



## Bermain-main dengan Perintah Dasar Linux

### 4.1 Perintah Dasar yang Berhubungan dengan Direktori.

- **ls**

Menampilkan isi dari suatu direktori. Perintah ini bisa berdiri sendiri ataupun dijalankan dengan argument.

Contoh option yang bisa anda sertakan adalah:

OPTION	FUNGSI
-a	bila anda ingin menampilkan semua file dan folder, termasuk file dan folder yang tersembunyi
-A	sama dengan -a, tetapi tidak menampilkan direktori . dan ..
-C	menampilkan direktori dengan output berbentuk kolom
-d	menampilkan direktori saja, isi direktori tidak ditampilkan
-f	menampilkan isi direktori tanpa diurutkan
-l	menampilkan isi direktori secara lengkap, mulai dari hak akses, owner, group dan tanggal file atau direktori tersebut dibuat
-l (satu)	menampilkan isi direktori dengan format satu direktori per baris

Contoh :

```
dhanie@localhost:~$ ls -l
total 28
drwxr-xr-x  2 dhania dhania 4096 2008-04-10 21:46 Desktop
drwxr-xr-x 21 dhania dhania 4096 2008-04-22 20:22 Documents
drwxr-xr-x  3 dhania dhania 4096 2008-04-22 20:22 Installer
drwxr-xr-x  2 dhania dhania 4096 2008-04-22 20:23 Lyric
drwxr-xr-x  6 dhania dhania 4096 2008-04-22 20:23 Picture
drwxr-xr-x 17 dhania dhania 4096 2008-04-22 20:23 Song
drwxr-xr-x  5 dhania dhania 4096 2008-04-22 20:22 Tutorial
```

```

dhanie@localhost:~$ ls -a
.                .dmrc            .ICEauthority   Picture
..              Documents        Installer       .qt
.alexandria     .fontconfig     .kde            .recently-used
.bash_history   .gcompris       .kderc          Song
.bash_logout    .gconf          .lessht        .superkaramba
.bash_profile   .gconfd         .local          .thumbnails
.bashrc         .gimp-2.2       Lyric           Tutorial
.bluefish       .gksu.lock      .mc             .update-notifier
.childsplay     .gnome          .mcp            .viminfo
.childsplay.score .gnome2         .mcpoprc       .Xauthority
.config         .gnome2_private .metacity       .xine
.DCOPserver_localhost__0 .gnucash       .mozilla       .xsession-errors
.DCOPserver_localhost_:0 .gststreamer-0.10 .nautilus
Desktop         .gtkrc-1.2-gnome2 .openoffice.org2

```

```

dhanie@localhost:~$ ls
Desktop Documents Installer Lyric Picture Song Tutorial
dhanie@localhost:~$ ls -l
Desktop
Documents
Installer
Lyric
Picture
Song
Tutorial

```

Masih banyak lagi option yang bisa anda berikan pada perintah dasar. Untuk mendapatkan informasi lengkapnya anda bisa membaca di manual. Masih ingat bagaimana cara membaca manual?

Selain option anda bisa memberikan argument berupa pathname. Yaitu direktori yang ingin anda lihat isinya.

Contoh :

```

dhanie@localhost:~$ ls /home/dhanie/Documents/
adsense          desain           kurikulum        quanta referensi
artikel penting Dhanie Ngeblog  Lamaran         template joomla 1.5
backup           dhoho           log install lamm tutorial
bayes_referensi Ebook            log install lamm~ zencart
Bikin modul     instalasi phpb  my blog planing
config.php      install zencart mysql_errrr.png
Demi Cinta     klas            OpenOfficeBase

```

```

dhanie@localhost:~$ ls -l /home/dhanie/Picture/snapshot_install_joomla/
total 1180
-rw-r--r-- 1 dhanie dhanie 92313 2008-04-08 18:15 snapshot10.png
-rw-r--r-- 1 dhanie dhanie 55623 2008-04-08 19:41 snapshot11.png
-rw-r--r-- 1 dhanie dhanie 81953 2008-04-08 19:42 snapshot12.png
-rw-r--r-- 1 dhanie dhanie 97532 2008-04-08 18:05 snapshot1.png
-rw-r--r-- 1 dhanie dhanie 108779 2008-04-08 18:06 snapshot2.png
-rw-r--r-- 1 dhanie dhanie 86914 2008-04-08 18:06 snapshot3.png
-rw-r--r-- 1 dhanie dhanie 131466 2008-04-08 18:07 snapshot4.png
-rw-r--r-- 1 dhanie dhanie 94798 2008-04-08 18:10 snapshot5.png
-rw-r--r-- 1 dhanie dhanie 108118 2008-04-08 18:12 snapshot6.png
-rw-r--r-- 1 dhanie dhanie 107936 2008-04-08 18:13 snapshot7.png
-rw-r--r-- 1 dhanie dhanie 139788 2008-04-08 18:14 snapshot8.png
-rw-r--r-- 1 dhanie dhanie 27579 2008-04-08 18:14 snapshot9.png

```

- **dir**

Memiliki fungsi yang sama dengan perintah `ls`, yaitu menampilkan isi direktori. Anda bisa membuka manual dari perintah `dir`. Pemberian option dan argument sama dengan perintah `ls`.

- **pwd (print working directory)**

Menampilkan direktori yang sedang aktif (current directory). Perintah ini tidak mempunyai option dan argumen.

Contoh :

```

dhanie@localhost:~$ pwd
/home/dhanie

```

- **mkdir**

Perintah untuk membuat direktori. Untuk mencoba perintah ini ikuti latihan dibawah ini. Pastikan direktori aktif anda saat ini di `/home`, di komputer saya saya berada di `/home/dhanie`. Dan anda saat ini belum mempunyai direktori yang bernama `dataku`.

Buat direktori dengan nama `dataku`, ketikkan

```

dhanie@localhost:~$ mkdir dataku

```

untuk melihat hasilnya lihat dengan perintah `ls`

```

dhanie@localhost:~$ ls
dataku Desktop Documents Installer Lyric Picture Song Tutorial

```

sekarang anda sudah memiliki direktori baru dengan nama `dataku`.

Bila anda sudah mempunyai direktori dengan nama `dataku` sebelumnya, maka saat anda membuat lagi dengan nama yang sama akan terjadi error.

```
dhanie@localhost:~$ mkdir dataku
mkdir: cannot create directory `dataku': File exists
```

Bersabarlah :-) dan ikuti langkah demi langkah latihan ini, maka anda akan memahami perintah dasar yang berhubungan dengan direktori. Next.....

- **cd**

Perintah untuk berpindah direktori aktif. Coba anda berpindah direktori ke `dataku`.

Contoh :

```
dhanie@localhost:~$ cd dataku
dhanie@localhost:~/dataku$ █
```

Sebelumnya saya berada di direktori aktif `/home/dhanie` kemudian saya ketikkan perintah untuk berpindah ke dalam direktori `dataku` yang sudah saya buat tadi. Direktori ini berada di dalam direktori

`/home/dhanie` jadi saya tidak perlu menuliskan `cd /home/dhanie/dataku` secara lengkap. Perhatikan, tulisan pada prompt berubah. Disitu menunjukkan kalau direktori aktif berubah ke `dataku`.

Bila sudah berpindah direktori aktif, coba lihat direktori aktif yang baru dengan perintah `pwd` diatas.

```
dhanie@localhost:~/dataku$ pwd
/home/dhanie/dataku
```

saat ini direktori aktif anda adalah `dataku`. Coba buat direktori baru didalam `dataku`. Masih ingat dengan perintah `mkdir` kan?

Buat direktori dengan nama `picture`, `song`, `video`, `document`

```
dhanie@localhost:~/dataku$ mkdir picture
dhanie@localhost:~/dataku$ mkdir song
dhanie@localhost:~/dataku$ mkdir video
dhanie@localhost:~/dataku$ mkdir document
```

Kemudian cek dengan perintah `ls`, bila hasilnya seperti dibawah ini, berarti anda berada di jalan yang benar :-)

```
dhanie@localhost:~/dataku$ ls
document picture song video
```

Sekarang coba anda berpindah direktori ke `picture`.

```
dhanie@localhost:~/dataku$ cd picture
dhanie@localhost:~/dataku/picture$
```

Sudah yakin berada di direktory yang benar? chek.. chek.. chek.. :-)

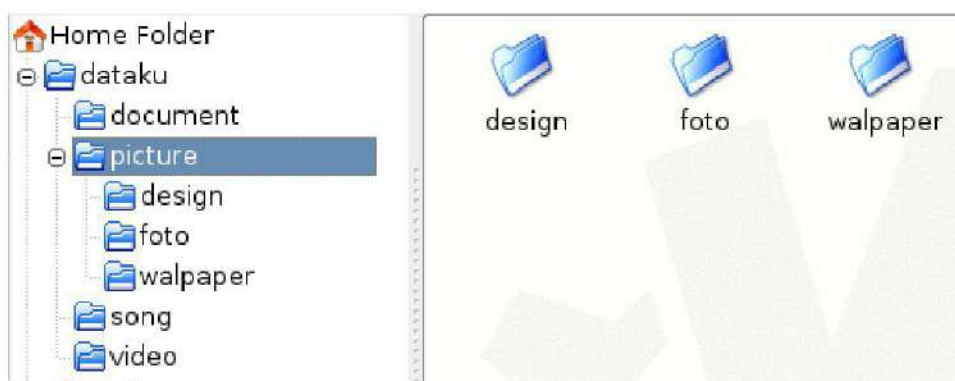
Sekarang coba anda ketikkan perintah dibawah ini dalam satu baris, kemudian tekan tombol Enter.

```
mkdir walpaper; mkdir foto; mkdir design
```

```
dhanie@localhost:~/dataku/picture$ mkdir walpaper; mkdir foto; mkdir design
```

Nah anda baru saja menjalankan tiga perintah sekaligus dalam satu baris perintah. Hal ini dimungkinkan di Linux. Anda bisa memberikan beberapa perintah sekaligus dengan memberikan tanda pemisah berupa **; (titik koma)**. Dalam hal ini, bekerja di mode teks relatif lebih mudah dan cepat daripada menggunakan GUI (mode grafik).

Oh iya.. anda bisa mengecek hasil direktori yang anda buat dari mode teks di mode grafik. Di file browser (saya menggunakan Kongueror) coba anda lihat di home folder.



Gambar 4.1 Ini adalah hasil direktori yang anda buat

Untuk kembali ke Direktori sebelumnya (satu tingkat di atasnya, sama dengan perintah **Up** di mode grafik) anda bisa mengetikkan perintah

```
dhanie@localhost:~/dataku/picture$ cd ..
```

```
dhanie@localhost:~/dataku/picture$ cd ..  
dhanie@localhost:~/dataku$ █
```

- **rmdir**

Perintah untuk menghapus direktori kosong.

Untuk keperluan mencoba menghapus direktori, kita harus membuat direktori lagi. Ayo.. belajar.. belajar.. jadi anda sebaiknya tidak malas mencoba lagi dan lagi ;-)

Buat dua direktori baru di dalam `dataku`. Beri nama `test`, dan `coba`.

```
dhanie@localhost:~/dataku$ mkdir test; mkdir coba
```

```
dhanie@localhost:~/dataku$ ls  
coba document picture song test video
```

Jadi didalam folder `dataku`, anda menambahkan dua direktori baru yaitu `test` dan `coba`.

Kemudian berpindahlah ke direktori `test`. Tentu anda masih ingat cara berpindah direktori. Betul anda bisa menggunakan `cd test`

Setelah direktori aktif anda berada di `test`, buatlah direktori lagi didalam `test`. Kali ini kita beri nama `a`, `b`, `c`,

Anda bisa melihat gambar dibawah ini.

```
dhanie@localhost:~/dataku$ cd test  
dhanie@localhost:~/dataku/test$ mkdir a; mkdir b; mkdir c  
dhanie@localhost:~/dataku/test$ ls  
a b c
```

Kemudian berpindah direktori lagi, sehingga direktori aktif anda adalah `dataku`.

```
dhanie@localhost:~/dataku/test$ cd ..  
dhanie@localhost:~/dataku$ █
```

Sekarang saatnya anda mencoba perintah untuk menghapus direktori.

Cobalah untuk menghapus direktori `test` terlebih dahulu.

Amati apa yang terjadi.

```
dhanie@localhost:~/dataku$ rmdir test  
rmdir: test: Directory not empty
```

Disini muncul error karena direktori `test` bukan direktori kosong.

Didalamnya berisi direktori. Sekarang coba hapus direktori `coba`. Perhatikan apa yang terjadi setelah perintah diberikan.

```
dhanie@localhost:~/dataku$ rmdir coba
dhanie@localhost:~/dataku$ ls
document picture song test video
```

perintah `rmdir` sukses menghapus direktori `coba`, karena direktori tersebut kosong alias tidak ada isinya.

Jadi anda harus ingat bila `rmdir` adalah perintah untuk menghapus direktori kosong (empty directory).

Lalu bagaimana cara menghapus direktori yang tidak kosong? Sabar... perintah tersebut pasti anda temukan nanti. Tapi tidak di bagian ini.

## 4.2 Perintah Dasar yang Berhubungan dengan Manajemen File

- **touch**

Perintah untuk mengganti waktu pembuatan suatu file. Tetapi bila file yang anda ketikkan belum ada maka secara otomatis file tersebut akan dibuat. Anda bisa menentukan tipe file yang anda inginkan. Tetapi file yang dibuat dengan perintah ini adalah file kosong (tidak ada isinya).

Contoh :

```
dhanie@localhost:~/dataku$ touch puisigombal.txt
dhanie@localhost:~/dataku$ ls
document picture puisigombal.txt song test video
```

- **cat**

Perintah `cat`, digunakan untuk menampilkan isi file. Biasanya file yang ditampilkan dengan perintah ini adalah file yang bertipe teks. Dan yang pasti bukan file kosong.

Contoh :

```
dhanie@localhost:~/dataku$ cat puisigombal.txt
dhanie@localhost:~/dataku$ █
```

karena file `puisigombal.txt` masih kosong maka tidak tampak apa-apa. Sekarang kita coba mengisi file tersebut dengan perintah **`cat > puisigombal.txt`** (penjelasan mengenai perintah ini akan dijelaskan kemudian).

Ketikkan `cat > puisigombal.txt` lalu tekan tombol Enter, kemudian mulailah mengetikkan isi file teks yang anda inginkan.

Contoh :

```
dhanie@localhost:~/dataku$ cat > puisigombal.txt
Mencintaimu seperti bernafas
Bagaimana mungkin aku menghentikan diriku sendiri
Setiap sel otakku memikirkan dirimu
Setiap sudut hatiku menyimpan rasa untukmu
Kuhembuskan udara cintaku melalui angin
Kusampaikan lukisan rasaku melalui pelangi
Kurapalkan mantra-mantra cintaku dalam setiap do'a
Dan aku tak bisa mengerti
Bagaimana mungkin aku bisa mengubah puisi
Yang benar-benar gombal abis ini... :-)
```

kalau sudah selesai simpan file anda dengan menekan **Ctrl + c**. Selain itu anda juga bisa menekan **Ctrl + z**.

Sekarang anda punya file yang sudah ada isinya dan bisa kita gunakan untuk latihan selanjutnya. Coba gunakan perintah `cat` untuk melihat isi file tersebut.

```
dhanie@localhost:~/dataku$ cat puisigombal.txt
Mencintaimu seperti bernafas
Bagaimana mungkin aku menghentikan diriku sendiri
Setiap sel otakku memikirkan dirimu
Setiap sudut hatiku menyimpan rasa untukmu
Kuhembuskan udara cintaku melalui angin
Kusampaikan lukisan rasaku melalui pelangi
Kurapalkan mantra-mantra cintaku dalam setiap do'a
Dan aku tak bisa mengerti
Bagaimana mungkin aku bisa mengubah puisi
Yang benar-benar gombal abis ini... :-)
```

- **more**

Perintah ini bisa digunakan untuk menampilkan isi file teks dengan tampilan perlayar. Anda akan membutuhkan perintah ini saat anda membaca file yang panjang. Untuk menampilkan layar selanjutnya gunakan tombol spasi. Tetapi dengan `more` anda tidak bisa melihat tampilan di layar yang terdahulu.



- **less**

Memiliki fungsi yang sama dengan more, tetapi anda bisa menampilkan tampilan layar terdahulu dengan menggunakan tombol **panah atas** atau **Page Up**.

- **cp**

Berfungsi untuk mengcopy atau menduplikat file dan direktori.

Format: `cp nama_file_asal nama_file_hasil`

Contoh :

```
dhanie@localhost:~/dataku$ cp puisigombal.txt hasilcopy.txt
dhanie@localhost:~/dataku$ ls
document hasilcopy.txt picture puisigombal.txt song test video
dhanie@localhost:~/dataku$ more hasilcopy.txt
Mencintaimu seperti bernafas
Bagaimana mungkin aku menghentikan diriku sendiri
Setiap sel otakku memikirkan dirimu
Setiap sudut hatiku menyimpan rasa untukmu
Kuhembuskan udara cintaku melalui angin
Kusampaikan lukisan rasaku melalui pelangi
Kurapalkan mantra-mantra cintaku dalam setiap do'a
Dan aku tak bisa mengerti
Bagaimana mungkin aku bisa menggubah puisi
Yang benar-benar gombal abis ini... :-)
```

Gambar diatas menunjukkan proses mengcopy puisigombal.txt menjadi hasilcopy.txt kemudian isi dari hasilcopy.txt ditampilkan dengan perintah more.

```
dhanie@localhost:~/dataku$ cp puisigombal.txt /home/dhanie/dataku/song
dhanie@localhost:~/dataku$ ls /home/dhanie/dataku/song
puisigombal.txt
```

Contoh kedua ini kita mengcopy file puisigombal.txt ke folder tertentu, yaitu ke folder `/home/dhanie/dataku/song`

- **mv**

Perintah untuk memindahkan file dan direktori. Perintah ini juga bisa digunakan untuk merename (mengganti) nama file atau direktori.

Format : `mv nama_file nama_file_baru`  
`mv nama_file direktori_tujuan`

Contoh :

```
dhanie@localhost:~/dataku$ mv hasilcopy.txt hasil.txt
dhanie@localhost:~/dataku$ ls
document hasil.txt picture puisigombal.txt song test video
```

Perintah diatas mengganti nama file `hasilcopy.txt` menjadi `hasil.txt`

```
dhanie@localhost:~/dataku$ mv song lagu
dhanie@localhost:~/dataku$ ls
document hasil.txt lagu picture puisigombal.txt test video
```

Perintah diatas mengganti nama direktori `song` menjadi `lagu`

```
dhanie@localhost:~/dataku$ mv /home/dhanie/dataku/document/ /home/dhanie/dataku/
lagu
```

Memindahkan direktori `document` ke direktori `lagu`.

- **rm**

Untuk menghapus (remove) file atau direktori.

Format: `rm nama_file`

Contoh :

```
dhanie@localhost:~/dataku$ rm hasil.txt
dhanie@localhost:~/dataku$ ls
lagu picture puisigombal.txt video
```

Dengan mengetikkan perintah `rm hasil.txt`, seperti contoh diatas, maka file tersebut dihapus dengan sukses.

Apakah anda masih ingat kita pernah mencoba menghapus direktori `test` dengan perintah `rmdir`, tetapi gagal karena direktori tersebut tidak kosong. Sekarang coba hapus direktori `test` dengan perintah `rm`.

```
dhanie@localhost:~/dataku$ rm test
rm: cannot remove `test': Is a directory
```

Loh ternyata kok masih gagal juga ya...\* garuk-garuk kepala \* :-D

Untuk menghapus direktori kosong, anda harus memberikan OPTION. Coba anda lihat manual dari perintah `rm`, kira-kira option yang mana yang harus kita berikan.

Kita harus menambahkan option `-R` yang artinya menghapus secara rekursif, jadi dengan adanya option ini, direktori atau file yang ada di dalam direktori `test` akan dihapus secara rekursif.

```
dhanie@localhost:~/dataku$ rm -R test
dhanie@localhost:~/dataku$ ls
document picture song video
```

- **find**

Mencari suatu file dalam direktori tertentu. Anda bisa melakukan pencarian berdasarkan nama, ukuran, waktu pembuatan file dsb. dengan memberikan option yang anda inginkan.

Format: `find perkiraan_direktori option`

Contoh :

Untuk mencari semua file teks di dalam direktori `/home/dhanie/dataku`. Perhatikan, option yang diberikan untuk melakukan pencarian ini adalah `-name`.

```
dhanie@localhost:~/dataku$ find /home/dhanie/dataku -name *.txt
/home/dhanie/dataku/lagu/puisigombal.txt
/home/dhanie/dataku/lagu/hasil.txt
```

- **which**

Menampilkan lokasi perintah dasar yang anda cari. Perintah ini juga bisa digunakan untuk mencari file program yang bisa dieksekusi.

Format : `which nama_perintah`

Contoh :

Untuk mencari lokasi perintah `mkdir`.

```
dhanie@localhost:~/dataku$ which mkdir
/bin/mkdir
```

Coba anda cari lokasi file eksekusi dari program `kssnapshot`, yaitu aplikasi untuk mengcapture tampilan monitor di Linux.

```
dhanie@localhost:~/dataku$ which kssnapshot
/usr/bin/kssnapshot
```

- **whereis**

Hampir sama dengan `which`, menampilkan lokasi perintah dasar, tetapi dengan `whereis` lokasi file binary, source dan manual juga ditampilkan.

Format: `whereis nama_perintah`

Contoh :

Coba cari lokasi perintah `mkdir` dengan menggunakan `whereis`.

```
dhanie@localhost:~/dataku$ whereis mkdir
mkdir: /bin/mkdir /usr/share/man/man1/mkdir.1.gz /usr/share/man/man2/mkdir.2.gz
```

Coba juga dengan mencari `ksnapshot`. Perhatikan perbedaannya dengan saat anda menggunakan `which`.

```
dhanie@localhost:~/dataku$ whereis ksnapshot
ksnapshot: /usr/bin/ksnapshot /usr/X11R6/bin/ksnapshot /usr/bin/X11/ksnapshot /usr/share/man/man1/ksnapshot.1.gz
```

- **tar**

Untuk mengextract (memekarkan) file yang di kompres dengan menggunakan perintah teks di linux, anda bisa menggunakan perintah ini. File yang bisa diextract dengan menggunakan `tar` adalah file-file yang berekstensi `.tar.gz` atau `.tar.bz2`

Contoh File :

**wordpress-2.3.3.tar.gz**

**xampp-linux-1.6.tar.gz**

**phpBB-3.0.0.tar.bz2**

Format: `tar option nama_file`

Contoh :

**dhanie@localhost:~\$ tar -xzvf wordpress-2.3.3.tar.gz**

Dimana parameter **x** adalah untuk memekarkan file, **z** untuk menyaring file hasil compresian dari format gzip , **v** untuk menampilkan proses sehingga user dapat mengetahui proses yang terjadi, dan **f** adalah ada namafile yang harus diikuti . Untuk yang berekstensi `tar.bz2` ganti option **z** dengan **j**. Anda bisa menghilangkan option yang tidak perlu.

- unzip

Perintah ini digunakan untuk mengekstrak file yang di kompress yang berekstensi .zip

Contoh file :

**zen-cart-v1.3.8a-full-fileset-12112007.zip**

**Joomla\_1.0.13-Stable-Full\_Package.zip**

Format : *unzip option nama\_file*

Contoh :

**dhanie@localhost:~\$ unzip Joomla\_1.0.13-Stable-Full\_Package.zip**

### 4.3 Perintah yang berhubungan dengan Pemrosesan String

String adalah serangkaian karakter. Linux menyediakan beberapa perintah yang dapat digunakan berkaitan dengan proses string, seperti mencari karakter, pengurutan dan lainnya.

- **head**

Perintah ini digunakan untuk menampilkan beberapa baris awal dari isi file. Misalnya ingin menampilkan 8 baris pertama saja. Secara default yang ditampilkan adalah sepuluh (10) baris awal file. Jika ingin menampilkan jumlah baris tertentu, tambahkan option **-nx**, dimana **x** adalah jumlah barisnya.

Format : *head option file\_yang\_ingin\_dilihat*

Contoh :

```
dhanie@localhost:~/dataku$ head /etc/inittab
# /etc/inittab: init(8) configuration.
# $Id: inittab,v 1.91 2002/01/25 13:35:21 miquels Exp $

# The default runlevel.
id:2:initdefault:

# Boot-time system configuration/initialization script.
# This is run first except when booting in emergency (-b) mode.
si::sysinit:/etc/init.d/rcS
```

- **tail**

Menampilkan isi akhir file. Untuk menampilkan beberapa baris terakhir dari isi file gunakan perintah **tail**. Secara default yang ditampilkan adalah sepuluh baris akhir file. Jika ingin menampilkan jumlah baris tertentu, tambahkan option **-nx**, dimana **x** adalah jumlah barisnya.

Format : `tail option file_yang_ingin_dilihat`

Contoh :

```
dhanie@localhost:~/dataku$ tail -n15 /etc/inittab
2:23:respawn:/sbin/getty 38400 tty2
3:23:respawn:/sbin/getty 38400 tty3
4:23:respawn:/sbin/getty 38400 tty4
5:23:respawn:/sbin/getty 38400 tty5
6:23:respawn:/sbin/getty 38400 tty6

# Example how to put a getty on a serial line (for a terminal)
#
#T0:23:respawn:/sbin/getty -L ttyS0 9600 vt100
#T1:23:respawn:/sbin/getty -L ttyS1 9600 vt100

# Example how to put a getty on a modem line.
#
#T3:23:respawn:/sbin/mgetty -x0 -s 57600 ttyS3
```

- **grep**

Anda dapat mencari karakter atau kata yang diinginkan dari sebuah file yang terdiri dari banyak kalimat. Perintah yang digunakan adalah **grep**. Dengan perintah ini pencarian lebih mudah dilakukan.

Format : `grep option karakter/kata`

Contoh :

Buatlah sebuah file teks baru yang bernama `tesfile.txt`. Masih ingat bagaimana cara membuat file? Anda bisa menggunakan perintah

```
cat > tesfile.txt
```

Jangan lupa untuk menekan **Ctrl + z** setelah selesai mengetik isi file.

Berikut ini adalah isi file tersebut :

**Ini adalah isi file.**

**Filenya bernama tesfile.**

**File ini akan dipakai sebagai latihan.**

**Latihan perang? ya bukan lah...**

```
dhanie@localhost:~/dataku$ cat > tesfile.txt
Ini adalah isi file.
Filanya bernama tesfile.
File ini akan dipakai sebagai latihan.
Latihan perang? ya bukan lah...
```

Selanjutnya kita akan mencari kata "bukan" dari file tersebut.

```
dhanie@localhost:~/dataku$ grep bukan tesfile.txt
Latihan perang? ya bukan lah...
```

Baris kata yang mengandung kata "bukan" akan ditampilkan.

- **wc**

Perintah untuk menampilkan jumlah baris, jumlah kata dan ukuran dari sebuah file.

Format : `wc option nama_file`

Contoh :

```
dhanie@localhost:~/dataku$ wc tesfile.txt
 4 18 117 tesfile.txt
```

4 : jumlah baris  
18 : jumlah kata  
117 : jumlah ukuran file

- **sort**

Apabila anda ingin menampilkan isi file teks secara urut. Gunakan perintah ini.

Format : `sort option nama_file`

Contoh :

Untuk latihan berikutnya buat file baru yang berisi daftar nama berikut ini. (Tentu anda sudah hafal cara membuat file teks dengan perintah cat.)



```
dhania@localhost:~/dataku$ cat > daftarnama.txt  
Brad Pitt  
Angelina Jolie  
Drew Barrymore  
Lindsay Lohan  
Pierce Brosnan  
Bryan Adams  
Christina Aguelera  
Rihana
```

Isi file tersebut akan diurutkan dengan perintah `sort` :

```
dhania@localhost:~/dataku$ sort daftarnama.txt  
Angelina Jolie  
Brad Pitt  
Bryan Adams  
Christina Aguelera  
Drew Barrymore  
Lindsay Lohan  
Pierce Brosnan
```

## 4.4 Perintah-perintah yang berhubungan dengan informasi system

- **uname**

Perintah ini akan menampilkan informasi system komputer anda, antara lain tipe mesin komputer, hostname, nama dan versi sistem operasi dan tipe prosesor. Informasi yang anda dapatkan sesuai dengan option yang anda berikan.

Format : `uname option`

Option yang bisa ditambahkan adalah sebagai berikut.

OPTION	FUNGSI
-a, -all	menampilkan semua informasi
-m, -machine	menampilkan tipe mesin/perangkat keras
-n, -nodename	menampilkan hostname
-r, -release	menampilkan rilis dari kernel sistem operasi
-s, -o	menampilkan nama sistem operasi
-p, --processor	menampilkan tipe prosesor
-v	menampilkan versi sistem operasi

Contoh :

```
dhanie@localhost:~/dataku$ uname -mnr  
localhost 2.6.18-6-686 i686_GNU/Linux
```

- **date**

Perintah untuk menampilkan tanggal dan waktu system

Contoh :

```
dhanie@localhost:~/dataku$ date  
Fri May 2 08:40:30 WIT 2008
```

- **cal**

Untuk menampilkan kalender.

Contoh :

```
dhanie@localhost:~/dataku$ cal
      May 2008
Su Mo Tu We Th Fr Sa
                1  2  3
 4  5  6  7  8  9 10
11 12 13 14 15 16 17
18 19 20 21 22 23 24
25 26 27 28 29 30 31
```

- **df**

Perintah untuk menampilkan penggunaan space filesystem dari hardisk anda.

Contoh :

```
dhanie@localhost:~/dataku$ df
Filesystem      1K-blocks      Used Available Use% Mounted on
/dev/hda1        14421344    3574864  10113920  27% /
tmpfs            225924         0     225924    0% /lib/init/rw
udev            10240          60      10180    1% /dev
tmpfs            225924         0     225924    0% /dev/shm
/dev/hda2       23070676    2123188  19775548  10% /home
/dev/sda1       1015904     881248   134656   87% /media/DHANIE_MNZ
```

- **du**

Perintah untuk menampilkan ukuran direktori atau file.

Contoh :

```
dhanie@localhost:~/dataku$ du -h
4.0K  ./picture/walpaper
4.0K  ./picture/foto
4.0K  ./picture/design
16K   ./picture
4.0K  ./video
4.0K  ./document
12K   ./lagu
40K   .
```

- **uptime**

Untuk mengetahui informasi tentang lama sistem berjalan setelah terakhir reboot atau mati. Informasi yang diberikan adalah waktu sekarang, lama sistem telah berjalan, berapa jumlah user yang login, dan load sistem sejak 1 menit, 5 menit dan 15 menit terakhir.

Contoh :

```
dhanie@localhost:~/dataku$ uptime
08:59:22 up 14:03,  2 users,  load average: 0.20, 0.12, 0.12
```

Tampak informasi komputer yang saya pakai, saat saya menjalankan perintah uptime adalah jam 08:59:22, komputer saya sudah saya pakai selama 14 jam 03 menit, dan seterusnya.

- **hostname**

Perintah untuk menampilkan nama dari komputer (hostname) Contoh

:

```
dhanie@localhost:~$ hostname
localhost
```

- **free**

Perintah untuk menampilkan penggunaan memori

Contoh :

```
dhanie@localhost:~$ free
              total        used         free       shared    buffers     cached
Mem:          451848        445956         5892           0         1796     139284
-/+ buffers/cache:  304876        146972
Swap:          987988           1040        986948
```

- **ps**

Perintah 'ps' merupakan akronim dari "process status". Akan memberikan informasi status proses pada sistem kita. Menampilkan berbagai informasi mengenai proses apa saja yang sedang aktif, siapa pemilik proses tersebut, berapa lama proses berjalan dan berapa ID atau nomor pengenal proses. Anda bisa membuka manual bila ingin mengetahui option yang diberikan.

Contoh :

```
dhanie@localhost:~$ ps
  PID TTY          TIME CMD
 6049 pts/1    00:00:00 bash
 6067 pts/1    00:00:00 ps
```

- **pstree**

Perintah ini fungsinya sama dengan perintah ps, tetapi ditampilkan dalam bentuk pohon. Sebenarnya di Linux tidak ada proses yang berdiri sendiri.

Contoh :

```
dhanie@localhost:~$ pstree
init--NetworkManager--{NetworkManager}
    |
    |--NetworkManagerD
    |--acpid
    |--atd
    |--avahi-daemon--avahi-daemon
    |--bonobo-activati
    |--cron
    |--cupsd
    |--2*[dbus-daemon]
    |--dbus-launch
    |--dcopserver
    |--dhcdbd
    |--dirmngr
    |--events/0
    |--exim4
    |--gconfd-2
    |--gdm--gdm--Xorg
    |           |
    |           |--startkde--kwrapper
    |           |           |--ssh-agent
    |
    |--6*[getty]
    |--hald--hald-runner--hald-addon-acpi
```

## 4.5 Perintah dasar yang berhubungan dengan User

Di Linux ada dua tipe user yang penting untuk diketahui. Kedua user itu adalah user biasa dan user root.

**user root** : user yang memiliki hak sebagai administrator, biasa juga disebut "super user". User root yang akan mengelola dan mengkonfigurasi komputer.

**user biasa**: user yang tidak memiliki hak akses sebagai administrator. User ini digunakan untuk melakukan kegiatan sehari-hari itu misalnya mengetik, browsing internet dan kegiatan lain yang tidak membutuhkan hak root.

Selain itu masih ada juga user system, misalnya **bin** dan **nobody**. User-user ini dibuat otomatis saat kita menginstall Linux.

- **who**

Perintah ini digunakan untuk menampilkan user yang sedang login saat ini. Informasi yang tampak adalah nama user, di terminal (pts) berapa user tersebut berada dan waktu loginnya.

Contoh :

```
dhanie@localhost:~$ who
dhanie  :0          2008-05-02 10:29
dhanie  pts/1       2008-05-02 10:42 (:0.0)
dhanie  pts/2       2008-05-02 10:59 (:0.0)
dhanie  pts/3       2008-05-02 13:53 (:0.0)
```

- **whoami**

Bila anda ingin menampilkan user yang sedang aktif anda bisa menggunakan perintah ini.

Contoh :

```
dhanie@localhost:~$ whoami
dhanie
```

- **id**

Gunakan perintah `id` untuk menampilkan identitas user. User yang ingin ditampilkan identitasnya adalah user yang sedang aktif (login).

Format : `id option`

Contoh :

```
dhanie@localhost:~$ id
uid=1000(dhanie) gid=1000(dhanie) groups=20(dialout),24(cdrom),25(floppy),29(audio),44(video),46(plugdev),106(netdev),109(powerdev),1000(dhanie)
```

- **tty**

Jika untuk menampilkan informasi tentang user yang sedang aktif anda menggunakan `whoami`, untuk menampilkan nama terminal dimana saat ini anda berada gunakan perintah `tty` ini. Di Linux anda bisa membuka beberapa terminal, ini disebut virtual terminal.

Contoh :

```
dhanie@localhost:~$ tty
/dev/pts/2
```

- **su** atau **su -**

Anda dapat berpindah dari user yang sedang aktif menjadi user lain tanpa harus melakukan logout. Gunakan perintah `su`

Format : `su option nama_user_pengganti`

Contoh :

```
[arie@localhost Desktop]$ su klas
```

Password:

Isikan password dari user klas, kemudian tekan Enter. Prompt berubah menjadi seperti dibawah ini.

```
[klas@localhost Desktop]$
```

User pengganti haruslah user yang sudah ada dalam sistem. Jika nama user pengganti tidak diketik, maka dianggap berpindah ke user root. Jadi perintah ini juga bisa anda gunakan saat anda ingin menjadi root. Gunakan password root yang anda setting pada saat menginstall Linux.

Contoh :

```
dhanie@localhost:/$ su
Password:
localhost:/# █
```

- **adduser** atau **useradd**

Untuk membuat user baru digunakan perintah **adduser** atau **useradd**. Perintah ini harus dijalankan melalui user root. Login atau bergantilah dari user biasa ke user root.

Format : `useradd option nama_user_baru`  
: `adduser option nama_user_baru`

Contoh :

```
localhost:~# adduser arie
Adding user `arie' ...
Adding new group `arie' (1002) ...
Adding new user `arie' (1002) with group `arie' ...
Creating home directory `/home/arie' ...
Copying files from `/etc/skel' ...
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
Changing the user information for arie
Enter the new value, or press ENTER for the default
  Full Name []: arie
  Room Number []:
  Work Phone []:
  Home Phone []:
  Other []:
Is the information correct? [y/N] y
localhost:~# █
```

Saat anda menjalankan perintah **adduser**, selain harus mengisi password, anda akan diminta untuk mengisi beberapa data yang diperlukan. Seperti nama, nomer telephone dsb.

Bila anda menggunakan perintah **useradd**, anda tidak perlu mengisi data-data tersebut. Anda bisa mengisi data dan memberikan password kemudian.

```
localhost:~# useradd tux
```

- **visudo** dan **sudo**



User biasa dapat menggunakan hak root sehingga dapat menjalankan perintah-perintah milik root. Untuk menggunakan hak root, digunakan perintah **sudo**. Tetapi sebelum menggunakan perintah sudo, user tersebut sudah harus masuk dalam daftar pengguna sudo. Untuk memasukkan user kedalam daftar, gunakan perintah **visudo** (ingat untuk bisa melakukan perintah visudo anda harus memiliki hak akses root).

Format : `visudo option`

Contoh :

```
[root@localhost Desktop]# visudo
```

Carilah kalimat seperti dibawah ini :

```
# User privilege specification
root    ALL=(ALL) ALL
```

Lalu tambahkan dibawah kalimat `root ALL=(ALL) ALL` ,nama user yang akan menjalankan sudo, menjadi :

```
# User privilege specification
root    ALL=(ALL) ALL
arie    ALL=(ALL) ALL
```

Setelah itu user arie dapat menjalankan perintah root. Untuk mencobanya, loginlah sebagai user arie (dengan perintah **SU** yang sudah anda pelajari sebelumnya) kemudian jalankan perintah `sudo adduser`.

Contoh :

```
[arie@localhost Desktop]$ sudo adduser
```

Password:

Anda akan diminta mengisikan password. Password yang diminta adalah password user arie, bukan root.

🔄 Beberapa distro membolehkan user biasa untuk langsung menjalankan sudo.

- **passwd**

Setelah membuat user baru dengan perintah **useradd**, kita perlu membuatkan password. Gunakan perintah **passwd**. Perintah ini juga

digunakan jika Anda ingin mengganti password yang sudah ada. Untuk mengganti password user lain, gunakan user root. Jika user ingin mengganti password nya sendiri, tidak diperlukan user root.

Format : `passwd option nama_user`

Contoh :

```
[root@localhost Desktop]# passwd tux
```

```
Changing password for user tux.
```

```
New UNIX password:
```

Isikan password yang diinginkan. Beberapa distro menghendaki minimal 6 karakter. Tekan enter setelah mengisi. Bila ada pesan seperti dibawah ini:

```
BAD PASSWORD: it is too short
```

```
Retype new UNIX password:
```

Password yang diisi dianggap terlalu pendek. Anda diminta mengisi ulang. Anda dapat merubah atau tetap menggunakan password semula. Pada contoh ini password tidak dirubah. Tekan enter setelah mengisi.

```
passwd: all authentication tokens updated successfully.
```

Muncul pesan bahwa perubahan password berhasil dilakukan. Proses penggantian password dari root jauh lebih mudah dibanding penggantian password dari posisi user itu sendiri. Kita lihat contoh jika user tux yang akan menggantikan password bagi dirinya.

Contoh :

```
[tux@localhost Desktop]# passwd
```

```
Changing password for user tux.
```

```
Changing password for tux
```

```
(current) UNIX password:
```

Isikan password awal yang sudah dibuat. Lalu tekan enter

```
New UNIX password:
```

```
Isikan password baru. Lalu tekan enter
```

```
BAD PASSWORD: it does not contain enough DIFFERENT characters
```

```
New UNIX password:
```

Muncul pesan bahwa karakter yang berbeda pada password baru dianggap kurang. Isikan kembali password baru dengan kombinasi karakter yang berbeda.

```
BAD PASSWORD: it is based on a dictionary word
passwd: Authentication token manipulation error
```

Muncul kembali pesan bahwa password yang diisi berdasarkan kamus. Dalam hal ini maksudnya mudah ditebak atau terlalu umum. Jika sampai tiga kali Anda tidak membuat password yang sesuai dengan syarat sistem maka akan muncul pesan "Authentication token manipulation error" dan selanjutnya Anda harus mengulangi perintah passwd.

🔄 Pada posisi user biasa ataupun root, jika perintah passwd tidak disertakan nama user-nya, maka dianggap akan mengganti password bagi dirinya sendiri.

- **userdel**

User yang sudah dibuat juga dapat dihapus. Gunakan perintah **userdel** untuk menghapus user.

Format : `userdel option nama_user_yang_akan_dihapus`

Contoh :

```
[root@localhost Desktop]# userdel -r tux
```

Setiap kali user baru dibuat, secara default sistem akan membuatkan home direktori bagi user tersebut. Nama home direktori sama dengan nama user-nya. Jika Anda ingin menghapus user, maka disarankan home direktori juga ikut dihapus. Jika tidak dihapus, bisa menimbulkan kerancuan, "*home direktorinya ada, tapi kok user-nya tidak ada?*". Untuk menghapus home direktori secara otomatis, tambahkan option **-r** seperti pada contoh.

-

- **groupadd**

Perintah ini digunakan untuk membuat group. Group yang dimaksud disini adalah sekelompok user yang saling bergabung dan mempunyai ketentuan tersendiri di kelompoknya. Setiap kali user baru dibuat, secara default sistem akan membuatkan sebuah grup yang namanya sama dengan nama user tersebut. Selain grup yang dibuat secara default oleh sistem, kita juga bisa membuat grup baru. Perintah yang digunakan adalah `groupadd`.

Format : `groupadd option nama_grup_baru`

Contoh :

```
|localhost:~# groupadd linuxer
```

```
|localhost:~# groupadd hacker
```

- **groupdel**

Group yang ada juga dapat dihapus. Gunakan perintah `groupdel`

Format : `groupdel nama_group_yang_akan_dihapus`

Contoh :

```
|localhost:~# groupdel hacker
```

#### 4.6 Redirection

Anda masih ingat dengan perintah `cat > nama_file.txt` ? Dengan mempelajari bagian Redirection ini, anda akan memahami perintah tersebut. Dalam UNIX/Linux, terdapat istilah *standard input*, *standard output* dan *standard error*.

- *Standard input* adalah masukan atau input standar dari suatu perintah atau program.
- *Standar output* adalah keluaran atau output standar dari suatu perintah atau program.
- *Standar error* adalah keluaran atau output standar jika pada perintah atau program terjadi kesalahan.

Standar input, standar output dan standar error dapat dibelokkan menuju tujuan lain. Misalnya membelokkan standar output suatu program

ke file, atau membelokkan standar input suatu program dari suatu file.

Proses membelokkan ini disebut *redirection*.

Simbol yang digunakan untuk membelokkan ini adalah :

> : untuk output

< : untuk input

>> : penambahan output

Untuk penggunaannya, user tinggal menambahkan simbol yang diinginkan pada perintah yang diberikan.

Format : `perintah option redirection file_penampung`

Contoh output :

```
dhanie@localhost:~/dataku$ ls /etc > isietc
```

Perintah tersebut akan menampilkan isi direktori `/etc` tetapi hasilnya (output) akan dimasukkan kedalam sebuah file yang bernama `isietc`.

Contoh input :

```
dhanie@localhost:~/dataku$ grep file < tesfile.txt
Ini adalah isi file.
Filanya bernama tesfile.
```

Perintah tersebut menerima input dari file `tesfile.txt` kemudian mencari kata-kata "**file**" dengan perintah `grep`.

Contoh penambahan :

Isi `daftarnama.txt` akan ditambahkan ke dalam `tesfile.txt` dengan perintah :

```
dhanie@localhost:~/dataku$ cat daftarnama.txt >> tesfile.txt
```

Lihat kembali isi file `tesfile.txt`

```
dhanie@localhost:~/dataku$ cat tesfile.txt
Ini adalah isi file.
Filenya bernama tesfile.
File ini akan dipakai sebagai latihan.
Latihan perang? ya bukan lah...
Brad Pitt
Angelina Jolie
Drew Barrymore
Lindsay Lohan
Pierce Brosnan
Bryan Adams
Christina Aguelera
```

#### 4.7 Pipe ( | )

Pipe atau pipeline atau pipa dalam bahasa Indonesia digunakan untuk komunikasi antar proses (perintah). Dengan pipe Anda dapat menghubungkan sebuah perintah yang menghasilkan sebuah output dengan perintah lain yang akan memproses output tersebut.

Format : `perintah option pipa perintah option`

Contoh :

Kita akan melihat isi direktori `/sbin` dengan perintah `ls -l`, isi direktori tersebut sangat banyak dan hanya ditampilkan sebagian akhirnya saja pada layar. Kita akan menghubungkan perintah `ls -l` dengan perintah `less` agar dapat melihat bagian lainnya.

Contoh :

```
[arie@localhost Desktop]$ ls -l /sbin | less
```

Simbol pipe ( | ) dapat digunakan lebih dari sekali. Contoh :

```
[arie@localhost Desktop]$ ls -l /sbin | less | grep mk
```

#### 4.8 Membersihkan Layar

Jika tampilan dilayar cukup banyak dan bisa membuat bingung, Anda dapat membersihkannya dengan perintah `clear` atau dengan shortcut `ctrl` dan `l`.

Contoh :

```
[arie@localhost Desktop]$ clear
```

## **KUIS :**

1. Buatlah perintah Command Line Interface ( CLI ) pada Linux untuk beberapa kondisi berikut:

- a. Buatlah direktori dengan nama informatika1 !
- b. Salinlah file dengan nama latihan1.txt dari direktori informatika1 kedalam informatika2 !



## Izin Akses File

Setiap file Linux memiliki status izin akses (file permission). Maksudnya setiap file memiliki informasi untuk mengatur siapa saja yang berhak untuk membaca, menjalankan atau mengubah file tersebut. Tujuannya adalah untuk menjaga privasi file, keamanan serta integritas sistem agar tidak terganggu. Izin akses ini otomatis diberikan saat file atau direktori tersebut dibuat.

### 5.1 Melihat Izin Akses File

Untuk mengetahui izin akses suatu file dapat digunakan perintah

`ls` dengan option `-l`

Contoh :

```
[arie@localhost Desktop]$ ls -l
drwxr-xr-x 2 arie arie 1024 Jan 24 05:36 dataku/
-rw-r--r-- 1 arie arie 5580 Jan 2 20:36 device.desktop
-rw-r--r-- 1 arie arie 4498 Jan 2 20:36 Home.desktop
-rw-rw-r-- 1 arie arie 225 Jan 15 20:57 kchmviewer.desktop
-rw-r--r-- 1 arie arie 0 Jan 23 05:32 tesfile
```

Pada tampilan tersebut, terdapat 9 kolom.

**-rw-r--r--** : ini adalah izin akses file  
**2** : link file  
**arie** : pemilik file  
**arie** : nama group pemilik file  
**5580** : ukuran file  
**Jan** : nama bulan pembuatan/modifikasi file  
**24** : tanggal pembuatan/modifikasi file  
**20:36** : jam pembuatan/modifikasi file  
**device.desktop** : nama file



Izin akses file ada tiga, yaitu :

- r** : read (membaca)
- w** : write (menulis)
- x** : execute (menjalankan)

Untuk izin akses file terdapat 10 digit karakter, yang dibagi menjadi tiga kelompok. Yaitu :

**-rwxrwxrwx**

**1 Karakter pertama ( - ),** digunakan untuk menentukan tipe file.

Tipe yang ada :

- : file biasa
- d** : direktori
- l** : link
- c** : special file
- s** : socket
- p** : name piped
- b** : block device

**3 karakter kedua (rwx),** digunakan untuk izin akses file terhadap **user** pemilik file.

**3 karakter ketiga (rwx),** digunakan untuk izin akses file terhadap **group** pemilik file.

**3 karakter keempat (rwx),** digunakan untuk izin akses terhadap **other** (user dan group yang lain).

Dengan demikian, pada contoh file **device.desktop** ini bisa dibaca sbb.

```
-rw-r--r-- 1 arie arie 5580 Jan 2 20:36 device.desktop
```

merupakan file biasa, pemilik file mempunyai izin akses **read** dan **write**, groupnya mempunyai izin akses **read** dan other mempunyai izin akses **read**.

Sekarang coba anda lihat ijin akses anda untuk file yang sudah pernah anda buat di direktori `/dataku` dengan perintah `ls -l`. Sebelum mengetikkan perintah tersebut pastikan direktori aktif anda adalah

`/dataku`

```
dhanie@localhost:~/dataku$ ls -l
total 16
drwxr-xr-x 3 dhane dhane 4096 2008-05-12 09:27 lagu
drwxr-xr-x 5 dhane dhane 4096 2008-04-30 08:40 picture
-rw-r--r-- 1 dhane dhane 401 2008-05-12 09:18 puisigombal.txt
drwxr-xr-x 2 dhane dhane 4096 2008-04-30 08:29 video
```

File `puisigombal.txt` milik user `dhane`. Untuk ijin aksesnya `Dhane` bisa membaca (`r`) dan menulis (`w`) file tersebut. Sedangkan user yang lain hanya bisa membaca saja.

## 5.2 Mengubah Izin Akses File

Ijin akses sebuah file dapat dirubah sesuai dengan kebutuhan. Untuk mengubahnya digunakan perintah `chmod`. Ada 2 macam mode mengubah Izin Akses File, yaitu **Symbolic mode** dan **Octal mode**.

### 5.2.1 Symbolic Mode

Untuk mengubah ijin akses file dengan Symbolic Mode, anda harus memahami aturan-aturannya seperti dibawah ini.

User pemilik file disingkat **u**

Group pemilik file disingkat **g**

User dan group lain disingkat **o (other)**

Atau bisa juga digunakan singkatan **a** untuk mewakili **ugo**. Untuk menambahkan ijin akses digunakan tanda **+** (plus) Untuk mengurangi ijin akses digunakan tanda **-** (minus)

Format : `chmod option ugo ijin_akses nama_file`

Contoh :

Lihat lebih dulu ijin akses file sebelum diubah

```
[arie@localhost Desktop]$ ls -l tesfile
```

```
-rw-r--r-- 1 arie arie 0 Jan 23 05:32 tesfile
```

Disini anda bisa mempraktekkannya dengan file milik anda

```
dhanie@localhost:~/dataku$ ls -l puisigombal.txt  
-rw-r--r-- 1 dhania dhania 401 2008-05-12 09:18 puisigombal.txt
```

Lalu ubahlah dengan menambahkan ijin execute pada user.

```
[arie@localhost Desktop]$ chmod u+x tesfile
```

Setelah mendapat ijin akses execute, file secara default berubah menjadi warna hijau.

Perhatikan perubahan ijin aksesnya dari `-rw-r--r--` (sebelum diubah ijin aksesnya) menjadi `-rwxr--r--` setelah diubah ijin aksesnya.

```
dhanie@localhost:~/dataku$ chmod u+x puisigombal.txt  
dhanie@localhost:~/dataku$ ls -l puisigombal.txt  
-rwxr--r-- 1 dhania dhania 401 2008-05-12 09:18 puisigombal.txt
```

Contoh-contoh lain :

```
[arie@localhost Desktop]$ chmod ugo-r tesfile
```

```
[arie@localhost Desktop]$ chmod ugo+rw tesfile
```

🔄 Setiap kali Anda melakukan perubahan ijin akses, cek kembali file tersebut untuk memastikan.

### 5.2.2 Octal Mode

Mode ini menggunakan angka octal (0-7) untuk melambangkan hak akses. Digambarkan sebagai berikut.

	user			group			others		
	r	w	x	r	w	x	r	w	x
400	-----+								
200	-----+								
100	-----+								
40	-----+								
20	-----+								
10	-----+								
4	-----+								
2	-----+								
1	-----+								
___+									
777									

Format: `chmod option nilai_oktal nama_file`

Contoh :

```
dhanie@localhost:~/dataku$ chmod 777 puisigombal.txt
dhanie@localhost:~/dataku$ ls -l puisigombal.txt
-rwxrwxrwx 1 dhania dhania 401 2008-05-12 09:18 puisigombal.txt
```

Setelah diubah ijin aksesnya file `puisigombal.txt` sekarang memiliki hak akses `-rwxrwxrwx`. Sehingga user, group, others bisa membaca, menulis dan menjalankan file tersebut.

Cobalah dengan kombinasi nilai octal yang lain lalu amati hasilnya.

**dhanie@localhost:~/dataku\$ chmod 664 puisigombal.txt**

```
dhanie@localhost:~/dataku$ ls -l puisigombal.txt
-rw-rw-r-- 1 dhania dhania 401 2008-05-12 09:18 puisigombal.txt
```

### 5.3 Kepemilikan File dan Group

Untuk keamanan dan privasi, setiap file di Linux memiliki identitas kepemilikan (ownership). Dengan adanya identitas ini maka akan jelas siapa pemilik file tersebut, siapa yang berhak membaca, menulis atau menjalankannya.

#### 5.3.1 Melihat Pemilik File dan Group

Untuk melihat kepemilikan suatu file dan group, gunakan perintah **ls** dengan option **-l**

Contoh :

```
[arie@localhost Desktop]$ ls -l tesfile
-rw-rw-rw- 1 arie arie 0 Jan 23 05:32 tesfile
```

Terlihat nama arie yang pertama adalah pemilik file dan nama arie yang kedua adalah nama groupnya. Secara default nama group sama seperti nama pemilik file.

#### 5.3.2 Mengubah Kepemilikan File & Direktori

Pemilik sebuah file atau direktori dapat diganti menjadi milik user yang lain. Untuk mengganti digunakan perintah **chown**

Format : *chown option pemilik\_baru  
nama\_file/direktori*

Contoh :

```
[root@localhost Desktop]# chown klas tesfile
```

Lihat kembali perubahannya,

```
root@localhost Desktop]# ls -l tesfile
-rw-rw-rw- 1 klas arie 0 Jan 23 05:32 tesfile
```

🔄 Perintah **chown** harus dilakukan melalui **root**. User pengganti sudah harus ada dalam sistem.

### 5.3.3 Mengubah Kepemilikan Group

Untuk mengubah pemilik group digunakan perintah **chgrp**. Perintah ini harus dilakukan melalui root dan group pengganti sudah harus ada dalam sistem.

Format : `chgrp option group_pengganti nama_file/direktori`

Contoh :

```
[root@localhost Desktop]# chgrp tesgrup tesfile
```

Lihat kembali perubahannya,

```
[root@localhost Desktop]# ls -l tesfile
```

```
-rw-rw-rw- 1 klas tesgrup 0 Jan 23 05:32 tesfile
```

## KUIS :

**1. Jelaskan permission yang dimiliki oleh sebuah file yang bernilai `rw-rw-r-x` ?**