



Roteiro de bordo linux / Camandos basicos

📅 Data	@10/02/2022
▼ Situation	Concluída
▼ Disciplina	Sistemas Operacionais
Σ Property	✓
📅 Data de entrega	@10/02/2022

1. pwd

O comando **pwd** e usado para descobrir o caminho do diretório de trabalho atual (pasta) em que você está. O comando retornará um caminho absoluto (completo), que é basicamente um caminho de todos os diretórios que começa com uma barra para a frente (/). Um exemplo de um caminho absoluto é **/home/username**.

```
dcdevs in ~  
λ pwd  
/home/dcdevs  
dcdevs in ~  
λ
```

2. cd

Para navegar pelos arquivos e diretórios Linux, e usado o comando **cd**. Requer o caminho completo ou o nome do diretório, dependendo do diretório de trabalho atual em que você está.

Se estivermos dentro de **/home/username/Documents** e queremos ir para fotos **Photos**, um subdiretorio de **Documents**. Para isso basta digitar: `cd Photos`

Outro cenário é se você quiser mudar para um diretório completamente novo, por exemplo, **/home/username/Movies**. Nesse caso se utiliza **cd** seguido do caminho absoluto: `cd /home/username/Movies`.

Existem atalhos para facilitar a navegação:

- **cd ..** (com dois pontos) para mover para um diretório antes
- **cd** para ir direto para o diretório **/home**
- **cd-** (com um hífen) para passar paara o diretorio anterior

```
dcdevs in ~
λ ls
Desktop      gnome-terminal  Music          Pictures      Templates
Documents    Imagens         photorec.se2   Public        Videos
Downloads    Modelos         photorec.ses   snap         'VirtualBox VMs'
dcdevs in ~
λ cd Desktop
dcdevs in ~/Desktop
λ pwd
/home/dcdevs/Desktop
dcdevs in ~/Desktop
λ cd www/portfolio
dcdevs in portfolio on 🔓 master [?]
λ cd -
~/Desktop
dcdevs in ~/Desktop
λ cd .
dcdevs in ~/Desktop
λ cd ../
dcdevs in ~
λ pwd
/home/dcdevs
dcdevs in ~
λ
```

3. ls

O comando **ls** e usado para visualizar o conteudo de um diretori. Por padrão esse comando exibira conteudos do seu diretorio de trabalho.

Se quisermos verificar o conteudo de um diretório precisamos digitar **ls** e o caminho do diretorio ex: `ls /home/username/Documents` para visualizar o conteudo de **/Documents**.

Existem variações que podemos utilizar como [option] no comando **ls**:

- **ls -R** lista todos os arquivos do diretórios e subdiretórios.
- **ls -a** mostra os arquivos escondidos

- **ls -al** listará os arquivos e diretórios com informações detalhadas como as permissões, tamanho, proprietário, etc.

```
dcdevs in ~
A ls
Desktop      gnome-terminal  Music      Pictures    Templates
Documents    Imagens         photorec.se2 Public       Videos
Downloads    Modelos         photorec.ses snap         'VirtualBox VMs'
dcdevs in ~
ls Desktop/www/Rocket.Q/public
images scripts styles
dcdevs in ~
A ls -R Desktop/www/Rocket.Q/public
Desktop/www/Rocket.Q/public:
images scripts styles
Desktop/www/Rocket.Q/public/images:
emptyState.png icon_check.svg icon_enter.svg icon_trash.svg icon_users.svg user.svg
home-bg-img.svg icon_copy.svg icon_lock-black.svg icon_usersF.svg logo.svg
Desktop/www/Rocket.Q/public/scripts:
main.js modal.js
Desktop/www/Rocket.Q/public/styles:
global.css homeEndCreatePass.css modal.css room.css
dcdevs in ~
A
```

```
dcdevs in ~
A ls -a Desktop/www/Rocket.Q
. . . .git node_modules package.json package-lock.json public src temp.txt
dcdevs in ~
A ls -al Desktop/www/Rocket.Q
total 156
drwxr-xr-x 6 dcdevs dcdevs 4096 Feb 16 09:04 .
drwxr-xr-x 4 dcdevs dcdevs 4096 Feb 16 09:04 ..
drwxr-xr-x 8 dcdevs dcdevs 4096 Feb 16 23:31 .git
drwxr-xr-x 162 dcdevs dcdevs 4096 Feb 16 09:04 node_modules
-rw-r--r-- 1 dcdevs dcdevs 573 Feb 16 09:04 package.json
-rw-r--r-- 1 dcdevs dcdevs 126928 Feb 16 09:04 package-lock.json
drwxr-xr-x 5 dcdevs dcdevs 4096 Feb 16 09:04 public
drwxr-xr-x 4 dcdevs dcdevs 4096 Feb 16 09:04 src
-rw-r--r-- 1 dcdevs dcdevs 137 Feb 16 09:04 temp.txt
dcdevs in ~
A
```

4. cat

cat (abreviação para concatenar) é um dos comandos mais usados no Linux. Ele é usado para listar o conteúdo de um arquivo no output padrão (stdout). Para executar este comando, digite **cat** seguido pelo nome do arquivo e sua extensão. Por exemplo: **cat file.txt**.

Aqui estão outras maneiras de usar o comando **cat**:

- **cat > nome de arquivo** cria um novo arquivo
- **cat filename1 filename2>filename3** junta-se a dois arquivos (1 e 2) e armazena a saída deles em um novo arquivo (3)
- Converter um arquivo para uso de caixa superior ou inferior, **cat filename | tr a-z A-Z >output.txt**

```
dcdevs in Rocket.Q on p main [X]
A cat package.json
{
  "name": "rocket.q",
  "version": "1.0.0",
  "description": "",
  "main": "src/server.js",
  "scripts": {
    "start": "nodemon ."
  },
  "repository": {
    "type": "git",
    "url": "git+https://github.com/Diego-Devs/Rocket.Q.git"
  },
  "keywords": [],
  "author": "Diego Silva",
  "license": "ISC",
  "bugs": {
    "url": "https://github.com/Diego-Devs/Rocket.Q/issues"
  },
  "homepage": "https://github.com/Diego-Devs/Rocket.Q#readme",
  "dependencies": {
    "ejs": "^3.1.6",
    "express": "^4.17.2"
  },
  "devDependencies": {
    "nodemon": "^2.0.15"
  }
}
dcdevs in Rocket.Q on p main [X]
A cat package.json > package-copy.json
dcdevs in Rocket.Q on p main [X]
```

```

}
dcdevs in Rocket.Q on ➤ main [X?]
A cat package.json > package-copy.json
dcdevs in Rocket.Q on ➤ main [X?]
A cat package-copy.json
{
  "name": "rocket.q",
  "version": "1.0.0",
  "description": "",
  "main": "src/server.js",
  "scripts": {
    "start": "nodemon ."
  },
  "repository": {
    "type": "git",
    "url": "git+https://github.com/Diego-DevsS/Rocket.Q.git"
  },
  "keywords": [],
  "author": "Diego Silva",
  "license": "ISC",
  "bugs": {
    "url": "https://github.com/Diego-DevsS/Rocket.Q/issues"
  },
  "homepage": "https://github.com/Diego-DevsS/Rocket.Q#readme",
  "dependencies": {
    "ejs": "^3.1.6",
    "express": "^4.17.2"
  },
  "devDependencies": {
    "nodemon": "^2.0.15"
  }
}
dcdevs in Rocket.Q on ➤ main [X?]
A

```

```

dcdevs in Rocket.Q on ➤ main [X?]
A cat > arquivo.txt
Podemos inserir aqui o conteúdo do arquivo
Apertando enter para cada
linha digitada
E ^C para sair^C
dcdevs in Rocket.Q on ➤ main [X?] took 1m 5s
A ls
arquivo.txt  node_modules  package-copy.json  package.json  package-lock.json  public  src
dcdevs in Rocket.Q on ➤ main [X?]
A cat package-copy.json package.json > arquivo.txt
dcdevs in Rocket.Q on ➤ main [X?]
A cat arquivo.txt

```

```

dcdevs in Rocket.Q on ➤ main [X?]
cat arquivo.txt
{
  "name": "rocket.q",
  "version": "1.0.0",
  "description": "",
  "main": "src/server.js",
  "scripts": {
    "start": "nodemon ."
  },
  "repository": {
    "type": "git",
    "url": "git+https://github.com/Diego-DevsS/Rocket.Q.git"
  },
  "keywords": [],
  "author": "Diego Silva",
  "license": "ISC",
  "bugs": {
    "url": "https://github.com/Diego-DevsS/Rocket.Q/issues"
  },
  "homepage": "https://github.com/Diego-DevsS/Rocket.Q#readme",
  "dependencies": {
    "ejs": "^3.1.6",
    "express": "^4.17.2"
  },
  "devDependencies": {
    "nodemon": "^2.0.15"
  }
}
}
dcdevs in Rocket.Q on ➤ main [X?]
A

```

5. cp

O comando **cp** é usado para copiar arquivos do diretório atual para um diretório diferente. Por exemplo, o comando **cp arquivo.txt arquivo-copy.txt** criaria uma cópia do **arquivo.txt** (no seu diretório atual)

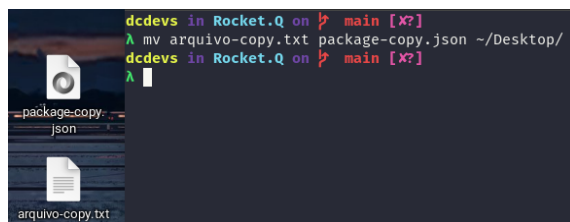
```

dcdevs in Rocket.Q on  main [x?]
λ cat > arquivo.txt
Teste
^C
dcdevs in Rocket.Q on  main [x?] took 6s
λ cat arquivo.txt
Teste
dcdevs in Rocket.Q on  main [x?]
λ cp arquivo.txt arquivo-copy.txt
dcdevs in Rocket.Q on  main [x?]
λ cat arquivo-copy.txt
Teste
dcdevs in Rocket.Q on  main [x?]
λ

```

6. mv

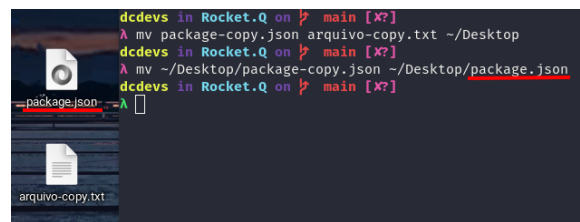
O principal uso do comando **mv** é mover arquivos, embora também possa ser usado para renomear arquivos. Os argumentos em **mv** são semelhantes ao comando **cp**. Você precisa digitar **mv**, o nome do arquivo e o diretório do destino. Por exemplo: arquivo **mv.txt /home/username/Documents**. Para renomear arquivos, o comando Linux é **mv oldname.ext newname.ext**



```

dcdevs in Rocket.Q on  main [x?]
λ mv arquivo-copy.txt package-copy.json ~/Desktop/
dcdevs in Rocket.Q on  main [x?]
λ

```



```

dcdevs in Rocket.Q on  main [x?]
λ mv package-copy.json arquivo-copy.txt ~/Desktop
dcdevs in Rocket.Q on  main [x?]
λ mv ~/Desktop/package-copy.json ~/Desktop/package.json
dcdevs in Rocket.Q on  main [x?]
λ

```

7. mkdir.

Use o comando **mkdir** para fazer um novo diretório — se você digitar **mkdir Music** ele criará um diretório chamado **Music**.

Há comandos extras **mkdir** também:

- Para gerar um novo diretório dentro de outro diretório, use este comando básico do Linux **mkdir Music/Newfile**
- Use se a opção **p** (parents) para para criar diretorio entre diretorios existentes. Por exemplo, **mkdir -p Music/2020/Newfile** criará o novo diretório "2020".

- Podemos criar estrutura de diretórios, utilizando entre {}(chave)
- Podemos utilizar das chaves também para criar varios diretorios ex: **mkdir -p teste{1..5}**, percebemos a diferença, para criar diretorio utiliza se o / antes das chaves e passado dentro dos chaves a junção de nomes dos diretórios separadas por virgula, ou numeros seguindo o modelo descrito.

```
dcdevs in ~/Desktop
λ mkdir media
dcdevs in ~/Desktop
λ ls
media
dcdevs in ~/Desktop
λ mkdir media/Music
dcdevs in ~/Desktop
λ tree media
media
├── Music
1 directory, 0 files
dcdevs in ~/Desktop
λ rm -r media
dcdevs in ~/Desktop
λ ls
dcdevs in ~/Desktop
λ mkdir media/Music
mkdir: cannot create directory 'media/Music': No such file or directory
dcdevs in ~/Desktop
λ mkdir -p media/Music
dcdevs in ~/Desktop
λ ls
media
dcdevs in ~/Desktop
λ tree media
media
├── Music
1 directory, 0 files
dcdevs in ~/Desktop
```

```
dcdevs in ~/Desktop
λ mkdir -p ./media/Music/{punk,MPB,Altores}
dcdevs in ~/Desktop
λ tree media
media
├── Music
│   ├── MPB
│   └── Altores
└── punk
4 directories, 0 files
dcdevs in ~/Desktop
λ
```

```
dcdevs in ~/Desktop
λ mkdir -p ./media/{document,image,Music/{MPB,Altores,puk}}
dcdevs in ~/Desktop
λ tree
media
├── document
├── image
└── Music
    ├── MPB
    └── Altores
        └── puk
7 directories, 0 files
dcdevs in ~/Desktop
λ rm -rf media/*
zsh: sure you want to delete all 3 files in /home/dcdevs/Desktop/media [yn]? y
dcdevs in ~/Desktop
λ mkdir -p teste{1..5}
dcdevs in ~/Desktop
λ ls
media teste1 teste2 teste3 teste4 teste5
dcdevs in ~/Desktop
```

8. rmdir

Se você precisar excluir um diretório, use o comando **rmdir** . No entanto, o rmdir só permite excluir diretórios vazios.

```
dcdevs in ~/Desktop/media
λ tree
.
├── diretorio1
│   ├── texto1.txt
│   └── texto2.txt
├── diretorio2
│   └── texto3.txt
├── diretorio3
│   └── subdiretorio
│       ├── arquivo.txt
│       └── su-sub-diretorio
└── diretorio4

6 directories, 4 files
dcdevs in ~/Desktop/media
λ rmdir diretorio3
rmdir: failed to remove 'diretorio3': Directory not empty
dcdevs in ~/Desktop/media
λ rmdir diretorio4
dcdevs in ~/Desktop/media
λ ls
diretorio1 diretorio2 diretorio3
dcdevs in ~/Desktop/media
λ
```

9. rm

O comando **rm** é usado para excluir diretórios e o conteúdo dentro deles. Se você quiser apenas excluir o diretório — como uma alternativa ao **rmdir** — use **rm -r**.

Nota: Tenha muito cuidado com este comando e verifique em qual diretório você está. Isso vai apagar tudo e não há desfaça.

- **rm -f <arquivo> | -f --force** ignore arquivos e argumentos inexistentes, nunca solicite
- **rm -i <arquivo> | -i** prompt antes de cada remoção
- **rm -r <diretorio> | -r, -R, --recursive** remove diretórios e seus conteúdos recursivamente

```
dcdevs in ~/Desktop/media
λ tree
.
├── arquivo1.txt
├── diretorio1
│   ├── texto1.txt
│   └── texto2.txt
├── diretorio2
│   └── texto3.txt
├── diretorio3
│   └── subdiretorio
│       ├── arquivo.txt
│       └── su-sub-diretorio
└── diretorio4

5 directories, 5 files
dcdevs in ~/Desktop/media
λ rm arquivo1.txt
dcdevs in ~/Desktop/media
λ rm diretorio1/texto2.txt
dcdevs in ~/Desktop/media
λ rm -i diretorio1/texto1.txt
rm: remove regular empty file 'diretorio1/texto1.txt'? yes
dcdevs in ~/Desktop/media took 10s
λ rmdir diretorio2
rmdir: failed to remove 'diretorio2': Directory not empty
dcdevs in ~/Desktop/media
λ rm -r diretorio2
```

```
dcdevs in ~/Desktop/media
λ tree
.
├── Diego-DevsS
│   ├── image
│   │   ├── ss.gif
│   │   ├── puddle_jumper_octodex.png
│   │   └── wave.gif
│   └── README.md
├── diretorio1
├── diretorio2
├── diretorio3
├── arquivo.txt
└── subdiretorio

5 directories, 5 files
dcdevs in ~/Desktop/media
λ ls -la Diego-DevsS
ls: cannot access 'Diego-DevsS': No such file or directory
dcdevs in ~/Desktop/media
λ rm -r Diego-DevsS
rm: remove write-protected regular file 'Diego-DevsS/.git/objects/pack/pack-aa06c0adfe48a492936dbf819d70e83970f298f8.pack'?
rm: remove write-protected regular file 'Diego-DevsS/.git/objects/pack/pack-aa06c0adfe48a492936dbf819d70e83970f298f8.idx'?
dcdevs in ~/Desktop/media took 5s
λ rm -rf Diego-DevsS
dcdevs in ~/Desktop/media
λ ls
diretorio1 diretorio3
dcdevs in ~/Desktop/media
λ
```

10. touch

O comando **touch** permite criar um novo arquivo em branco através da linha de comando Linux. Como exemplo, digite toque

/home/username/Documents/Web.html para criar um arquivo HTML intitulado **Web** sob o diretório **Documents**.

```
dcdevs in ~/Desktop/media
λ mkdir html css script
dcdevs in ~/Desktop/media
λ ls
css  html  script
dcdevs in ~/Desktop/media
λ touch README.md
dcdevs in ~/Desktop/media
λ touch css/index.css
dcdevs in ~/Desktop/media
λ touch html/index.html script/index.js
dcdevs in ~/Desktop/media
λ tree
.
├── css
│   └── index.css
├── html
│   └── index.html
├── README.md
└── script
    └── index.js

3 directories, 4 files
dcdevs in ~/Desktop/media
λ
```

```
dcdevs in ~/Desktop/media
λ tree
.
├── css
│   └── index.css
├── html
│   └── index.html
├── README.md
└── script
    └── index.js

3 directories, 4 files
dcdevs in ~/Desktop/media
λ touch script/index{1..4}.js
dcdevs in ~/Desktop/media
λ tree
.
├── css
│   └── index.css
├── html
│   └── index.html
├── README.md
└── script
    ├── index1.js
    ├── index2.js
    ├── index3.js
    ├── index4.js
    └── index.js

3 directories, 8 files
dcdevs in ~/Desktop/media
λ
```

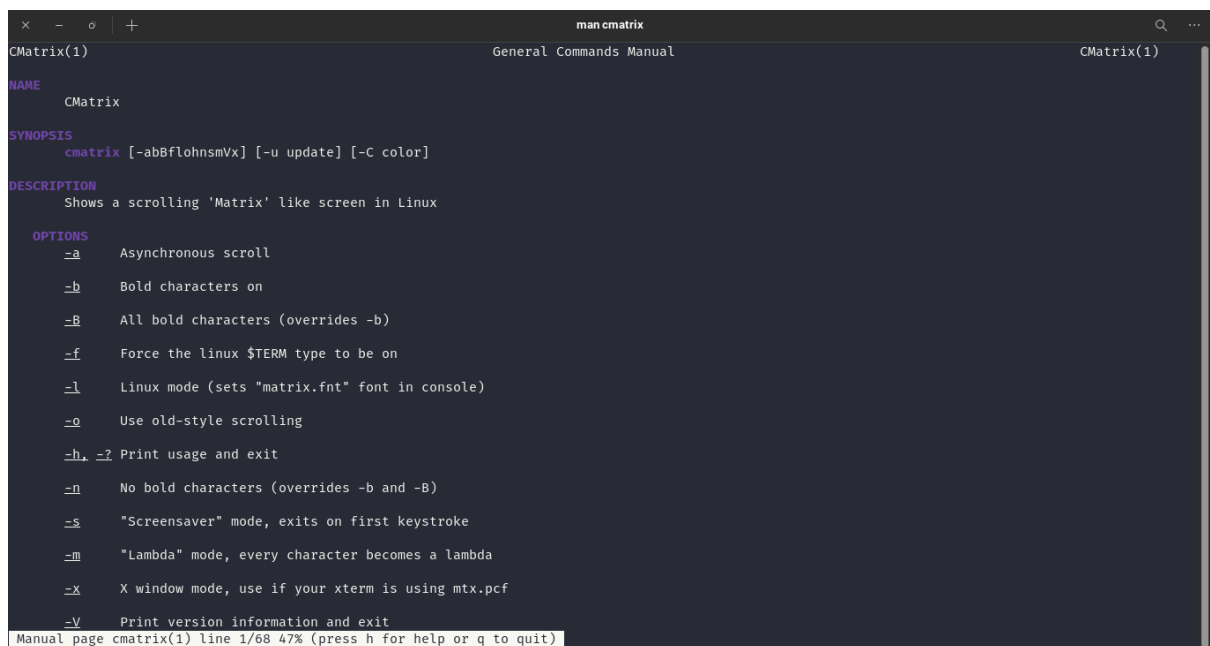
14. sudo

Abreviação de "**SuperUser Do**", este comando permite que você execute tarefas que requerem permissões administrativas ou raiz. No entanto, não é aconselhável usar este comando para uso diário, pois pode ser fácil para um erro ocorrer se você fez algo errado.


```
dcdevs in ~  
λ sudo su  
[sudo] password for dcdevs:  
diego-72#
```

30. man

Confused about the function of certain Linux commands? Don't worry, you can easily learn how to use them right from Linux's shell by using the **man** command. For instance, entering **man tail** will show the manual instruction of the tail command.



```
man cmatrix  
CMatrix(1)                                General Commands Manual                                CMatrix(1)  
  
NAME  
    CMatrix  
  
SYNOPSIS  
    cmatrix [-abBflohnsmVx] [-u update] [-C color]  
  
DESCRIPTION  
    Shows a scrolling 'Matrix' like screen in Linux  
  
OPTIONS  
    -a      Asynchronous scroll  
    -b      Bold characters on  
    -B      All bold characters (overrides -b)  
    -f      Force the linux $TERM type to be on  
    -l      Linux mode (sets "matrix.fnt" font in console)  
    -o      Use old-style scrolling  
    -h, -?  Print usage and exit  
    -n      No bold characters (overrides -b and -B)  
    -s      "Screensaver" mode, exits on first keystroke  
    -m      "Lambda" mode, every character becomes a lambda  
    -x      X window mode, use if your xterm is using mt.x.pcf  
    -V      Print version information and exit  
Manual page cmatrix(1) line 1/68 47% (press h for help or q to quit)
```