

EXERCÍCIO 2

1. Crie um script que recebe um nome de usuário como argumento. Em seguida valide se este usuário existe.
2. Crie um script que exiba na tela quantos processos estão em execução.
3. Crie um script que exibe quantas linhas há em um arquivo. O nome do arquivo deve ser passado por parâmetro.
4. Crie um script que receba N parâmetros. Em seguida verifique se todos os N parâmetros foram passados.
5. Crie um script para validar se um arquivo existe. A entrada com o nome do arquivo deve ser passada por parâmetro.
6. Crie um script para validar se um arquivo existe. A entrada com o nome do arquivo deve ser interativa.
7. Crie um script interativo que recebe como entrada um nome de diretório de origem e um nome de diretório de *backup*. Realize o *backup* entre os diretórios.
8. Crie um script que receba por argumento um nome de arquivo. Este arquivo deve conter alguns nomes. Em seguida, realize uma verificação para garantir que o arquivo existe. Se o arquivo existe crie um diretório para cada um dos nomes do arquivo.
 - Siga o formato para criar o nome do diretório: `aluno_<nome_aluno>`
9. Elabore um script para coletar e armazenar informações do SO:
 - Pode ser passado como argumento o nome de usuário. Caso não haja argumento, deve ser atribuído o usuário atual.
 - As informações coletadas devem ser armazenadas em um arquivo txt. O nome deve ter o seguinte formato:
<AAAAMMDD>_<HH:MM:SS>_EstadoSistema_<usuário>.txt
 - Dica de formatação: `date +"%Y%m%d_%T"`
 - As seguintes informações (nesta sequência) devem ser gravadas:
 - Número total de processos ativos no sistema.
 - Número de processos ativos do usuário.
 - Nome do usuário.
 - Quantidade de processos sendo executados excluindo-se os do usuário.
 - Quantidade de processos sendo executados como root.
10. (DESAFIO) Crie um script para ler um arquivo CSV, em seguida exiba os dados de forma organizada na tela.