

## Atividade 1 - Aula 3 - Comandos Linux

Objetivo: Dando continuidade na aula anterior, vamos aprender alguns comandos novos.

Comandos que serão utilizados neste exercício:

Comando	Descrição
<b>find</b>	pesquisa arquivos em uma hierarquia de diretórios.
<b>mv</b>	move (ou renomeia) arquivos/diretórios.  Se o caminho-destino for o mesmo local, irá renomear, se for diferente, irá mover.
<b>grep</b>	procura padrões em um arquivo.
<b>sed</b>	Editor de textos não interativo.
<b>cd</b>	permite ao usuário mudar o diretório de trabalho.
<b>ls</b>	lista o conteúdo de um diretório.
<b>diff</b>	compara dois arquivos linha a linha.
<b>cat</b>	envia o conteúdo de um ou mais arquivos para a saída padrão ou para um outro arquivo.
<b>tree</b>	lista o conteúdo de um diretório usando o formato de árvore. Ele tem a mesma função do comando ls. A diferença consiste na maneira como as informações são exibidas.

Essa página é bem interessante porque explica o significado dos comandos e contém exemplos: <https://guialinux.uniriotec.br/nome-do-comando>

=====

Por Exemplo o comando **find**: <https://guialinux.uniriotec.br/find>

=====

A Figura abaixo é uma ilustração sobre a estrutura de pastas que serão criadas.

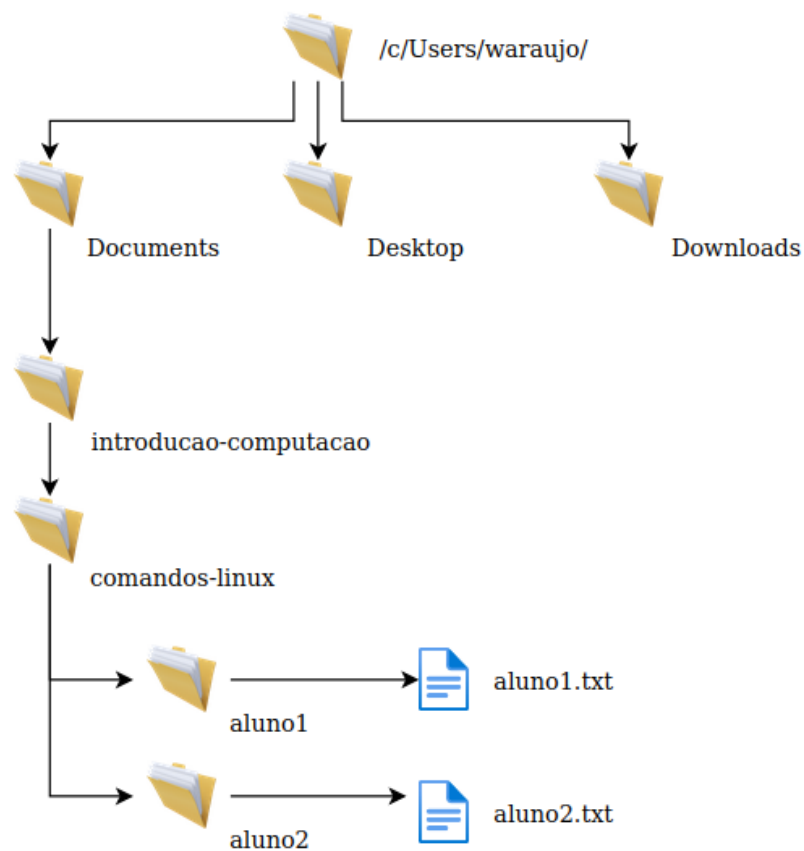


Figura 1 - Exercício sobre Diretório e Arquivos

Lembre-se: é a partir do símbolo `$` que iremos digitar os comandos.

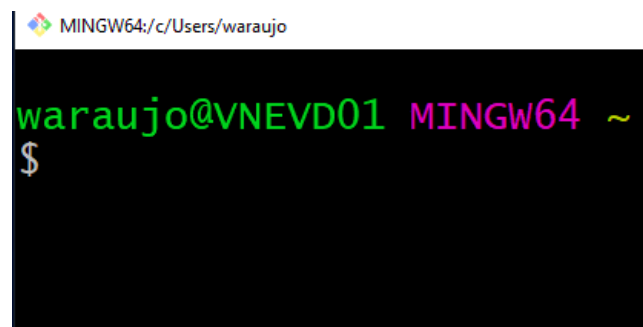


Figura 2 - Terminal Git Bash

## Passo-a-Passo para realização do exercício.

- 1) Imagina que você não lembra onde está o arquivo **aluno1.txt**, o comando **find** irá nos ajudar.

Sabemos que o nome do arquivo contém a palavra **aluno**. Então eu tenho que dizer para o **find**, "por favor, procure para mim o arquivo que comece com **aluno\*** seguido de qualquer coisa".

Sintaxe: **find** [caminho do diretório] [expressão]

```
$ find Documents -name "aluno1.txt"
$ find Documents -name "aluno*"
$ find Documents -name "Aluno*"
```

# quando você insere o **.** (ponto), você está pedindo para procurar a partir do diretório atual incluindo o subdiretórios.

```
$ find . -name "aluno*"
```

# Agora procure sem se importar com o maiúsculas e minúsculas (-iname)

```
$ find Documents -iname "Aluno*"
$ find . -iname "Aluno*"
```

- 
- 2) Vamos alterar o nome da pasta **comandos-linux** da Figura 1.

```
# Vamos para o diretório introducao-computacao
$ cd Documents/introducao-computacao/

# agora iremos renomear a pasta para backend-python
$ mv comandos-linux backend-python

# verifique se a pasta foi alterada para backend-python
$ ls
```

- 
- 3) Vamos verificar o conteúdo do arquivo **aluno2.txt**

```
# Vamos para o diretório aluno2
$ cd backend-python/aluno2/

# ver o conteúdo do arquivo. O parâmetro -n, enumera as linhas do arquivo.
$ cat aluno2.txt -n
```

```
$ cat aluno2.txt -n
1
2  Vida Nova Escola de Tecnologia
3  Curso Desenvolvedor Backend Python
4  Sou o aluno1
5  Sou o aluno2
```

Seu arquivo está conforme a Figura acima ?? Se não estiver, avise o professor.

- 
- 4) O arquivo que acabamos de ver foi do aluno2.txt, repare que não deveríamos ter a frase "Sou o aluno1". Vamos remover essa frase do arquivo ?

```
# apagando a linha 4 ("Sou o aluno1")
$ sed -i 4d aluno2.txt
```

```
$ cat aluno2.txt
```

```
$ cat aluno2.txt

Vida Nova Escola de Tecnologia
Curso Desenvolvedor Backend Python
Sou o aluno2
```

- 
- 5) Podemos pesquisar alguma palavra ou frase em um arquivo utilizando o comando `grep`.

Sintaxe: `grep [palavra para pesquisar] [nome do arquivo]`

```
$ grep "python" aluno2.txt
$ grep "Python" aluno2.txt
$ grep "python" aluno2.txt -i
```

- 
- 6) Nosso diretório atual é:  
[/Documents/introducao-computacao/backend-python/aluno2](#)

Agora vamos para o diretório backend-python.  
\$ `cd ..`

- 
- 7) O comando `diff` verifica qual a diferença de conteúdo que tem entre dois arquivos.

Sintaxe: `diff [arquivo1] [arquivo2]`

```
$ diff aluno1/aluno1.txt aluno2/aluno2.txt.
```

---

8) Nosso diretório atual é:

[/Documents/introducao-computacao/backend-python/](#)

Agora vamos para o diretório Documents.

```
$ cd ../..
```

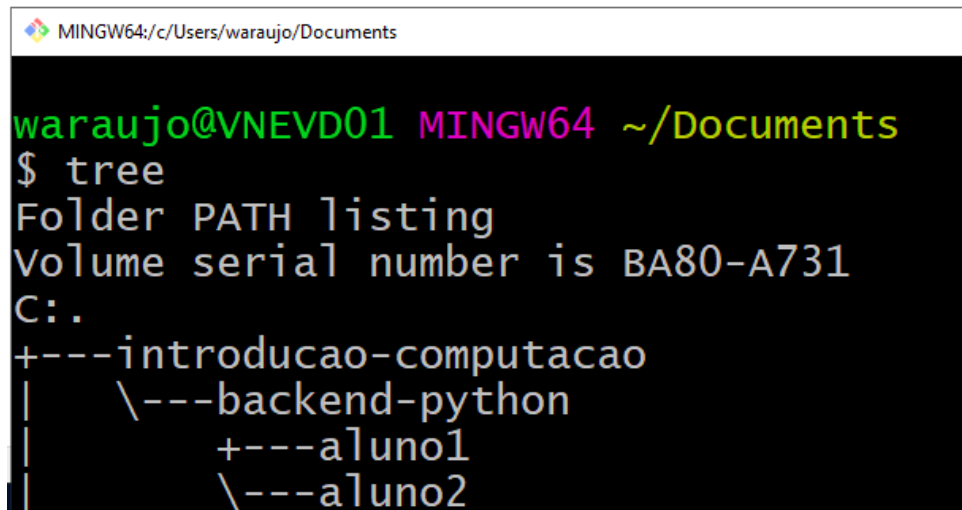
---

9) Para finalizar vamos ver como ficou nossa estrutura de diretórios ?

Vamos criar um [Alias](#) (apelido para o comando):

```
$ alias tree='cmd //c tree //a'
```

```
$ tree
```



```
MINGW64: c:/Users/waraujo/Documents

waraujo@VNEVD01 MINGW64 ~/Documents
$ tree
Folder PATH listing
Volume serial number is BA80-A731
C:..
+---introducao-computacao
|   \---backend-python
|       +---aluno1
|       \---aluno2
```