

Shell Script: condicionais if e case

Estrutura condicional if

```
if [ condição ]; then
    comando1
    comando2
fi
```

OBS: É preciso colocar espaço entre o colchete e o IF, assim como deve ter espaço entre os colchetes e a condição que está dentro deles.

Exemplo:

Se o conteúdo da variável USUARIO, na qual colocamos o nome do usuário do sistema, for igual à “adeoliveira”, crie a pasta “adeoliveira-backup”.

```
$ if [ $USUARIO = 'adeoliveira' ]; then
$   mkdir adeoliveira-backup
$ fi
```

Outras maneiras de escrever o mesmo código:

```
$ test $USUARIO = "adeoliveira" && mkdir adeoliveira-backup
$ [ $USUARIO = "adeoliveira" ] && mkdir adeoliveira-backup
```

Estrutura condicional if else

```
if [ condição ]; then
    comando1
    comando2
else
    comando3
    comando4
fi
```

Exemplo:

Se o conteúdo da variável USUARIO, na qual colocamos o nome do usuário do sistema, for igual à “adeoliveira”, crie a pasta “adeoliveira-backup”. Caso o conteúdo da variável seja diferente, crie uma pasta chamada apenas “backup”.

```
if [ $USUARIO = 'adeoliveira' ]; then
    mkdir adeoliveira-backup
else
    mkdir backup
fi
```

Operadores lógicos

-a (equivalente ao “e”)

-o (equivalente ao “ou”)

```
if [ condição1 -a condição2 ]; then
    comando1
    comando2
fi
```

```
if [ condição1 -o condição2 ]; then
    comando1
    comando2
fi
```

Estrutura condicional case

O comando **case** usada para “estrutura de menu”.

Verifica o conteúdo de variáveis;

```
case $VARIABEL in
    1) comando1;;
    2) comando2;;
    3) comando3;;
    4) comando4; comando5;;
esac
```

Exemplo:

Verifique o conteúdo da variável USUARIO que guarda o nome do usuário do sistema e imprima uma mensagem de boas vindas.

```
case $USUARIO in
    ana) echo “Bem-vinda Ana”;;
    carla) echo “Bem-vinda Carla”;;
    adeoliveira) echo “Bem-vinda Andréa”;;
esac
```

Opção Padrão do case

*) indica qual comando deve ser realizado caso nenhum das opções anteriores seja atendida

Exemplo:

Verifique o conteúdo da variável USUARIO que guarda o nome do usuário do sistema e print uma mensagem de boas vindas. No caso do usuário não ser o listado nas opções conhecida (ana, carla ou adeoliveira), imprima na tela uma mensagem geral de boas vindas.

```

case $USUARIO in
    ana) echo "Bem-vinda Ana";;
    carla) echo "Bem-vinda Carla";;
    adeoliveira) echo "Bem-vinda Andréa";;
    *) echo "Bem-vindo !!";;
esac

```

Criando Script no nano - Exemplos:

1. As maçãs de um agricultor custam R\$ 3,00 cada se forem compradas menos de uma dúzia, é R\$ 2,00 se forem compradas pelo menos 12. Crie um script que leia o número de maçãs compradas, calcule e escreva o custo total.

```

GNU nano 2.5.3      File: ./frutas.sh      Modified
#!/bin/bash
#Aula12
#Script que calcula o preço das frutas, caso sejam compradas menos de
#uma dúzia a unidade será 3,00 e a partir de uma dúzia unidade custa 2,00

# 1 MANEIRA DE CALCULAR
echo "Digite o numero de maçãs"
read FRUTA
[ $FRUTA -lt 12 ] && VALOR=$(( $FRUTA*3 ))
[ $FRUTA -ge 12 ] && VALOR=$(( $FRUTA*2 ))
[ $FRUTA -lt 12 -o $FRUTA -ge 12 ] && echo "valor a ser pago $VALOR" || echo "ERRO - digite um valor valido"

```

```

GNU nano 2.5.3      File: ./frutas2.sh
#!/bin/bash
#2 MANEIRA
echo "Digite o numero de maçãs"
read FRUTA
if [ $FRUTA -lt 12 ];then
    VALOR=$(( $FRUTA*3 ))
else
    VALOR=$(( $FRUTA*2 ))
fi
echo "Valor a ser pago $VALOR,00"

```

```

GNU nano 2.5.3      File: ./frutas3.sh      Modified
#!/bin/bash
#3 Maneira
echo "Digite o numero de maçãs"
read FRUTA
if [ $FRUTA -lt 12 ];then
    VALOR=$(( $FRUTA*3 ))
# VALOR=$(( $FRUTA*2 ))
fi

if [ $FRUTA -ge 12 ];then
    VALOR=$(( $FRUTA*2 ))
fi

if [ $FRUTA -lt 12 -o $FRUTA -ge 12 ]; then
    echo " Valor a ser pago é $VALOR,00"
else
    echo "ERRO - Digite valor valido."
fi
# Pode usar o "else" da primeira condicional que esta comentada
# ou a segunda condicional

```

2. Crie um script para uma lanchonete que leia o código do item do pedido, a quantidade e calcule o valor a ser pago. Não esqueça de adicionar uma mensagem de erro.

CODIGO	PRODUTO	PREÇO
1	X-Misto	5,00
2	Refrigerante	2,00
3	Sorvete	3,00

```
GNU nano 2.5.3 File: ./lanchonete.sh

#!/bin/bash
#Aula 12
#Script para uma lanchonete que leia o código do item do pedido,
#a quantidade e calcule o valor a ser pago.
# Não esqueça de incluir uma mensagem de erro.
echo 'Digite o código do produto'
read PRODUTO
case $PRODUTO in
  1) VALOR=5;NOME=X-Misto;;
  2) VALOR=2;NOME=Refrigerante;;
  3) VALOR=3;NOME=Sorvete;;
  *) echo 'CODIGO DO PRODUTO INVÁLIDO';;
esac
echo 'Digite a quantidade'
read QUANTIDADE
VALORFINAL=$((VALOR*QUANTIDADE))

echo "Valor a pagar $VALORFINAL,00"
echo "$QUANTIDADE - $NOME"
```

```
GNU nano 2.5.3 File: ./lanchonete2.sh Modified

#!/bin/bash
#Aula 12
#Script para uma lanchonete que leia o código do item do pedido,
#a quantidade e calcule o valor a ser pago.
# Não esqueça de incluir uma mensagem de erro.
echo 'Digite o código do produto'
read PRODUTO
echo "Digite a quantidade"
read QUANTIDADE

case $PRODUTO in
  1)
    echo "Você pediu: $QUANTIDADE - X-misto";
    echo "Valor a pagar: $((QUANTIDADE*5)),00";;
  2)
    echo "Você pediu: $QUANTIDADE - Refrigerantes";
    echo "Valor a pagar: $((QUANTIDADE*2)),00";;
  3)
    echo "Você pediu: $QUANTIDADE - Sorvetes";
    echo "Valor a pagar: $((QUANTIDADE*3))";;
  *)
    echo 'CODIGO INVÁLIDO';;
esac
```

3. Usando a estrutura “if” and “else” escreve um script que recebe um valor como parâmetro do script e informa se esse valor é maior, igual ou menor que 10.

```
alisandra@hpl: ~
GNU nano 2.5.3 File: ./test.sh

#!/bin/bash
# Usando a estrutura “if” and “else” escreve um script
# que recebe um valor como parâmetro do script e
# informa se esse valor é maior, igual ou menor que 10

if [ $1 -gt 10 ]
then
    echo "maior que 10"
else
    if [ $1 -eq 10 ]
    then
        echo "igual á 10"
    else
        echo "menor que 10"
    fi
fi
```