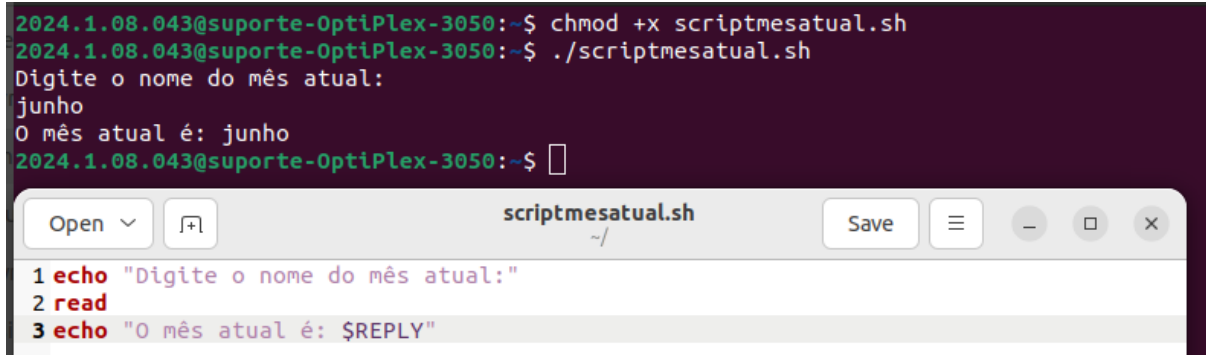


Bartholomeu Vitor Neto  
2024.1.08.043

1. Qual variável de ambiente guarda o valor lido pelo comando `read`, caso não tenha nenhuma criada para guardar o valor? Faça um script utilizando essa variável de ambiente, onde a entrada será o nome do mês atual. Mostre a execução. Depois acrescente um timer de 8 segundos, como mostrado em aula e apresente a execução.

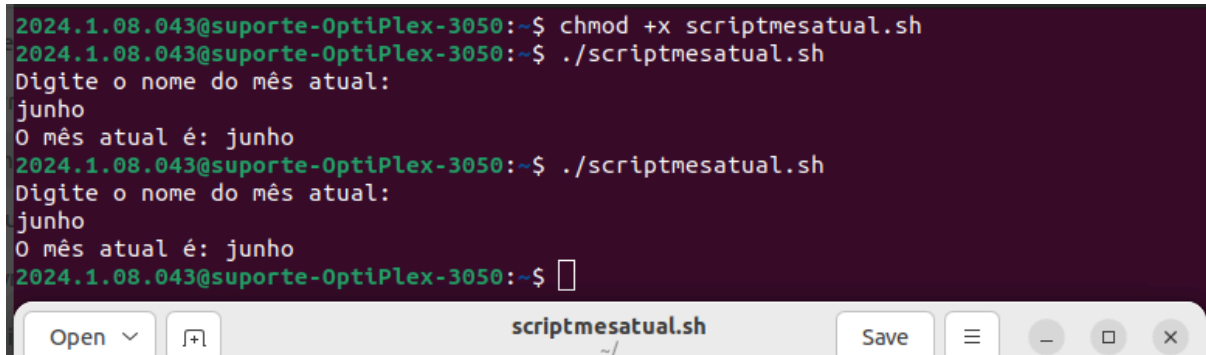
A variável de ambiente que guarda o valor lido pelo comando `read`, caso não tenha nenhuma variável específica criada para guardar o valor, é a `$REPLY`.

```
2024.1.08.043@suporte-OptiPlex-3050:~$ chmod +x scriptmesatual.sh
2024.1.08.043@suporte-OptiPlex-3050:~$ ./scriptmesatual.sh
Digite o nome do mês atual:
junho
O mês atual é: junho
2024.1.08.043@suporte-OptiPlex-3050:~$
```



```
1 echo "Digite o nome do mês atual:"
2 read
3 echo "O mês atual é: $REPLY"
```

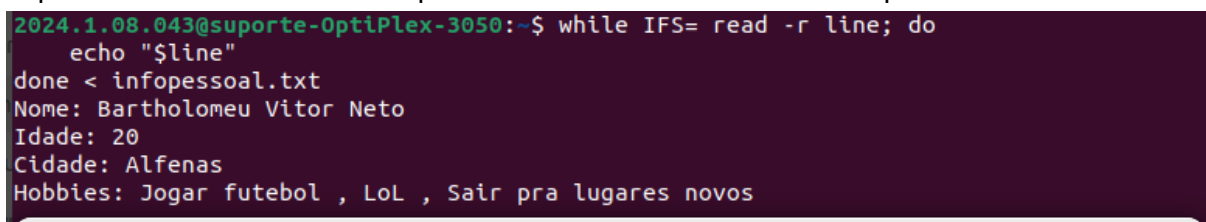
```
2024.1.08.043@suporte-OptiPlex-3050:~$ chmod +x scriptmesatual.sh
2024.1.08.043@suporte-OptiPlex-3050:~$ ./scriptmesatual.sh
Digite o nome do mês atual:
junho
O mês atual é: junho
2024.1.08.043@suporte-OptiPlex-3050:~$ ./scriptmesatual.sh
Digite o nome do mês atual:
junho
O mês atual é: junho
2024.1.08.043@suporte-OptiPlex-3050:~$
```



```
1 echo "Digite o nome do mês atual:"
2 read
3 sleep 8
4 echo "O mês atual é: $REPLY"
```

2. Crie um arquivo txt sobre você (nome, idade, cidade, hobbies, etc) e faça a leitura do arquivo com o comando `read`. Apresentando todo o conteúdo do arquivo no terminal.

```
2024.1.08.043@suporte-OptiPlex-3050:~$ while IFS= read -r line; do
    echo "$line"
done < infopessoal.txt
Nome: Bartholomeu Vitor Neto
Idade: 20
Cidade: Alfenas
Hobbies: Jogar futebol , LoL , Sair pra lugares novos
```



```
1 Nome: Bartholomeu Vitor Neto
2 Idade: 20
3 Cidade: Alfenas
4 Hobbies: Jogar futebol , LoL , Sair pra lugares novos
```

3.Desenvolva um script utilizando o comando case. Cada opção chama uma função e apresenta o resultado da função. Exemplos:

apresenta o resultado da função. Exemplos:

Digite um número entre 1 e 4: 4	Digite um número entre 1 e 4: 1
Digite um número: 5	Digite um número: 10
Subtraindo 10: -5	Multiplicando por 10: 100

```
2024.1.08.043@suporte-OptiPlex-3050:~$ chmod +x scriptcase.sh
2024.1.08.043@suporte-OptiPlex-3050:~$ ./scriptcase.sh
2024.1.08.043@suporte-OptiPlex-3050:~$ chmod +x scriptcase.sh
2024.1.08.043@suporte-OptiPlex-3050:~$ ./scriptcase.sh
Escolha uma operação:
1. Adição
2. Subtração
3. Multiplicação
4. Divisão
1
Digite dois números para adicionar:
5
9
Passo 1: Somando 5 com 9
Passo 2: Resultado da adição: 14
2024.1.08.043@suporte-OptiPlex-3050:~$ ./scriptcase.sh
Escolha uma operação:
1. Adição
2. Subtração
3. Multiplicação
4. Divisão
1
Digite dois números para adicionar:
9
5
Passo 1: Somando 9 com 5
Passo 2: Resultado da adição: 14
2024.1.08.043@suporte-OptiPlex-3050:~$ ./scriptcase.sh
Escolha uma operação:
1. Adição
2. Subtração
3. Multiplicação
4. Divisão
3
Digite dois números para multiplicar:
5
5
Passo 1: Multiplicando 5 por 5
Passo 2: Resultado da multiplicação: 25
2024.1.08.043@suporte-OptiPlex-3050:~$
```

```
scriptcase.sh
~/
Open  [?] Save [≡] [—]

1 addition() {
2     echo "Digite dois números para adicionar:"
3     read num1
4     read num2
5     result=$((num1 + num2))
6     echo "Passo 1: Somando $num1 com $num2"
7     echo "Passo 2: Resultado da adição: $result"
8 }
9 subtraction() {
10    echo "Digite dois números para subtrair:"
11    read num1
12    read num2
13    result=$((num1 - num2))
14    echo "Passo 1: Subtraindo $num2 de $num1"
15    echo "Passo 2: Resultado da subtração: $result"
16 }
17 multiplication() {
18    echo "Digite dois números para multiplicar:"
19    read num1
20    read num2
21    result=$((num1 * num2))
22    echo "Passo 1: Multiplicando $num1 por $num2"
23    echo "Passo 2: Resultado da multiplicação: $result"
24 }
25 division() {
26    echo "Digite dois números para dividir:"
27    read num1
28    read num2
29    result=$(bc <<< "scale=2; $num1 / $num2")
30    echo "Passo 1: Dividindo $num1 por $num2"
31    echo "Passo 2: Resultado da divisão: $result"
32 }
33 main() {
34    echo "Escolha uma operação:"
35    echo "1. Adição"
36    echo "2. Subtração"
37    echo "3. Multiplicação"
38    echo "4. Divisão"
39    read choice
40
41    case $choice in
42        1) addition ;;
43        2) subtraction ;;
44        3) multiplication ;;
45        4) division ;;
46        *) echo "Opção inválida. Por favor, escolha uma opção válida." ;;
47    esac
48 }
49 main
```

4. Modifique o script abaixo para que apresente o dobro apenas dos números entre 20 e 30.

```
1#!/bin/bash
2#Function Output
3
4function quadrado {
5    read -p "Digite um número entre 20 e 30: " numero
6    echo ${ numero * $numero }
7}
8
9#Atribuindo o valor da função à variável do shell valor:
10 valor=`quadrado`
11
12#Mostrando o valor:
13 echo "O quadrado do número é $valor"
```



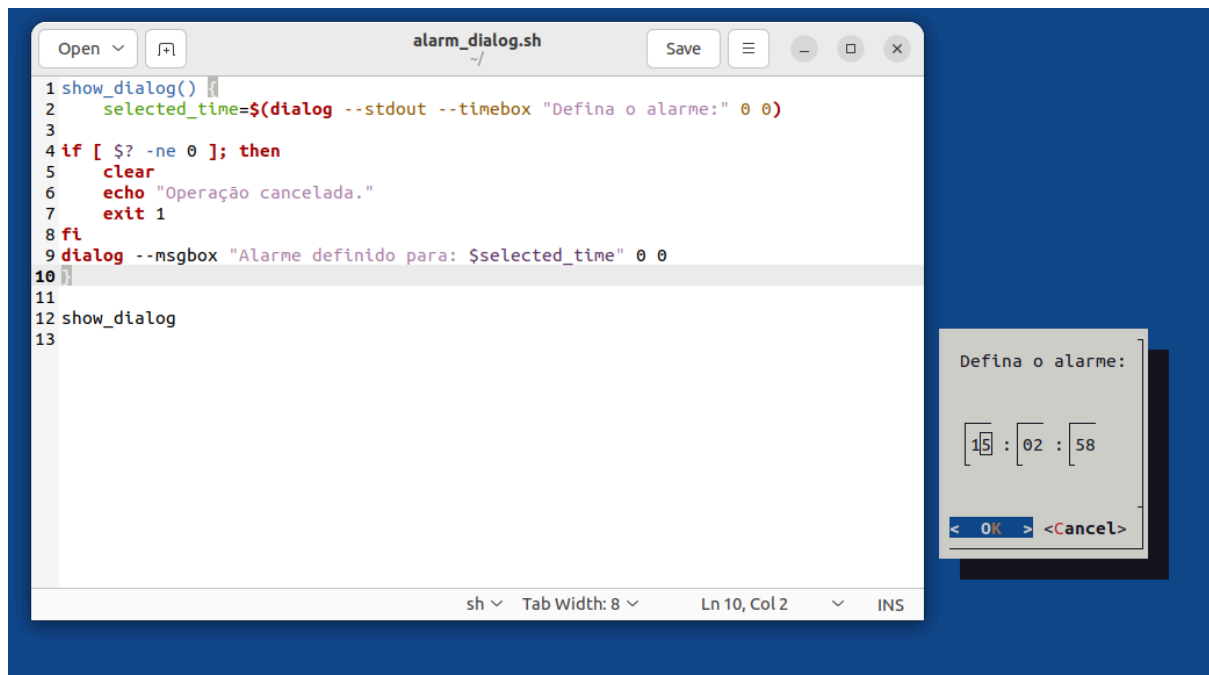
```
1 function dobro {
2     read -p "Digite um número entre 20 e 30: " numero
3     if ((numero >= 20 && numero <= 30)); then
4         resultado=$((numero * 2))
5         echo "O dobro do número é $resultado"
6     else
7         echo "O número digitado não está entre 20 e 30."
8     fi
9 }
10 dobro
```

5. Crie um script como whiptail, receba um número e mostre o dobro daquele número.



```
1 show_dialog() {
2 num=$(dialog --stdout --inputbox "Digite um número:" 0 0)
3 if [ $? -ne 0 ]; then
4     clear
5     echo "Operação cancelada."
6     exit 1
7 fi
8 if ! [[ "$num" =~ ^[0-9]+$ ]]; then
9     dialog --msgbox "Por favor, insira um número válido." 0 0
10    show_dialog
11 else
12    double=$((num * 2))
13    dialog --msgbox "O dobro do número é: $double" 0 0
14 fi
15 }
16 show_dialog
17
```

6. Escolha um widget do comando dialog e mostre sua execução. Escolha um diferente dos que foram mostrados em aula.



7. Desenvolva um ckecklist, utilize sua criatividade. Pode ser whiptail ou dialog.

