

# Trabalhando com Arquivos e Diretórios

Prof: Diógenes



Instituto Federal  
de Brasília



# Objetivos

- Entender Arquivos e Diretórios, incluindo os arquivos ocultos
- Diretório HOME
- Utilizar apropriadamente os caminhos absolutos e relativos

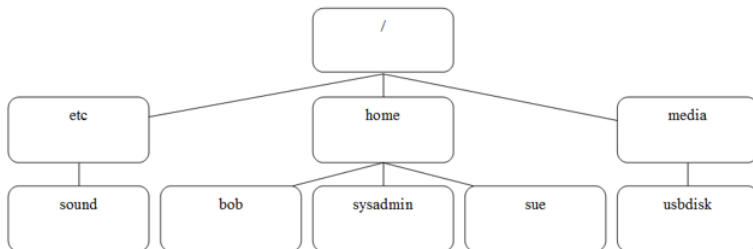


# Entendendo Diretórios



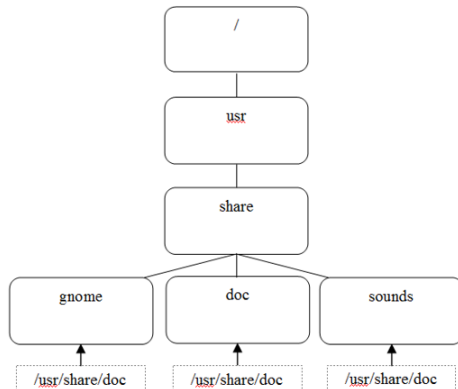
# Man Pages

- Arquivo contém dados (texto, gráficos, etc)
- Diretórios armazenam nomes de arquivos
- Diretório de nível superior: / (diretório raiz)



# Caminho de diretório

- Direções para um arquivo ou diretório específico
- As direções fornecidas no diretório / são chamadas de caminhos "**absolutos**"
- As direções fornecidas a partir do diretório atual são chamadas de caminhos "**relativos**"



# O diretório home

- Cada usuário tem um diretório `home`
- Normalmente `/home/bob` para um usuário chamado **bob**
- Local para armazenar seus próprios arquivos
- Normalmente, os usuários não podem acessar o diretório `home` de outros usuários
- O caractere `~` simboliza o diretório inicial



# Diretório Atual

- O diretório em que seu shell está atualmente
- Pode ser exibido com o comando `pwd`
- Também pode ser exibido no seu prompt



# Mudando de diretórios

- Use o comando `cd`
- Sem argumentos, leva você ao seu diretório home
- `~bob` se refere ao diretório home de **bob**





# Caminhos Absolutos vs Relativos

- Caminhos absolutos sempre fornecem instruções tendo como referência o diretório raiz (/)
- Os nomes de caminho relativos sempre fornecem instruções tendo como referência o diretório atual.
- Para se referir a um diretório acima do diretório atual, use os caracteres ..
- Para se referir ao diretório atual, use o caractere .



# Manipulando arquivos



# Listando arquivos num diretório

- Os arquivos são listados com o comando **ls**
- Diferentes tipos de arquivos podem ser destacados por nomes de arquivos coloridos:
  - **arquivo simples**: Um arquivo que não é um tipo de arquivo especial; também chamado de arquivo normal
  - **diretório**: Um arquivo de diretório (contém outros arquivos)
  - **executável**: Um arquivo que pode ser executado como um programa
  - **link simbólico**: Um arquivo que aponta para outro arquivo
- A exibição de nomes de arquivos em cores é o resultado da opção **--color**



# Opções comuns do ls

Muitas opções para o comando `ls`, incluindo:

- `-a` - exibe todos os arquivos, incluindo arquivos ocultos
- `-l` - lista de exibição longa
- `-h` - Fornece tamanhos de arquivo em tamanhos legíveis para humanos
- `-R` - Listagem recursiva
- `-S` - Ordena a saída com base no tamanho do arquivo
- `-t` - Ordena a saída com base no tempo de modificação
- `-d` - Não exibir o conteúdo do diretório



# Copiando arquivos

- O comando **cp** copia arquivos
  - `cp [origem] [destino]`
- Use a opção **-v** para exibir o processo de cópia
- Use **-i** para avisar sobre arquivos sobrescritos
- Use **-n** para evitar sobrescrever
- Use **-r** para copiar estruturas de diretórios



# Movendo arquivos

- O comando **mv** move arquivos:
  - `mv [origem] [destino]`
- O comando **mv** também é usado para renomear arquivos
- Opções suportadas que funcionam da mesma forma que o comando **cp**: **-i**, **-n** e **-v**



# Criando arquivos

- Editores podem ser usados para criar arquivos (discutidos posteriormente neste curso)
- Use o comando `touch` para criar um arquivo vazio
- O comando `touch` também atualiza o registro de data e hora de modificação de um arquivo



# Removendo arquivos

- O comando **rm** é usado para excluir arquivos
- A exclusão de arquivos é permanente!
- Use **-i** para evitar exclusão acidental ao usar globs
- Exclua os diretórios com a opção **-r** ou **rmdir** se o diretório estiver completamente vazio





# Fazendo diretórios

- Use the `mkdir` command to create directories

