

Instruções de Loop - Exercícios

1.

Crie um script que mostre todos os usuários “humanos” da máquina seguidos de seu UID, Diretório Home e Nome/Descrição.

Exemplo de Saída:

USUARIO	UID	DIR HOME	NOME OU DESCRIÇÃO
ricardo	1000	/home/ricardo	Ricardo Prudenciatto
aluno1	1001	/home/aluno1	Aluno de Teste 1
aluno2	1002	/home/aluno2	Aluno de Teste 2
andre	1003	/home/andre	Andre Novais

Dicas:

- Baseie-se pelas informações do arquivo /etc/passwd
- O arquivo /etc/login.defs mostra através dos parâmetros UID_MIN e UID_MAX os UID atribuídos aos usuários “normais” do sistema
- Para definir apenas a quebra de linha como separador no IFS use:
IFS=\$'\n'
- Use o a opção -e do echo para controlar o espaçamento da saída com tabs (\t)

2.

Crie um script que receba um nome de processo como argumento e que constantemente irá verificar se o processo está em execução.

Caso não esteja, deve ser exibida uma mensagem na sessão do usuário a cada x segundos.

O processo deve rodar como Daemon, ou seja, deve estar sempre em execução, verificando o processo em questão a cada x segundos.

Exemplo de Execução:

```
$ ./MonitoraProcesso.sh firefox &  
ATENÇÃO!!!!!!! O processo firefox NÃO está em execução!  
ATENÇÃO!!!!!!! O processo firefox NÃO está em execução!  
ATENÇÃO!!!!!!! O processo firefox NÃO está em execução!  
ATENÇÃO!!!!!!! O processo firefox NÃO está em execução!  
ATENÇÃO!!!!!!! O processo firefox NÃO está em execução!  
ATENÇÃO!!!!!!! O processo firefox NÃO está em execução!
```

3.

Fazer um script que inspecione os diretórios /home/ de todos os usuários em busca de arquivos com as extensões .mp3, .mp4 e .jpg.

O script deve gerar como saída final um relatório com a quantidade de cada tipo de arquivo para cada usuário.

Exemplo de Execução:

```
$ ./InspecionaHome.sh
```

Usuario: aluno1

Arquivos JPG: 8

Arquivos MP3: 0

Arquivos MP4: 12

Usuario: aluno2

Arquivos JPG: 0

Arquivos MP3: 0

Arquivos MP4: 0

Usuario: ricardo

Arquivos JPG: 685

Arquivos MP3: 0

Arquivos MP4: 81

Dicas:

- Você pode utilizar a seguinte sintaxe para o comando find: find /home/usuario -name '*.jpg' -o -name '*.mp4' -o -name '*.mp3'

4.

Fazer um script que crie um arquivo e o preencha com um conjunto de caracteres até que um determinado tamanho em bytes seja atingido.

O usuário deverá fornecer o nome do arquivo, o conjunto de caracteres que será utilizado para preenchimento do arquivo, e o tamanho final do arquivo em bytes.

Crie também uma lógica que mostre o andamento da criação desse arquivo, de 10 em 10%, e o tamanho atual do arquivo.

Se o arquivo já existir, toda a informação anterior será apagada.

Exemplo de Execução:

\$./CriaArquivo.sh

Informe o nome do arquivo a ser criado: arquivo_teste.out

Informe um conjunto de caracteres que será usado para preencher o arquivo: a

Informe o tamanho final do arquivo (em bytes): 100000

Concluído: 10% - Tamanho do Arquivo: 10000

Concluído: 20% - Tamanho do Arquivo: 20000

Concluído: 30% - Tamanho do Arquivo: 30000

Concluído: 40% - Tamanho do Arquivo: 40000

Concluído: 50% - Tamanho do Arquivo: 50000

Concluído: 60% - Tamanho do Arquivo: 60000

Concluído: 70% - Tamanho do Arquivo: 70000

Concluído: 80% - Tamanho do Arquivo: 80000

Concluído: 90% - Tamanho do Arquivo: 90000

Concluído: 100% - Tamanho do Arquivo: 100000

Dicas:

- Para saber o tamanho atual do arquivo, use o comando:
 - `stat --printf=%s arquivo`
- Use a opção % do expr, que retorna o resto de uma divisão, para saber quando exibir a mensagem de porcentagem (10, 20, 30, etc)