

SHELL SCRIPT - ESTRUTURAS DE REPETIÇÃO

Rafael Antonello

Teste de Condição

- As condições de teste podem ser feitas em 3 categorias:
 - ▣ Comparação de strings, aritmética e de arquivos

Teste de Condição

if [condição]; then

comandos

else

comandos

fi

```
echo "Digite um numero"
read numero
if [ $numero -eq 0 ]; then
echo "O numero lido eh zero"
else
echo "O numero lido nao eh zero"
fi
exit 0
```

Teste de Condição

□ Comparação de aritmética

□ Verdadeiro se:

- | | |
|--------------------------------|--------------------------------|
| □ <code>expr1 -eq expr2</code> | São iguais |
| □ <code>expr1 -ne expr2</code> | Não são iguais |
| □ <code>expr1 -gt expr2</code> | <code>expr1 > expr2</code> |
| □ <code>expr1 -ge expr2</code> | <code>expr1 >= expr2</code> |
| □ <code>expr1 -lt expr2</code> | <code>expr1 < expr2</code> |
| □ <code>expr1 -le expr2</code> | <code>expr1 <= expr2</code> |
| □ <code>! expr</code> | Not expr |

Teste de Condição

```
#!/bin/bash
echo -n "digite o valor do número: "
read var1
if [ $var1 -eq 1 ]
then
    echo "Vc escolheu o Valor um"
elif [ $var1 -eq 2 ]; then
    echo "Vc escolheu o valor dois"
elif [ $var1 -eq 3 ]; then
    echo "Vc escolheu o valor três"
else
    echo "Vc escolheu uma opção inválida"
fi
```

Exercício Condicional

Retome o script esp01, verifique o valor do primeiro número, se este for maior do que o segundo multiplique-os e apresente o resultado. Se for inferior ao segundo número subtraia o primeiro do segundo e apresente o resultado.

esp01 – para lembrar ☺

Digite Primeiro Nro:

Digite Segundo Nro:

Após a entrada de dados imprima no vídeo a seguinte frase:

O Usuario XXX digitou os numeros XX e XX

A soma dos números foi XXX

Teste de Condição

□ Comparação de strings

□ Verdadeiro se:

- `string1 == string2` São iguais
- `string1 != string2` Não são iguais
- `-n string` String não é nulo
- `-z string` String é null (vazio)
- `string1 < string2` String1 um vem antes de
 `string2`
- `string1 > string2` String2 um vem antes de
 `string1`

Teste de Condição

```
#!/bin/bash
varx="Fulano"
echo "Digite um nome : "
read nome
if [ $nome == $varx ]; then
    echo "Acertou o nome"
else
    echo "Errou o nome"
fi
```


Verificar se uma palavra vem antes de outra

```
#!/bin/bash
echo "Digite a primeira string :"
read string1
echo "Digite a segunda string :"
read string2
if [ $string1 \< $string2 ]; then
    echo "String1 vem antes"
else
    echo "String2 vem antes"
fi
```

Comando elif

- Permite juntar uma condição if junto com um else

```
if [ condição1 ]; then
    comandos
elif [ condição2 ]; then
    comandos
elif [ condição3 ]; then
    comandos
fi
```

Substitui isso



```
if [ condição1 ]; then
    comandos
else
    if [ condição2 ]; then
        comandos
    else
        if [ condição3 ]; then
            comandos
        fi
    fi
fi
```

Operadores lógicos

- `&& [[(cond1) && (cond2)]]`

- ▣ `ou [cond1] && [cond2]`

- ▣ Ex.

- `if [[($var -le 10) || ($var -gt 20)]]; then`

- `|| [[(cond1) || (cond2)]]`

- ▣ Ex.

- `if [[($var -le 10) || ($var -gt 20)]]; then`

Comando case

- É um comando if mais elegante
- Com várias condições

```
case "$var" in
value1)
commands;
;;
value2)
commands;
;;
*)
commands;
;;
esac
```

Comando case

□ Exemplo

```
#!/bin/bash
echo "Digite um valor "
read var
case $var in
1) echo "O valor digitado foi um;;"
2) echo "O valor digitado foi dois;;"
esac
```

Exercício

- Utilize a estrutura case para fazer um programa que retorne o nome de um produto a partir do código do mesmo. Considere os seguintes códigos:
- 1 -> Parafuso;
- 2 -> Porca;
- 3 -> Pregoeiro;
- 4 -> Martelo;

SHELL SCRIPT - ESTRUTURAS DE REPETIÇÃO

Rafael Antonello