Por: Alice Dantas

Navegação no sistema de arquivos

pwd (print working directory)

Mostra o caminho completo do diretório atual.

cd (change directory)

Muda de diretório.

cd /home/querida → Vai para a pasta /home/querida.
cd ~ → Vai direto para o seu diretório pessoal (/home/querida).
cd → Também leva ao seu diretório pessoal (/home/querida)
cd .. → Sobe um nível na hierarquia (vai para o diretório pai).
cd ../../.. → Sobe os diretórios.

ls (list)

Lista os arquivos e pastas de um diretório.

ls → Lista arquivos e pastas do diretório atual.
ls /caminho_para_pasta → Lista o conteúdo de um diretório específico.
ls -l → Mostra uma listagem detalhada (permissões, dono, tamanho, data).
ls -a → Mostra todos os arquivos, inclusive os ocultos (começam com .)
ls -la ou ls -al → Combina as opções acima.
ls -R → Lista recursivamente os subdiretórios.

Por: Alice Dantas

Manipulação de arquivos e diretórios

- Comandos para criar, apagar, copiar e mover arquivos

mkdir (make directory)

Cria pastas novas.
mkdir nova_pasta → cria uma pasta chamada nova_pasta dentro do diretório atual.
mkdir dir1 dir2 dir3 → cria várias pastas ao mesmo tempo, dentro do diretório atual.
mkdir /home/querida/dir1/dir2 → cria um diretório em um caminho específico.
mkdir -p /home/usuario/pasta1/pasta2/pasta3 → permite criar um diretório **e todos os diretórios necessários** no caminho, caso eles não existam.

touch

Cria um arquivo vazio.
touch nome_do_arquivo → cria um arquivo chamado nome_do_arquivo, no diretório atual.

rmdir

Apaga **somente** diretórios vazios.
rmdir nome_do_diretorio_vazio → remove o diretório vazio.

rm (remove)

Pode apagar arquivos e diretórios.
rm arquivo.txt → remove o arquivo “arquivo.txt”.
rm -r nome_do_dir → remove o diretório e seu conteúdo (incluindo subdiretórios e arquivos).

cp (copy)

Cria uma cópia de arquivos e diretórios. Ps.: escrever o caminho do destino.
cp arq_ou_dir_origem arq_ou_dir_destino → copia arquivos ou diretórios para outros arquivos ou diretórios.
cp -r dir_origem dir_destino → copia o diretório “dir_origem” e todos seus arquivos e subdiretórios para o diretório “dir_destino”.
cp -a dir_origem dir_destino → faz a cópia, preservando a estrutura original, como subdiretórios, data, autor, etc.

mv (move)

Move arquivos e diretórios. Ps.: escrever o caminho do destino.
mv nome_ou_caminho_do_arquivo_origem diretório_destino → move um arquivo para um diretório.
mv arq1.txt arq2.txt /home/usuario/backup/ → move múltiplos arquivos, no caso “arq1.txt” e “arq2.txt”, para o mesmo diretório.
mv caminho_do_drt1 caminho_do_drt2 → move o diretório “dr1” para o diretório “dr2”.

Por: Alice Dantas

- Comandos para editar ou visualizar conteúdo de arquivos

nano

Abre um arquivo existente no editor, ou cria um arquivo e o abre, se ele não existir.

nano arquivo_fofinho → Se “arquivo_fofinho” existir, ele apenas abrirá pelo editor, senão, ele cria “arquivo_fofinho” vazio e o abre no editor.

cat

Ele pode mostrar o conteúdo de um arquivo no terminal, concatenar arquivos (juntar vários arquivos em um só), e criar novos arquivos de texto.

cat arquivo.txt → exibe o conteúdo do arquivo arquivo.txt no terminal.
cat arq1.txt arq2.txt → exibe o conteúdo de arq1.txt seguido pelo conteúdo de arq2.txt.
cat arq1.txt arq2.txt > arq_combinado.txt → isso cria o arquivo “arq_combinado.txt” com o conteúdo de arq1.txt seguido de arq2.txt.
cat > novo_arquivo.txt → Cria um arquivo chamado “novo_arquivo.txt”. Após executar esse comando, basta digitar o conteúdo do arquivo diretamente. Quando terminar, pressione **Ctrl + D** para salvar e sair. O conteúdo digitado será salvo no arquivo novo_arquivo.txt.
cat >> arquivo_existente.txt → Adiciona conteúdo a um arquivo existente, sem substituir o conteúdo atual. Após digitar o conteúdo, pressione **Ctrl + D** para salvar. O conteúdo digitado será adicionado ao final de arquivo_existente.txt.

cat com | grep

Exibe o conteúdo de um arquivo com o cat e, em seguida, filtrar esse conteúdo usando o grep.

cat arquivo.txt | grep "termo_de_busca" → O *cat* exibe o conteúdo do arquivo. O *pipe* redireciona a saída do comando *cat* para o *grep*, ou seja, o conteúdo de “arquivo.txt” é passado como entrada para o *grep*. O *grep* filtra o conteúdo exibido, mostrando apenas as linhas que contêm o texto ou padrão especificado em *termo_de_busca.*

paste

Combina linhas de vários arquivos lado a lado, unindo os conteúdos em colunas separadas por tabulação.

paste arquivo1 arquivo2 → mostra no terminal o conteúdo dos dois arquivos lado a lado.
paste arquivo1 arquivo2 > arquivo3 → salva dentro de arquivo3 (preexistente) o conteúdo de “arquivo1” e “arquivo2” lado a lado.
paste -d “,” arquivo1 arquivo2 > arquivo3 → salva dentro de arquivo3 (preexistente) o conteúdo de “arquivo1” e “arquivo2” lado a lado. Contudo, o uso do -d permite que escolha o que vai separar os conteúdos do arquivo; nesse caso, uma vírgula.
paste alunos.txt notas.txt | column -t > resultado.txt → salva dentro de resultado.txt (preexistente) o conteúdo de “alunos.txt” e “notas.txt” formatados como tabela.

Por: Alice Dantas

echo (eco de repetição de som)

Imprime uma mensagem no terminal. Mas também pode ser usado para gravar texto em arquivos, utilizando o operador de redirecionamento **>** ou **>>**.

echo "Olá, Mundo!" → imprime o texto “Olá, Mundo!” no terminal.
echo "conteúdo" > arquivo.txt → ao usar >, ele **cria um novo arquivo** com o nome fornecido (se o arquivo não existir) ou **substitui o conteúdo de um arquivo existente** (se o arquivo já existir).
echo "Texto adicional" >> arquivo.txt → adiciona a string "Texto adicional" ao final de arquivo.txt sem apagar o conteúdo existente.

grep

Busca por padrões em arquivos de texto.

grep "Josefa" alunos.txt → busca a palavra “Josefa” no arquivo “alunos.txt”.
grep -i “ola” alunos.txt → busca a palavra “ola” ignorando as maiúsculas e minúsculas no arquivo.
grep -E "[0-9]{3}-[0-9]{4}" arquivo.txt → busca por uma expressão regular no arquivo “arquivo.txt”.
grep "erro" *.txt → busca a palavra “erro” em todos os arquivos cuja extensão é “.txt”
grep -l "erro" *.log → exibe apenas os arquivos que contêm o padrão “erro”.
grep -v "erro" log.txt → exibe todas as linhas do arquivo “log.txt” que **não contêm** a palavra “erro”.

less

less arquivo.txt → permite navegar pelo conteúdo do arquivo de forma interativa, página por página.

Funcionalidades principais:

- Use as setas ↑ ↓ para subir e descer linha a linha.
- Use Page Up / Page Down para navegar mais rápido.
- Digite /palavra e pressione Enter para **buscar** dentro do texto.
- n vai para a **próxima ocorrência** da busca, N para a anterior.
- Pressione q para **sair** do less.

head

Exibe o início de um arquivo.

head arquivo.txt → mostra as 10 primeiras linhas do arquivo.
head -n numero_qualquer arquivo.txt → personaliza o número de linhas a serem mostradas.

tail

Exibe o final de um arquivo.

tail arquivo.txt → mostra as 10 últimas linhas do arquivo.
tail -n numero_qualquer arquivo.txt → personaliza o número de linhas a serem mostradas.