



**SENAI**

A solid red vertical bar on the left side of the slide.

# WINDOWS CLI

## (Command Line Interface)



# CLI (Command Line Interface)

O **CLI** (Command Line Interface, ou Interface de linha de comando) é tudo que **utilizamos e interagimos através de comandos** em um sistema operacional.

*CLI é uma ferramenta muito utilizada por administradores de sistemas operacionais pois além de ser uma ferramenta muito poderosa, ela permite automatização de tarefas através de **scripts** programáveis, além de ser mais rápida que a interface gráfica, já que não precisa ficar renderizando diversos tipos de informação em tela.*

*Tudo que fazemos no CLI podemos fazer no GUI, mas o contrário não é verdade, pois alguns programas só existem em linha de comando, tornando impossível de realizar a mesma ação na interface gráfica.*



# CLI (Command Line Interface)

*O Windows atualmente possui duas ferramentas de CLI nativas no sistema operacional:*

## **Prompt de Comando**

*Ferramenta de linha de comando clássica do sistema operacional, onde executamos comandos DOS (os primeiros comandos criados junto com o MS-DOS e Windows)*

## **Powershell**

*Ferramenta de linha de comando poderosa que trás diversas ferramentas para administradores de rede, os comandos executados se chamando cmdlets (ou commandlets).*

***Powershell executa comandos do Prompt de Comandos, mas não o contrário.***



Ícone do Prompt de  
Comando



Ícone do  
Powershell



# CLI (Command Line Interface)

*Antes de começarmos a vermos os comandos, vamos entender algumas particularidades da linha de comando*

Expressão	Descrição
/ ou \	Indica a separação de diretórios, quando usado sozinho, indica a raiz do sistema operacional, no caso do Windows o C:\
.	Indica o diretório atual onde o CLI está aberto.
..	Indica o diretório acima de onde o CLI está aberto.
../..	Indica dois diretórios acima de onde o CLI está aberto
< >	Notação de uma variável, onde o usuário precisa fornecer um valor para o comando, sem utilizar os caracteres < e >.



# Navegação entre diretórios

Uma das funções mais comuns que executamos em um sistema operacional, e as vezes nem percebemos, é a navegação entre diretórios.

Quando fazemos esse processo através da GUI é automático e literalmente leva um clique, porém também é possível usamos a CLI para listarmos e navegarmos entre diretórios.

## Exemplos de uso:

```
cd Desktop
```

```
cd ..
```

```
cd /
```

Comando	Descrição
dir	Lista o conteúdo do diretório atual
Get-ChildItem	Lista o conteúdo do diretório atual (Powershell)
cd <diretório>	Navega para o diretório especificado
Set-Location <diretório>	Navega para o diretório especificado (Powershell)



**cd** é a abreviação de **change directory** ou *mudar de diretório em português*.



# Criação de diretórios e arquivos

Criar diretórios (pastas) é outra tarefa diária de um usuário de computador, que também pode ser feita através de linha de comando.

Para criarmos um diretório utilizamos o comando **mkdir** ou apenas **md** informando o nome do diretório que queremos criar.

## Exemplos de uso:

```
mkdir teste
```

```
md "Minha pasta"
```

Comando	Descrição
mkdir <diretório>   md <diretório>	Cria um diretório com o nome informado

 **mkdir** ou **md** é a abreviação de **make directory** ou criar diretório em português.

 A notação <> indica um valor que precisa ser informado pelo usuário, mas sem utilizar os caracteres <>



# Criação de diretórios e arquivos

Também é possível criarmos arquivos usando a CLI, usando o **type nul > arquivo.txt** no prompt de comando ou **New-Item -ItemType File -Path "arquivo.txt"** no Powershell.

**Exemplos de uso:**

```
type nul > arquivo.html
New-Item -ItemType File -Path "script.ps1"
```

Comando	Descrição
type nul > arquivo.txt	Cria um novo arquivo (necessário especificar nome e extensão do arquivo)
New-Item -ItemType File -Path "arquivo.txt"	Cria um novo arquivo (necessário especificar nome e extensão do arquivo)





# Exclusão de diretórios e arquivos

Para excluirmos um diretório ou um arquivo, podemos utilizar os comandos, **del**, **rd** e **Remove-Item**.

## Exemplos de uso:

`del arquivo.txt`

`rd temp`

`Remove-Item temp`

Comando	Descrição
<code>del &lt;arquivo&gt;</code>	Remove um arquivo especificado. (Não remove diretórios)
<code>rd &lt;diretório&gt;</code>	Remove um diretório especificado.
<code>Remove-Item &lt;arquivo   diretório&gt;</code>	Remove um arquivo ou um diretório especificado. (Somente Powershell)



A notação **<>** indica um valor que precisa ser informado pelo usuário, mas sem utilizar os caracteres **<>**



# Renomear diretórios e arquivos

O comando *ren* ou *Rename-Item* é utilizado para renomear arquivos ou diretórios já existentes.

## Exemplos de uso:

```
ren arquivo.txt alunos.txt
```

```
Rename-Item "Nova pasta" Aula
```

Comando	Descrição
<code>ren &lt;arquivo   diretório&gt; &lt;novo-nome&gt;</code>	Renomeia um arquivo ou diretório especificado. (São necessários 2 parâmetros, o nome do arquivo/diretório original e o novo nome)
<code>Rename-Item &lt;arquivo   diretório&gt; &lt;novo-nome&gt;</code>	Renomeia um arquivo ou diretório especificado. (São necessários 2 parâmetros, o nome do arquivo/diretório original e o novo nome, funciona somente no Powershell)



A notação `<>` indica um valor que precisa ser informado pelo usuário, mas sem utilizar os caracteres `<>`



# Mover diretórios e arquivos

Mover um arquivo ou diretório significa mudar o lugar onde aquele item está armazenado para um novo local, para isso são utilizados os comandos `move`, `mv` ou `Move-Item`.  
O destino sempre é um diretório.

## Exemplos de uso:

```
mv aula.txt Aulas
```

```
Move-Item index.html public
```

Comando	Descrição
<code>move &lt;arquivo   diretório&gt; &lt;novo-local&gt;</code>	Move um arquivo ou diretório para o destino especificado. (São necessários 2 parâmetros, o nome do arquivo/diretório e o novo local)
<code>mv &lt;arquivo   diretório&gt; &lt;novo-local&gt;</code>	Move um arquivo ou diretório para o destino especificado. (São necessários 2 parâmetros, o nome do arquivo/diretório e o novo local)
<code>Move-Item &lt;arquivo   diretório&gt; &lt;novo-local&gt;</code>	Move um arquivo ou diretório para o destino especificado. (São necessários 2 parâmetros, o nome do arquivo/diretório e o novo local, funciona somente no Powershell)



A notação `<>` indica um valor que precisa ser informado pelo usuário, mas sem utilizar os caracteres `<>`

# Cópia de diretórios e arquivos

Para copiarmos um arquivo ou diretório para um novo local, utilizamos os comandos `copy`, `cp` ou `Copy-Item`. A cópia, como o próprio nome já diz, cria um novo arquivo como uma cópia do original, resultando em dois arquivos.

## Exemplos de uso:

`copy arquivo.txt "Arquivo cópia.txt"`

`Copy-Item Aula C:/`

Comando	Descrição
<code>copy &lt;arquivo   diretório&gt; &lt;novo-arquivo   novo-diretório&gt;</code>	Cria uma cópia do arquivo ou diretório especificado. (São necessários 2 parâmetros, o nome do arquivo/diretório e o nome da cópia)
<code>cp &lt;arquivo   diretório&gt; &lt;novo-arquivo   novo-diretório&gt;</code>	Cria uma cópia do arquivo ou diretório especificado. (São necessários 2 parâmetros, o nome do arquivo/diretório e o nome da cópia)
<code>Copy-Item &lt;arquivo   diretório&gt; &lt;novo-arquivo   novo-diretório&gt;</code>	Cria uma cópia do arquivo ou diretório especificado. (São necessários 2 parâmetros, o nome do arquivo/diretório e o nome da cópia, funciona somente no Powershell)



A notação `<>` indica um valor que precisa ser informado pelo usuário, mas sem utilizar os caracteres `<>`

# Edição de arquivos

*As versões mais recentes do Windows não possuem um comando para editar um arquivo diretamente via CLI, porém podemos chamar um editor de arquivo para essa finalidade, para isso, utilizamos o comando notepad. Esse comando só pode editar arquivos baseados em texto, como o arquivos txt, html, py, js, entre outros.*

## **Exemplo de uso:**

`notepad arquivo.txt`

Comando	Descrição
notepad <arquivo>	Invoca o programa bloco de notas para editar um arquivo baseado em texto.



É possível obtermos mais informações sobre o comando como sua sintaxe e os parâmetros que ele aceita na execução utilizando o comando de ajuda. O comando de ajuda é diferente para o Prompt de Comando e o Powershell.

**Exemplo de uso:**

```
rmmdir /?  
Get-Help Get-ChildItem
```

Comando	Descrição
comando /?	Obtém mais informações sobre o comando especificado no <b>Prompt de Comando</b> . Troque a palavra <i>comando</i> pelo comando.
Get-Help comando	Obtém mais informações sobre o comando especificado no <b>Powershell</b> . Troque a palavra <i>comando</i> pelo comando.



## Prática

*Criando e gerenciando um projeto usando CLI.*

*Vamos criar um projeto de software bem simples, sem nos preocuparmos com código, apenas para praticar os conceitos vistos nessa aula, para isso iremos utilizar todos os comandos que visitamos durante este material.*

<https://prof-renato.notion.site/CLI-WINDOWS-PR-TICA-1d1d15a33297806a93d2ea6fcbde57a3?pvs=4>



## Referências

- <https://devblogs.microsoft.com/scripting/table-of-basic-powershell-commands/>
- <https://zup.com.br/blog/cli-interface-de-linha-de-comando>
- [https://www.alura.com.br/artigos/cli-interface-linha-comandos?srsId=AfmBOoqPU17qSEr3kWqRCDJ7YLnml\\_gR7TqFHA3GjeG5FKAMJsp56Y-\\_](https://www.alura.com.br/artigos/cli-interface-linha-comandos?srsId=AfmBOoqPU17qSEr3kWqRCDJ7YLnml_gR7TqFHA3GjeG5FKAMJsp56Y-_)
- <https://blog.engdb.com.br/o-que-e-cli/>
- <https://learn.microsoft.com/pt-br/powershell/scripting/overview?view=powershell-7.5>
- <https://tecnoblog.net/responde/o-que-e-o-powershell-do-windows/>





