

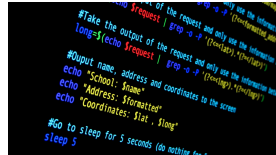
Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte
campus João Câmara

Curso Técnico de Nível Médio em Informática na Forma Integrada, presencial

Disciplina: Fundamentos de Sistemas Operacionais e Sistemas Operacionais de Rede

Professor: César Azevedo

Variáveis e exit codes



Trabalhando com variáveis no Shell Script

- Shell Script é uma linguagem onde as variáveis não possuem tipo definido (não tipada).
- Não é necessário especificar o tipo de dado que a variável vai armazenar.
- Para declarar uma variável, siga a sintaxe: NOME_DA_VARIAVEL=valor.
- É comum usar letras maiúsculas ao criar nomes de variáveis (exemplo VAR)
 - NOME=parâmetro: (não usar espaços depois do nome da variável)
 - exemplos:
 - Para armazenar o resultado de um comando
 - VARCOMANDO=`date` ou
 - VARCOMANDO=\$(date)
 - Para armazenar uma string ou número:
 - VARSTRING=ALUNO
 - VARNUMERO=123
 - para exibir a variável, usar echo \$NOMEDAVARIAVEL
 - echo \$VARCOMANDO
 - echo \$VARSTRING
 - echo \$VARNUMERO

O uso das aspas no Shell Script

- Se usarmos **echo *** o shell irá mostrar todos os arquivos do diretórios local
- Se quisermos imprimir o caractere *, devemos “protegê-lo” com aspas duplas “ ” ou aspas simples (apóstrofes) ‘ ’ ou barra invertida: \
 - echo “*”
 - echo ‘*’

- echo *
- Exemplo:
 - se usarmos ls aluno*
 - o shell irá listar todos os arquivos que começam com o nome aluno
 - se usarmos ls "alunos*"
 - o shell irá listar um arquivo chamado alunos*
 - qual a diferença entre aspas duplas (") e aspas simples (')?
 - aspas duplas "protegem" todos os caracteres, exceto:
 - cifrão, crase e a barra
 - aspas simples "protegem" tudo, inclusive os caracteres anteriores.
 - Exemplos
 - echo "\$TESTE" (irá mostrar o conteúdo da variável)
 - echo '\$TESTE' (irá mostrar \$TESTE)
 - echo \ \$TESTE (irá mostrar \$TESTE)

Códigos de Saída (Exit Codes)

- Os códigos de saída indicam se um script foi executado com sucesso ou apresentou algum erro.
- Eles variam de 0 a 255.
 - O código 0 indica sucesso.
 - Qualquer valor diferente de 0 indica que ocorreu um erro.
 - O significado de cada código é definido pelo desenvolvedor.
 - A variável \$? retorna o código de saída do último comando executado.
 - exemplo 1:
 - touch teste
 - echo \$?
 - 0
 - exemplo 2
 - touch -xxx teste
 - echo \$?
 - 1
 - normalmente, os códigos de erro são iguais a 1 ou 2

- Códigos de Erro no Manual (man)
 - Normalmente, o manual de um comando (man) inclui uma seção com os códigos de erro.
 - Cada comando pode explicar esses códigos de forma diferente; alguns comandos não especificam esses códigos explicitamente.
 - man do comando `ls`:

```
Exit status:
  0      if OK,
  1      if minor problems (e.g., cannot access subdirectory),
  2      if serious trouble (e.g., cannot access command-line argument).

AUTHOR
  Written by Richard M. Stallman and David MacKenzie.

REPORTING BUGS
  GNU coreutils online help: <http://www.gnu.org/software/coreutils/>
  Report ls translation bugs to <http://translationproject.org/team/>
```

- É útil armazenar o código de erro de cada comando para que o programador possa verificar qual erro ocorreu durante a execução do script.

Recebendo Entrada de Usuários com o Comando read

- `read NOME` espera o usuário digitar algo e armazena o texto digitado na variável `NOME`.
- `echo $NOME` exibe o conteúdo armazenado na variável ``NOME``.
- `read VAR1 VAR2 VAR3` permite que o usuário digite três valores, separando-os por espaço, e armazena cada um nas variáveis ``VAR1``, ``VAR2``, e ``VAR3``.
 - exemplo 1:
 - `read VAR1 VAR2 VAR3`
 - Curso Shell Script
 - VAR1 recebe Curso VAR2 recebe Shell VAR3 recebe Script
 - exemplo 2:
 - `read VAR1 VAR2`
 - Curso Shell Script
 - VAR1 recebe Curso VAR2 recebe Shell Script
 - exemplo 3: `read -p "Informe o seu Curso: " CURSO`
 - Informe o seu Curso
 - `CURSO <-` recebe entrada do teclado
 - exemplo 4: `read -s CODIGO`
 - *o valor digitado não aparece na tela (como em uma senha)*

Para realizar o script a seguir, crie primeiramente um arquivo em seu diretório home chamado **lista_alunos.txt** (vim lista_alunos.txt) com o seguinte conteúdo:

Beatriz Machado
Ana Trim
Zilah Freitas
Fabio Dantas
Marta Nogueira
Diogo Cortez
Valerio Pimenta

Exemplo:

Crie o script BuscaAluno.sh que recebe um nome e verifica se o mesmo existe no arquivo **lista_alunos.txt**.

solução:

```
#!/bin/bash
#####
# BuscaAluno.sh - Procurar nome completo de um aluno
#
# Autor: SeuNome
#
# Data de criação DD/MM/YYYY
#
# Descrição: Script de exemplo para o comando read.
#           A partir de uma entrada do usuário, será feita uma
#           busca no arquivo lista_alunos.txt
#
# Exemplo de uso: ./BuscaAluno.sh
#####

ARQALUNOS="/home/info4/lista_alunos.txt"

clear
echo "=====Script de Busca de Alunos=====
echo " "
echo -n "Por favor, informe o primeiro nome do aluno: "
read ALUNO

echo " "
echo "O nome completo do aluno é : "`grep $ALUNO $ARQALUNOS`
echo " "
echo "Fim do Script"
```

OBS.: Não esquecer de dar permissão de execução ao script antes de executá-lo (chmod +x BuscaAluno.sh ou chmod 744 BuscaAluno.sh)

Desafio: como buscar o nome do aluno ignorando se o usuário digitou maiúsculo ou minúsculo?

Referências:

<https://www.vivaolinux.com.br/artigo/O-que-e-Shell-Script>

<https://www.shellscript.sh>