciamento de arquivos! 

Criação e Remoção

- **mkdir** : Cria um novo diretório.
 - **mkdir documentos** : Cria um diretório chamado "documentos".
- **touch** : Cria um novo arquivo vazio.
 - **touch meu_arquivo.txt** : Cria um arquivo de texto chamado "meu_arquivo.txt".
- **rm** : Remove um arquivo. **Cuidado** : Use com cautela, pois a remoção é permanente.
 - **rm arquivo.txt** : Remove o arquivo "arquivo.txt".
 - **rm -fr nome_do_diretório** : Apaga o diretório e todo o seu conteúdo recursivamente. **Atenção** : Use este comando com cuidado, pois não há uma lixeira para recuperar arquivos apagados desta forma.
 - **-f** : Força a remoção, ignorando mensagens de proteção.
 - **-i** : Interage, pedindo confirmação antes de apagar cada arquivo.
- **rmdir** : Remove um diretório vazio.
 - **rmdir pasta_vazia** : Remove a pasta "pasta_vazia".

Cópia e Movimentação

- **cp** : Copia arquivos e diretórios.
 - **cp arquivo.txt documentos** : Copia o arquivo "arquivo.txt" para o diretório "documentos".
 - **cp -r diretorio1 diretorio2** : Copia recursivamente o diretório "diretorio1" para "diretorio2".
 - **cp *.txt pasta_destino/** : Copia todos os arquivos com extensão .txt para pasta_destino.
 - **cp -p arquivo1 arquivo2 diretorio_destino/** : A opção -p mantém atributos como data de modificação e permissões (metadados).
 - **cp -i arquivo1 arquivo2 diretorio_destino/** : A opção -i pede confirmação se um arquivo com o mesmo nome existir no destino.
- **mv** : Move ou renomeia arquivos e diretórios.
 - **mv arquivo.txt outro_nome.txt** : Renomeia o arquivo "arquivo.txt" para "outro_nome.txt".
 - **mv arquivo.txt documentos** : Move o arquivo "arquivo.txt" para o diretório "documentos".

Abrir arquivos de texto

Para visualizar ou editar arquivos de texto :

- **Visualizar** : Use o comando cat, less, ou more.

cat nome_do_arquivo.txt

less nome_do_arquivo.txt

more nome_do_arquivo.txt

- **Editar** : Use um editor de texto como nano, vim, ou gedit.

`nano nome_do_arquivo.txt`

`vim nome_do_arquivo.txt`

`gedit nome_do_arquivo.txt` # Abre uma interface gráfica (caso instalada)

Abrir qualquer arquivo com o programa padrão

O comando `xdg-open` pode ser usado para abrir qualquer arquivo com o aplicativo padrão associado :

- **`xdg-open nome_do_arquivo`**
 - Esse comando detecta automaticamente o tipo de arquivo e usa o aplicativo padrão para abrir.
- **`cat arquivo.txt`**
 - Este comando exibe o conteúdo de um arquivo de texto diretamente no terminal.
- **`less arquivo_grande.log`**
 - Similar ao `cat`, mas permite navegar pelo conteúdo do arquivo página por página.

Gerenciamento de Pacotes

- **`sudo apt update`** : Atualiza o índice de pacotes.
- **`sudo apt upgrade`** : Atualiza todos os pacotes instalados.
- **`sudo apt install nome_do_pacote`** : Instala um novo pacote.
 - `sudo apt install firefox` : Instala o navegador Firefox.

Outros Comandos Úteis

- **`man nome_do_comando`** : Acessa o manual do comando.
- **`clear`** : Limpa a tela do terminal.
- **`history`** : Mostra o histórico de comandos.
 - **`history -c`** : Limpa completamente o histórico de comandos.
 - **`history -d <número>`** : Apaga o comando com o número especificado.
 - **`history -a`** : Adiciona os comandos da sessão atual ao arquivo de histórico.

- **Ctrl+R** : Ativa a busca reversa. Comece a digitar parte de um comando e o terminal irá procurar por comandos anteriores que correspondam à sua entrada.
- **Alsamixer** : mostra opções de controle de saída de áudio (caixas de som, fones de ouvido, etc).

Exemplo Prático

Vamos criar uma estrutura de diretórios para organizar seus documentos :

1. `mkdir documentos`
2. `cd documentos`
3. `mkdir textos imagens`
4. `touch texto1.txt texto2.txt`
5. `cp imagem1.jpg imagem2.png imagens`

Com esses comandos, você criou um diretório chamado "documentos" com subdiretórios para "textos" e "imagens", e adicionou alguns arquivos de exemplo.

Dicas :

- **sudo** : Permite executar comandos com privilégios de administrador.
- **Tab** : Completa automaticamente nomes de arquivos e comandos.
- **?** (ponto de interrogação) : Mostra opções de um comando.
- **^C** : Interrompe um comando em execução.

Recursos Adicionais :

- **Tutoriais online** : Existem diversos tutoriais em vídeo e texto sobre o terminal Linux.
- **Comunidades** : Participar de fóruns e comunidades Linux pode te ajudar a tirar dúvidas e aprender com outros usuários.

Com a prática, você se tornará cada vez mais familiar com o terminal e poderá realizar tarefas complexas de forma eficiente. **Lembre-se** : o terminal é uma ferramenta poderosa, mas também exige cuidado. Utilize-o com sabedoria para evitar danos ao seu sistema.

Windows

Sugestão : Criaremos duas listas separadas, uma para o **Prompt de Comando** e outra para o **PowerShell**, destacando as diferenças e semelhanças entre eles.

(CMD)

Lista de Comandos para Terminal do Windows

Prompt de Comando (cmd)

Atalhos :

- **Ctrl+C** : Interrompe a execução de um comando.
- **Ctrl+Z** : Encerra o programa atual.
- **↑ ↓** : Navega pelo histórico de comandos.
- **Tab** : Completa automaticamente nomes de arquivos e comandos.

Comandos :

- **dir** : Lista os arquivos e subdiretórios de um diretório.
- **cd** : Altera o diretório atual.
- **cls** : Limpa a tela.
- **help** : Exibe ajuda sobre um comando específico.
- **exit** : Encerra o Prompt de Comando.

Navegação e Listagem :

- **cd ..** : Volta um nível no diretório.
- **cd** : Vai para a raiz do disco.
- **dir /w** : Lista os arquivos em formato amplo.
- **dir /p** : Lista os arquivos em páginas.

Criação e Remoção :

- **mkdir** : Cria um novo diretório.
- **rmdir** : Remove um diretório vazio.
- **del** : Apaga um arquivo.

Cópia e Movimentação :

- **copy** : Copia um ou mais arquivos.
- **move** : Move um ou mais arquivos.

Abrir arquivos de texto :

- **type** : Exibe o conteúdo de um arquivo de texto.
- **more** : Exibe o conteúdo de um arquivo de texto página por página.

Abrir qualquer arquivo com o programa padrão :

- **start** : Abre um arquivo com o programa associado.

Gerenciamento de Pacotes :

- **Não disponível** : O Prompt de Comando não possui um gerenciador de pacotes integrado.

Outros Comandos Úteis :

- **ipconfig** : Exibe informações de configuração da rede.
- **ping** : Envia pacotes ICMP para um host.
- **tracert** : Traça a rota de um pacote IP.
- **tasklist** : Lista os processos em execução.
- **shutdown** : Desliga ou reinicia o computador.

Exemplo Prático :

```
C :\> cd Documents
```

```
C :\Documents> dir
```

```
Volume in drive C is OS
```

```
Volume Serial Number is 1234-5678
```

```
Directory of C :\Documents
```

```
05/01/2023  10 :24 AM    <DIR>        Subpasta
```

```
22/02/2023  03 :15 PM                123 byte arquivo.txt
```

```
1 file(s)      123 bytes
```

1 dir(s) 21.484.174.336 bytes free

PowerShell

Atalhos :

- Semelhantes ao Prompt de Comando.

Comandos :

- **Get-ChildItem** : Lista os itens em um diretório (equivalente a dir).
- **Set-Location** : Altera o diretório atual (equivalente a cd).
- **Clear-Host** : Limpa a tela (equivalente a cls).
- **Get-Help** : Exibe ajuda sobre um comando.
- **Exit** : Encerra o PowerShell.

Outras funcionalidades :

- **Pipelines** : Conectam o resultado de um comando à entrada de outro.
- **Cmdlets** : Comandos pré-definidos para diversas tarefas.
- **Scripts** : Arquivos com sequências de comandos.

Exemplo Prático :

PowerShell :

```
PS C :\> Get-ChildItem Documents
```

```
Directory : C :\Documents
```

Mode	LastWriteTime	Length	Name
d----	05/01/2023 10 :24 AM		Subpasta
-a----	22/02/2023 03 :15 PM	123	arquivo.txt

Use o código.

Considerações Finais

- **PowerShell** é mais poderoso e flexível que o Prompt de Comando, oferecendo uma vasta gama de funcionalidades para administração de sistemas.
- **Prompt de Comando** é mais simples e adequado para tarefas básicas.
- Para tarefas mais complexas e automatização, o **PowerShell** é a melhor escolha.

Observação : Esta lista é apenas um resumo dos comandos mais comuns. Ambos os ambientes possuem muitos outros comandos e parâmetros que podem ser explorados.