

LAPORAN
UJIAN AKHIR SEMESTER
SISTEM OPERASI



Nama : Farhan Mawaludin
NIM : 2341720258
Kelas : 1B
Prodi : D-IV Teknik Informatika

JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG
2024

Deskripsi Program

```
=====
                  Selamat Datang di Manajemen File
=====
===== Dibuat Oleh =====
                  Farhan Mawaludin
                  2341720258
===== Kelas TI-1B =====
===== Menu =====
1. Lihat Isi Direktori
2. Buat Direktori
3. Pindah Direktori
4. Lihat Path Saat Ini
5. Lihat Isi File
6. Salin Berkas
7. Ubah Nama File
8. Hapus Berkas
9. Pindahkan File
10. Lihat Pengguna
11. Keluar
Pilih menu: █
```

Manajemen file adalah program yang berfungsi untuk mengelola sebuah file dan folder. Didalam program manajemen file diatas terdapat 11 fitur yaitu:

1. Lihat isi direktori

Menampilkan daftar file dan direktori beserta informasi detail seperti permissions, owner, size, dan modification time menggunakan perintah `ls -l`. Pengguna dapat melihat isi dari direktori yang sedang aktif.

2. Buat direktori

Membuat direktori baru dengan nama yang dimasukkan oleh pengguna menggunakan perintah `mkdir -p`. Fitur ini memastikan direktori baru dibuat sesuai dengan input pengguna.

3. Pindah direktori

Mengubah direktori kerja saat ini ke direktori yang ditentukan oleh pengguna menggunakan perintah `cd`. Jika direktori tujuan tidak valid, program akan menampilkan pesan kesalahan.

4. Lihat path saat ini

Menampilkan path direktori kerja saat ini menggunakan perintah `pwd`. Ini membantu pengguna mengetahui lokasi direktori tempat mereka bekerja.

5. Lihat isi file

Menampilkan isi file yang dipilih oleh pengguna menggunakan perintah `cat`. Program ini memeriksa keberadaan file sebelum membukanya, dan memberikan pesan kesalahan jika file tidak ditemukan.

6. Salin berkas

Menyalin file dari lokasi sumber ke lokasi tujuan yang ditentukan oleh pengguna menggunakan perintah cp. Pengguna dapat menduplikasi file dengan mudah.

7. Ubah nama file

Mengubah nama file dari nama lama ke nama baru sesuai input pengguna menggunakan perintah mv. Fitur ini memungkinkan pengguna mengganti nama file sesuai kebutuhan.

8. Hapus berkas

Menghapus file yang ditentukan oleh pengguna menggunakan perintah rm. Program ini memastikan file dihapus sesuai dengan input pengguna.

9. Pindahkan file

Memindahkan file dari lokasi sumber ke lokasi tujuan yang ditentukan oleh pengguna menggunakan perintah mv. File akan dipindahkan sepenuhnya dari sumber ke tujuan.

10. Lihat pengguna

Menampilkan daftar pengguna yang sedang login ke sistem menggunakan perintah who. Ini berguna untuk mengetahui siapa saja yang sedang aktif di sistem.

11. Keluar

Menampilkan pesan jika berhasil keluar.

Penjelasan Program

1. Lihat isi direktori

```
# Menampilkan isi direktori saat ini
lihat_isi_direktori() {
    ls -l
}
```

Pada file ini terdapat program untuk melihat daftar isi dari sebuah direktori/folder menggunakan fungsi `lihat_isi_direktori`. Sebelum kondisi dijalankan, user akan langsung melihat daftar isi dari direktori saat ini tanpa harus memasukkan path direktori atau lokasi direktori yang akan dicek. Fungsi `lihat_isi_direktori` langsung menjalankan perintah `ls -l` yang berfungsi untuk melihat daftar isi dari direktori yang sedang aktif secara mendetail. Informasi yang ditampilkan meliputi permission, jumlah hardlink (mengacu pada elemen induknya), owner, group, size, last modification time, dan directory name. Fitur ini memudahkan user untuk mendapatkan informasi lengkap mengenai isi direktori saat ini tanpa perlu memasukkan input tambahan.

2. Buat direktori

```
buat_direktori() {
    read -p "Masukkan nama direktori baru: " direktori
    mkdir -p "$direktori"
    echo "Direktori $direktori berhasil dibuat."
}
```

Pada file ini terdapat program untuk membuat sebuah direktori baru menggunakan fungsi `buat_direktori`. User akan diminta untuk memasukkan nama direktori baru melalui prompt dengan perintah `read -p "Masukkan nama direktori baru: "` direktori. Setelah user memasukkan nama direktori, program akan menjalankan perintah `mkdir -p "$direktori"` yang berfungsi untuk membuat direktori baru dengan nama yang telah dimasukkan. Opsi `-p` memastikan bahwa semua direktori dalam path yang diberikan akan dibuat jika belum ada. Setelah direktori berhasil dibuat, program akan menampilkan pesan "Direktori \$direktori berhasil dibuat." untuk mengonfirmasi bahwa direktori baru telah berhasil dibuat. Fitur ini memudahkan user dalam membuat direktori baru dengan cepat dan mudah.

3. Pindah direktori

```
# Pindah ke direktori yang diinginkan
pindah_direktori() {
    read -p "Masukkan direktori tujuan: " direktori
    cd "$direktori" || { echo "Gagal pindah ke direktori $direktori"; return; }
    echo "Direktori saat ini: $(pwd)"
}
```

Fungsi `pindah_direktori()` merupakan sebuah program sederhana yang memungkinkan pengguna untuk secara interaktif memilih dan pindah ke direktori tujuan di dalam sistem file. Saat program dijalankan, pengguna diminta untuk memasukkan direktori tujuan yang diinginkan. Setelah menerima input dari pengguna, program akan mencoba untuk berpindah ke direktori tersebut menggunakan perintah `cd`. Jika perpindahan berhasil, program akan memberikan umpan balik kepada pengguna dengan menampilkan pesan yang menyertakan direktori saat ini yang baru. Namun, jika perintah `cd` gagal karena alasan tertentu, seperti direktori tujuan tidak ada atau tidak dapat diakses, program akan memberikan pesan kesalahan yang sesuai kepada pengguna dan tetap mempertahankan direktori saat ini tanpa melakukan perubahan. Dengan demikian, fungsi ini menyediakan pengalaman interaktif yang sederhana namun berguna untuk navigasi direktori dalam sistem file.

4. Lihat path saat ini

```
# Menampilkan path saat ini
lihat_path_saat_ini() {
    pwd
}
```

Fungsi `lihat_path_saat_ini()` adalah sebuah program yang bertujuan untuk memberikan informasi kepada pengguna mengenai lokasi atau path dari direktori saat ini di dalam sistem file. Dalam implementasinya, program menggunakan perintah `pwd`, yang secara spesifik menampilkan path lengkap dari direktori saat ini di terminal atau lingkungan kerja yang sedang digunakan. Dengan demikian, fungsi ini memberikan pengguna kemampuan untuk dengan cepat dan mudah memeriksa direktori mana yang sedang mereka akses dalam sistem file, yang berguna untuk navigasi atau referensi saat bekerja dengan file dan direktori.

5. Lihat isi file

```
# Melihat isi file
lihat_isi_file() {
    read -p "Masukkan nama file: " file
    if [ -f "$file" ]; then
        cat "$file"
    else
        echo "File $file tidak ditemukan."
    fi
}
```

Fungsi `lihat_isi_file()` memberikan pengguna kemampuan untuk melihat isi dari sebuah file yang mereka tentukan. Program meminta pengguna untuk memasukkan nama file yang ingin dilihat isinya. Setelah menerima input, program memeriksa apakah file tersebut ada di dalam sistem file. Jika file ditemukan, program akan menampilkan seluruh isi dari file tersebut di terminal menggunakan perintah `cat`. Namun, jika file tidak ditemukan, program akan memberikan umpan balik kepada pengguna bahwa file yang dimaksud tidak ditemukan. Dengan demikian, fungsi ini memberikan cara yang cepat dan mudah bagi pengguna untuk memeriksa konten dari sebuah file di dalam sistem file mereka, yang berguna dalam proses pembacaan atau verifikasi isi file secara efisien.

6. Salin berkas

```
# Menyalin berkas
salin_berkas() {
    read -p "Masukkan nama file sumber: " sumber
    read -p "Masukkan nama file tujuan: " tujuan
    cp "$sumber" "$tujuan"
    echo "Berkas $sumber berhasil disalin ke $tujuan."
}
```

Fungsi `salin_berkas()` mengizinkan pengguna untuk menyalin sebuah file dari lokasi sumber ke lokasi tujuan yang mereka tentukan. Program ini memulai dengan meminta pengguna untuk memasukkan nama file sumber yang ingin mereka salin, serta nama file tujuan di mana mereka ingin menyalinnya. Setelah menerima kedua input tersebut, program menggunakan perintah `cp` untuk menyalin file dari sumber ke tujuan yang ditentukan oleh pengguna. Jika proses penyalinan berhasil, program memberikan umpan balik kepada pengguna dengan pesan yang menyatakan bahwa berkas sumber telah berhasil disalin ke lokasi tujuan yang ditentukan. Dengan demikian, fungsi ini memberikan pengguna kemampuan untuk dengan mudah menyalin berkas antara lokasi yang ditentukan dalam sistem file mereka, yang dapat berguna dalam pengelolaan dan organisasi berkas.

7. Ubah nama berkas

```
# Mengubah nama berkas
ubah_nama_berkas() {
    read -p "Masukkan nama file lama: " lama
    read -p "Masukkan nama file baru: " baru
    mv "$lama" "$baru"
    echo "Berkas $lama berhasil diubah namanya menjadi $baru."
}
```

Fungsi `ubah_nama_berkas()` adalah sebuah program yang memungkinkan pengguna untuk mengubah nama dari sebuah berkas di dalam sistem file. Program ini meminta pengguna untuk memasukkan nama berkas yang ingin mereka ubah namanya, serta nama baru yang diinginkan. Setelah menerima kedua input tersebut, program menggunakan perintah `mv` untuk melakukan pengubahan nama berkas dari nama lama

menjadi nama baru. Jika proses pengubahan nama berhasil dilakukan, program memberikan umpan balik kepada pengguna dengan menampilkan pesan yang menyatakan bahwa berkas lama telah berhasil diubah namanya menjadi nama baru yang ditentukan. Dengan demikian, fungsi ini memberikan cara yang cepat dan mudah bagi pengguna untuk mengubah nama sebuah berkas di dalam sistem file mereka.

8. Hapus berkas

```
# Menghapus berkas
hapus_berkas() {
    read -p "Masukkan nama file yang akan dihapus: " file
    rm "$file"
    echo "Berkas $file berhasil dihapus."
}
```

Fungsi `hapus_berkas()` memberikan pengguna kemampuan untuk menghapus sebuah berkas dari sistem file. Program meminta pengguna untuk memasukkan nama berkas yang ingin dihapus. Setelah menerima input, program akan mencoba menghapus berkas tersebut menggunakan perintah `rm`. Jika berkas berhasil dihapus, program akan memberikan umpan balik kepada pengguna dengan menampilkan pesan bahwa berkas tersebut berhasil dihapus. Namun, jika berkas tidak dapat dihapus karena alasan apapun, seperti izin akses yang tidak mencukupi atau berkas tidak ada, program tidak memberikan penanganan khusus untuk situasi tersebut. Dengan demikian, fungsi ini memberikan cara yang sederhana namun efektif bagi pengguna untuk menghapus sebuah berkas dari sistem file mereka, dengan memberikan umpan balik yang sesuai setelah proses penghapusan selesai.

9. Pindahkan file

```
# Memindahkan file
pindahkan_file() {
    read -p "Masukkan nama file sumber: " sumber
    read -p "Masukkan nama file tujuan: " tujuan
    mv "$sumber" "$tujuan"
    echo "Berkas $sumber berhasil dipindahkan ke $tujuan."
}
```

Fungsi `pindahkan_file()` menyediakan pengguna dengan kemampuan untuk memindahkan sebuah file dari lokasi sumber ke lokasi tujuan yang mereka tentukan. Pertama, pengguna diminta untuk memasukkan nama file sumber yang ingin dipindahkan, dan kemudian diminta untuk memasukkan nama file tujuan tempat mereka ingin memindahkan file tersebut. Setelah menerima kedua input tersebut, program menggunakan perintah `mv` untuk memindahkan file dari lokasi sumber ke lokasi tujuan yang ditentukan. Jika operasi pemindahan berhasil, program memberikan umpan balik kepada pengguna dengan menampilkan pesan bahwa berkas telah berhasil dipindahkan dari lokasi sumber ke lokasi tujuan. Dengan demikian, fungsi ini memberikan pengguna dengan cara yang cepat dan mudah untuk melakukan pemindahan file di dalam sistem file mereka.

10. Lihat pengguna

```
# Melihat pengguna yang sedang login
lihat_pengguna() {
    who
}
```

Fungsi `lihat_pengguna()` dirancang untuk memberikan informasi kepada pengguna tentang pengguna yang sedang masuk atau aktif dalam sistem pada saat itu. Program ini secara langsung menggunakan perintah `who`, yang bertugas untuk menampilkan daftar pengguna yang saat ini login ke dalam sistem, beserta informasi tambahan seperti nama pengguna, terminal yang digunakan, waktu masuk, dan alamat IP. Dengan menggunakan perintah ini, fungsi menyediakan cara yang cepat dan mudah bagi pengguna untuk memeriksa siapa saja yang sedang menggunakan sistem pada saat tertentu, yang dapat bermanfaat dalam situasi seperti memantau aktivitas pengguna, mengidentifikasi sesi yang aktif, atau hanya sekedar mengetahui siapa saja yang sedang login ke dalam sistem pada saat tersebut.

11. Keluar

```
# Jika pilihan adalah 11 (Keluar), maka keluar dari loop
if [ "$pilihan" -eq 11 ]; then
    break
fi
```

Kondisi `if ["$pilihan" -eq 11]; then break fi` dalam sebuah skrip shell digunakan untuk menghentikan atau keluar dari loop ketika variabel pilihan bernilai 11. Program memeriksa apakah nilai pilihan sama dengan 11 menggunakan operator `-eq`. Jika benar, perintah `break` dijalankan untuk keluar dari loop. Jika nilai pilihan bukan 11, blok `if` diabaikan dan loop terus berjalan. Bagian kode ini menyediakan mekanisme untuk keluar dari loop berdasarkan nilai tertentu yang diinput oleh pengguna atau ditentukan oleh program.

Pengujian Program

1. Lihat isi direktori

```
Pilih menu: 1
total 28
-rw-r--r-- 1 fqrhqn fqrhqn    0 Mei  6 17:20 coba1
-rw----- 1 fqrhqn fqrhqn   11 Mei  6 17:45 coba.sh
-rw-rw-r-- 1 fqrhqn fqrhqn    0 Mei  6 13:47 farhan
-rw-r----- 1 fqrhqn fqrhqn    0 Mei  6 13:48 farhanbaru
-rw----- 1 fqrhqn fqrhqn    0 Mei  6 17:49 'farhan mawaludin'
-r--r--r-- 1 fqrhqn fqrhqn    0 Mei  6 17:28 file1
-r--r--r-- 1 fqrhqn fqrhqn    0 Mei  6 17:30 file22
-r--r--r-x 1 fqrhqn fqrhqn    0 Mei  6 17:32 file3
-rw-r--r-- 1 fqrhqn fqrhqn   42 Mei  6 18:01 hello.txt
-rwxr-xr-x 1 fqrhqn fqrhqn 3230 Mei 27 18:31 manajemen.sh
drwxr-x--- 2 fqrhqn fqrhqn 4096 Mei  6 13:48 mydir
-rw-r--r-- 1 fqrhqn fqrhqn   12 Mei  6 17:58 world2.txt
-rw-rw-rw- 1 fqrhqn fqrhqn   12 Mei  6 17:57 world.txt
drwx----- 2 fqrhqn fqrhqn 4096 Mei  6 13:49 xdir
-rw----- 1 fqrhqn fqrhqn    0 Mei  6 13:49 xfiles
```

Menampilkan daftar semua file dan direktori dalam direktori saat ini dengan rincian detail (misalnya, hak akses, ukuran, tanggal modifikasi) menggunakan perintah `ls -l`.

2. Buat direktori

```
Pilih menu: 2
Masukkan nama direktori baru: test
Direktori test berhasil dibuat.
```

Meminta pengguna untuk memasukkan nama direktori baru yang ingin dibuat, kemudian membuat direktori tersebut menggunakan perintah `mkdir -p`. Setelah pembuatan, program menampilkan pesan konfirmasi.

3. Pindah direktori

```
Pilih menu: 3
Masukkan direktori tujuan: xdir
Direktori saat ini: /home/fqrhqn/sisop/P12/xdir
```

Meminta pengguna untuk memasukkan nama direktori tujuan, lalu mencoba untuk pindah ke direktori tersebut menggunakan perintah `cd`. Jika gagal, program menampilkan pesan kesalahan; jika berhasil, menampilkan path direktori saat ini.

4. Lihat path saat ini

```
Pilih menu: 4
/home/fqrhqn/sisop/P12
```

Menampilkan path direktori saat ini menggunakan perintah `pwd`.

5. Lihat isi file

```
Pilih menu: 5
Masukkan nama file: hello.txt
NAMA : FARHAN MAWALUDIN
NIM  : 2341720258
```


Meminta pengguna untuk memasukkan nama file yang ingin dilihat. Jika file ada, program menampilkan isi file menggunakan perintah cat. Jika tidak, menampilkan pesan kesalahan.

6. Salin berkas

```
Pilih menu: 6
Masukkan nama file sumber: hello.txt
Masukkan nama file tujuan: xdir
Berkas hello.txt berhasil disalin ke xdir.
```

Meminta pengguna untuk memasukkan nama file sumber dan tujuan, lalu menyalin file dari sumber ke tujuan menggunakan perintah cp. Program menampilkan pesan konfirmasi setelah salinan selesai.

7. Ubah nama file

```
Pilih menu: 7
Masukkan nama file lama: farhan
Masukkan nama file baru: newfarhan
Berkas farhan berhasil diubah namanya menjadi newfarhan.
```

Meminta pengguna untuk memasukkan nama file lama dan nama file baru, kemudian mengubah nama file menggunakan perintah mv. Program menampilkan pesan konfirmasi setelah perubahan nama selesai.

8. Hapus berkas

```
Pilih menu: 8
Masukkan nama file yang akan dihapus: newfarhan
Berkas newfarhan berhasil dihapus.
```

Meminta pengguna untuk memasukkan nama file yang ingin dihapus, lalu menghapus file tersebut menggunakan perintah rm. Program menampilkan pesan konfirmasi setelah penghapusan selesai.

9. Pindahkan file

```
Pilih menu: 9
Masukkan nama file sumber: file1
Masukkan nama file tujuan: file3
mv: replace 'file3', overriding mode 0445 (r--r--r-x)?
Berkas file1 berhasil dipindahkan ke file3.
```

Meminta pengguna untuk memasukkan nama file sumber dan tujuan, kemudian memindahkan file dari sumber ke tujuan menggunakan perintah mv. Program menampilkan pesan konfirmasi setelah pemindahan selesai.

10. Lihat pengguna

```
Pilih menu: 10
fqrhqn tty2 2024-06-03 11:09 (tty2)
```

Menampilkan daftar pengguna yang saat ini login ke sistem menggunakan perintah who.

11. Keluar

```
Pilih menu: 11  
Keluar dari program.
```

Menampilkan Output keluar dari program jika tidak ingin menjalankan manajemen file.