



visitante@portaldoshell: ~/cursos

```
$ man programacao-shell-linux
```

EMENTA DO CURSO

Programação Shell **Linux**

com **Julio Cezar Neves**

[ok] 9 módulos · [ok] 188 tópicos · [ok] + de 50 horas ao vivo

```
$ portaldoshell_
```

portaldoshell.com.br

01 Visão geral do sistema operacional Linux

- 1 Quem não precisa de Shell?
- 2 Por que Shell?
- 3 Tarefas do Shell
- 4 Ordem de execução
- 5 Exame da linha de comandos recebida
 - 5.1 Comando
 - 5.2 Atribuição
- 6 Resolução de redirecionamentos
- 7 Substituição de variáveis
- 8 Substituição de metacaracteres
- 9 Expansão de chaves {...}
- 10 Passa linha de comando para o kernel
- 11 Principais Shells
 - 11.1 Bourne Shell
 - 11.2 Bourne-Again Shell
 - 11.3 Korn Shell
 - 11.4 C Shell
- 12 Como fazer comentários
- 13 Usando aspas, apóstrofes e barra invertida
- 14 Crase e parênteses resolvendo crise entre parentes
- 15 Direcionando os caracteres de redirecionamento

02 Pesquisa e alteração de arquivos

- 1 Edição de arquivos – sed
 - 1.1 Imprimindo linhas escolhidas
 - 1.1.1 A opção -n (não imprima tudo)
 - 1.1.2 Definindo intervalos de pesquisa
 - 1.2 Deletando linhas escolhidas
 - 1.3 Inserindo e trocando linhas
 - 1.4 A opção -e
 - 1.5 Abortando o sed
 - 1.6 O comando de substituição
 - 1.6.1 Dicas sobre delimitadores
 - 1.6.2 Guardando cadeias para uso posterior
 - 1.7 O comando y
 - 1.8 Sed e etc.
 - 1.8.1 Gravando em outro arquivo
 - 1.8.2 Inserindo dados de outro arquivo
 - 1.8.3 Numerando linhas
 - 1.8.4 A opção -r

- 1.8.5 A opção -i
- 1.8.6 A opção -f
- 1.8.7 A opção -s
- 1.8.8 Evitando o pipe
- 1.8.9 sed multilinha
 - 1.8.9.1 Pattern Space (ou Pattern Buffer)
 - 1.8.9.2 Hold Space (ou Hold Buffer)
 - 1.8.9.3 Comandos que interagem com os buffers
 - 1.8.9.4 Gráfico dos comandos multilinhas
- 1.8.10 Alterando o fluxo do programa
 - 1.8.10.1 Especificando rótulos (labels)
 - 1.8.10.2 Testando condições
 - 1.8.10.3 Desvio incondicional

2 Às vezes os navegadores ajudam

3 A família de comandos grep

- 3.1 A opção -c (count ou contar)
- 3.2 A opção -l
- 3.3 A opção -v
- 3.4 A opção -f (file)
- 3.5 A opção -o (only matching)
 - 3.5.1 Pesquisando palavras
 - 3.5.2 Antes ou depois

03 Trabalhando com cadeias

1 Os comandos para cortar e colar

- 1.1 Cortando pedaços de arquivos – cut
 - 1.1.1 A opção -c (character)
 - 1.1.2 A opção -f (field)
 - 1.1.3 A opção -d (delimiter)
 - 1.1.4 Opções do cut muito úteis porém pouco usadas
- 1.2 Colando arquivos – paste
 - 1.2.1 A opção -d (delimiter)
 - 1.2.2 A opção -s (serial)
 - 1.2.3 Perfumarias úteis

2 O tr traduz, transcreve ou transforma cadeias de caracteres?

- 2.1 A opção -s
- 2.2 A opção -d
- 2.3 A opção -c
- 2.4 A opção -t

3 Exprimindo o expr de forma expressa

- 3.1 Execução de operações aritméticas
 - 3.1.1 Aritmética com expr
 - 3.1.2 Aritmética com bc
 - 3.1.3 O interpretador aritmético do Shell

03 Trabalhando com cadeias

(continuação)

- 3.2 Pegando tamanho de cadeias
 - 3.2.1 Com o expr
 - 3.2.2 Com Expansão de Parâmetros
- 3.3 Pegando subcadeias
 - 3.3.1 Com expr
 - 3.3.2 Com Expansão de Parâmetros
- 3.4 Localizando caracteres
- 3.5 Pesquisando casamento com match

4 O uniq é único

- 4.1 As opções -d, -w e -u
- 4.2 A opção -c

5 Mais redirecionamento sob o bash

04 Recebendo dados externos

- 1 Executando um programa
- 2 Usando variáveis
 - 2.1 Para criar variáveis
 - 2.2 Para exibir o conteúdo das variáveis
- 3 Passando e recebendo parâmetros
- 4 Comando que passa parâmetros - xargs
 - 4.1 A opção -0 ou --null
 - 4.2 As opções -i e -I
 - 4.3 A opção -n
 - 4.4 A opção -L
 - 4.5 As opções -t e -p

05 Comandos condicionais

- 1 O bom e velho if
- 2 Testando o test
 - 2.1 O test de roupa nova
 - 2.2 Se alguém disser que eu disse, eu nego...
 - 2.3 O conector and
 - 2.4 O conector or (ou)
- 3 Os operadores que se disfarçam de if
 - 3.1 && (and ou e lógico)
 - 3.2 || (or ou ou lógico)
 - 3.3 Operadores aritméticos para testar
 - 3.4 E tome de test
 - 3.5 Diferenças entre o novo ([[]) e o velho ([]) test
- 4 O caso em que o case casa melhor

06 Comandos de loop

- 1 O forró do for
 - 1.1 Uma forma (sintaxe) de usar o for
 - 1.2 Outra forma de usar o for
 - 1.3 IFS - Inter Field Separator
 - 1.4 Mais uma sintaxe do for
- 2 É while, uai!
- 3 O until é útil
- 4 O continue e o break

07 Formatando leitura e escrita

- 1 Que posição você prefere?
- 2 Afinal como é que se lê?
 - 2.1 Lendo arquivos
 - 2.2 Leitura dinâmica
 - 2.3 Leitura sob o bash
 - 2.3.1 Opção -p
 - 2.3.2 Opção -t
 - 2.3.3 Opção -n
 - 2.3.4 Opção -s
 - 2.3.5 Opção -d
 - 2.3.6 Opção -a
 - 2.3.7 Opção -r
 - 2.4 Esquisitices do IFS no read
- 3 Outra forma de ler e gravar em arquivos
 - 3.1 O comando exec
- 4 Já sei ler. Será que sei escrever?

08 Trabalhando com parâmetros e variáveis

- 1 Exportar é o que importa
- 2 É . e pronto
- 3 Principais variáveis do Bash
 - 3.1 CDPATH
 - 3.2 HOME
 - 3.3 LANG
 - 3.4 OLDPWD
 - 3.5 PIPESTATUS
 - 3.6 PROMPT_COMMAND
 - 3.7 PS1 e PS2
 - 3.8 RANDOM
 - 3.9 REPLY
- 4 Expansão de Parâmetros
 - 4.1 Para fazer Indireção
 - 4.2 Expansão de nomes de variáveis
 - 4.3 Para assumir valores padrão (default)



08 Trabalhando com parâmetros e variáveis (continuação)

- 4.4 Atribuição de valor padrão (default)
- 4.5 Informe de erro
- 4.6 Gera um valor alternativo
- 4.7 Remoção de subcadeias
- 4.8 Troca de subcadeias
- 4.9 Mudanças de caixa (alta e baixa)

5 ganhando o jogo com mais curingas

- 5.1 Estendendo pesquisa de arquivos pelo nome
- 5.2 cdspell
- 5.3 cmdhist
- 5.4 dotglob

6 Vetores (arrays)

- 6.1 Um pouco de manipulação de vetores
- 6.2 Vetores associativos
- 6.3 Lendo um arquivo para um vetor

09 Etcetera

1 Eval, o comando que avalia e executa

2 Wait a minute Mr. Postman

3 Para evitar trapalhadas use o trap

4 Funções

- 4.1 Uma função "on error"

5 FIFO

- 5.1 Substituição de processos
- 5.2 Coprocessos

6 Fatiando opções com o getopt

- 6.1 Variáveis usadas pelo getopt:
- 6.2 Manipulação de erros
- 6.3 Como esse cara funciona?

7 Em busca do erro perdido

8 Copiar e colar nas áreas de transferência

- 8.1 A área de transferência: você sabia?
- 8.2 Opções do comando xclip

9 Macetes, macetes & macetes



visitante@portaldoshell: ~

```
$ exit
```

**Do `echo` ao `trap`, sem pular
nenhum degrau.**

Datas, investimento e inscrição:
portaldoshell.com.br/cursos

```
[ ok ] ementa carregada · [ .. ] aguardando a próxima turma
```

\$ portaldoshell_

portaldoshell.com.br · instagram.com/portaldoshell