

# **LINUX FUNDAMENTAL**



## **MODUL PELATIHAN**

**Jakarta 2006**

---



**InfoSolusi**

**PT. Inforindo Intersolusi  
Gedung Kindo Lt. 2  
Jl. Duren Tiga Raya No. 101  
Jakarta 12760**

## **KATA PENGANTAR**

Puji syukur kami panjatkan ke hadirat Tuhan Yang Maha Kuasa, atas tersusunnya Modul Pelatihan Linux Fundamental ini.

Setelah mengikuti Modul Pelatihan Linux Fundamental ini, para peserta pelatihan diharapkan mampu memahami dasar-dasar pengoperasian sistem operasi Linux dan dapat menggunakannya dalam lingkungan pekerjaan di bidang teknologi dan sistem informasi.

Modul ini masih jauh dari sempurna, maka saran dan kritik para pembaca sangat kami harapkan dan akan kami jadikan acuan untuk penyusunan edisi-edisi berikutnya.

Akhir kata, kami mengucapkan terima kasih kepada semua pihak yang telah membantu secara langsung maupun tidak langsung dalam penyusunan modul ini. Mudah-mudahan jerih payah kita menjadi amal yang baik, dan ilmu yang ada dalam modul ini bermanfaat bagi peserta serta kesejahteraan manusia pada umumnya.

Jakarta, November 2006

Penyusun

## Aturan Penulisan Modul

Aturan penulisan yang digunakan dalam modul ini adalah sebagai berikut :

1. Istilah bahasa Inggris dalam bidang komputer akan ditulis dalam *huruf miring*.
2. Contoh penggunaan dalam sistem Linux akan ditulis dengan huruf `Courier New` yang dibedakan menjadi dua, yaitu :
  - `Courier New Normal` menunjukkan apa yang ditampilkan oleh komputer
  - **`Courier New Bold`** menunjukkan apa yang harus diketikkan oleh *user*

Contoh :

```
$ cat latih  
saya sedang belajar
```

3. Tombol keyboard akan ditulis dalam huruf **`Courier New Bold`** dalam kurung siku.  
Contoh : **`[Enter]`**
4. Kombinasi tombol keyboard akan ditulis dalam huruf **`Courier New Bold`** yang dipisahkan oleh tanda **`+`**.  
Contoh : **`[Ctrl]+[D]`**
5. Prompt shell akan ditunjukkan dalam bentuk berikut :
  - `$` untuk user biasa
  - `#` untuk superuser

# DAFTAR ISI

|  |     |
|--|-----|
| Kata Pengantar.....                      | i   |
| Aturan Penulisan Modul.....              | ii  |
| Daftar Isi.....                          | iii |
| <br>                                     |     |
| Bab 1.Instalasi Linux Red Hat 9.....     | 1   |
| 1.1.Persiapan Sebelum Instalasi.....     | 1   |
| 1.2.Tahapan Instalasi.....               | 1   |
| Bab 2.Mengenal Linux.....                | 16  |
| 2.1.Apakah Linux itu?.....               | 16  |
| 2.2.Sejarah Linux.....                   | 16  |
| 2.3.Distribusi Linux.....                | 16  |
| 2.4.Penggunaan Linux.....                | 16  |
| 2.5.Informasi tentang Linux.....         | 16  |
| 2.6.Memulai Linux .....                  | 17  |
| 2.6.1.Login pada Modus Teks .....        | 17  |
| 2.6.2.Login pada Modus Grafik.....       | 17  |
| 2.7.Keluar dari Sistem Linux.....        | 17  |
| 2.7.1.Logout pada Modus Teks.....        | 17  |
| 2.7.2.Logout pada Modus Grafik.....      | 17  |
| Bab 3. Menggunakan Desktop.....          | 18  |
| 3.1.Desktop.....                         | 18  |
| 3.1.1.Panel.....                         | 18  |
| 3.1.2.Desktop Icon.....                  | 19  |
| 3.1.3.Sistem Menu.....                   | 20  |
| 3.2.Background Desktop.....              | 20  |
| 3.3.Lock Screen.....                     | 21  |
| 3.4.Screensaver.....                     | 21  |
| Bab 4. Shell Prompt.....                 | 23  |
| 4.1.Mengakses Shell Prompt.....          | 23  |
| 4.2.Shell bash.....                      | 23  |
| 4.3.File Startup.....                    | 23  |
| 4.4.Perintah Shell.....                  | 23  |
| 4.4.1.ls - list.....                     | 24  |
| 4.4.2.passwd.....                        | 24  |
| 4.4.3.more.....                          | 24  |
| 4.4.4.less.....                          | 24  |
| 4.4.5.man - manual page.....             | 25  |
| 4.4.6.w dan who.....                     | 25  |
| 4.4.7.cat - concatenate.....             | 25  |
| 4.4.8.pwd - print working directory..... | 25  |
| 4.4.9.cd - change directory.....         | 25  |
| 4.4.10.mkdir - make directory.....       | 26  |
| 4.4.11.cp - copy.....                    | 26  |
| 4.4.12.mv - move.....                    | 26  |
| 4.4.13.rm - remove.....                  | 26  |
| Bab 5. File dan Direktori.....           | 27  |
| 5.1.Direktori Utama Sistem Linux.....    | 27  |
| 5.2.Home Directory.....                  | 28  |

|   |    |
|---|----|
| 5.3.Current & Parent Directory.....                             | 28 |
| 5.4.Absolute Path dan Relative Path .....                       | 28 |
| 5.5.Penamaan Suatu File/Direktori.....                          | 28 |
| 5.6.Atribut File/Direktori.....                                 | 28 |
| 5.6.1.Tipe File.....  | 28 |
| 5.6.2.Permission File / Ijin akses.....                         | 29 |
| 5.6.3.Hard Link.....  | 30 |
| 5.6.4.Owner.....  | 31 |
| 5.6.5.Group.....  | 31 |
| 5.6.6.Ukuran.....   | 31 |
| 5.6.7.Tanggal.....  | 31 |
| 5.6.8.Jam.....  | 31 |
| 5.6.9>Nama.....   | 31 |
| 5.7.Nautilus File Manager.....                                  | 31 |
| 5.7.1.Menjalankan Nautilus.....                                 | 31 |
| 5.7.2.Mengubah Tampilan Nautilus.....                           | 32 |
| 5.7.3.Membuat Direktori.....                                    | 32 |
| 5.7.4.Mengubah Nama File atau Direktori.....                    | 32 |
| 5.7.5.Memindahkan File atau Direktori.....                      | 32 |
| 5.7.6.Menyalin File atau Direktori.....                         | 32 |
| 5.7.7.Menghapus File atau Direktori.....                        | 32 |
| 5.7.8.Memindahkan File dan Direktori ke Direktori Trash.....    | 33 |
| Bab 6. Editor Teks vi.....                                      | 34 |
| 6.1.Memulai vi.....   | 34 |
| 6.2.Modus Operasi.....  | 34 |
| 6.3.Pergerakan Kursor.....                                      | 34 |
| 6.4.Menyisipkan Teks.....                                       | 34 |
| 6.5.Menghapus Teks.....   | 34 |
| 6.6.Menyimpan File dan Keluar dari vi.....                      | 34 |
| 6.7.Copy, Cut, Paste.....                                       | 35 |
| 6.8.Penggantian Teks.....                                       | 35 |
| 6.9.Membatalkan perintah sebelumnya - undo.....                 | 35 |
| 6.10.Membuka dokumen lain sewaktu mengedit file.....            | 35 |
| 6.11.Keluar sementara dari vi dan mengeksekusi perintah.....    | 35 |
| 6.12.Pencarian String dalam vi.....                             | 35 |
| 6.12.1.Pencarian String.....                                    | 35 |
| 6.12.2.Pattern Matching dalam Pencarian String.....             | 36 |
| 6.13.Mengedit Teks dengan Menggunakan Perintah Baris Akhir..... | 36 |
| Bab 7. Pemrosesan String.....                                   | 37 |
| 7.1.wc - word count.....  | 37 |
| 7.2.cut.....  | 37 |
| 7.3.head.....   | 37 |
| 7.4.tail.....   | 38 |
| 7.5.sort .....  | 38 |
| 7.6.uniq.....   | 38 |
| 7.7.tr .....  | 39 |
| 7.8.grep.....   | 39 |
| 7.9.find.....   | 39 |
| Bab 8.Shell Script I.....                                       | 41 |
| 8.1.Pengantar Shell Script.....                                 | 41 |
| 8.1.1.Membuat Shell Script.....                                 | 41 |

|  |    |
|--|----|
| 8.1.2.Menjalankan Shell Script.....                | 41 |
| 8.2.Variabel .....                                 | 42 |
| 8.2.1.Menyimpan Harga Variabel.....                | 42 |
| 8.2.2.Menampilkan Harga Variabel.....              | 42 |
| 8.2.3.Menghapus Variabel .....                     | 42 |
| 8.3.Parameter .....                                | 42 |
| 8.3.1.Parameter Posisi (Positional Parameter)..... | 42 |
| 8.3.2.Parameter Khusus (Special Parameter).....    | 43 |
| 8.4.Pengutipan.....                                | 44 |
| 8.5.Kutip Balik.....                               | 45 |
| 8.6.Substitusi Nama File.....                      | 45 |
| 8.7.Input dan Output .....                         | 46 |
| 8.7.1.read.....                                    | 46 |
| 8.7.2.echo.....                                    | 46 |
| 8.8.Pencabangan dengan If.....                     | 47 |
| 8.8.1.if 47  |    |
| 8.8.2.Exit Status.....                             | 47 |
| 8.8.3.Variabel \$?.....                            | 48 |
| 8.8.4.Bentuk else if.....                          | 48 |
| 8.9.Evaluasi Kondisi.....                          | 49 |
| 8.9.1.test.....                                    | 49 |
| 8.9.2.Operator String.....                         | 49 |
| 8.9.3.Operator Integer.....                        | 49 |
| 8.9.4.Operator File.....                           | 50 |
| 8.9.5.Operator Logika.....                         | 50 |
| 8.9.6.Operator or (-o).....                        | 51 |
| 8.10.Pengulangan dengan for.....                   | 51 |
| Bab 9.Shell Script II.....                         | 54 |
| 9.1.Pencabangan dengan case.....                   | 54 |
| 9.1.1.case.....                                    | 54 |
| 9.1.2.Operasi Logika or Antar Dua Pola.....        | 55 |
| 9.1.3.Bentuk && (and-if).....                      | 55 |
| 9.1.4.Bentuk    (or-if).....                       | 55 |
| 9.2.Ekspresi Aritmetika.....                       | 56 |
| 9.2.1.expr.....                                    | 56 |
| 9.2.2.let.....                                     | 57 |
| 9.2.3.(( )).....                                   | 57 |
| 9.3.Pengulangan dengan while dan until.....        | 57 |
| 9.3.1.while.....                                   | 57 |
| 9.3.2.true.....                                    | 58 |
| 9.3.3.until.....                                   | 58 |
| 9.3.4.false.....                                   | 58 |
| 9.3.5.break.....                                   | 59 |
| 9.3.6.continue.....                                | 59 |
| 9.3.1.I/O Redirection dalam Pengulangan.....       | 60 |
| 9.4.Fungsi.....                                    | 60 |
| 9.4.1.return.....                                  | 60 |
| Bab 10.Komunikasi dan Internet.....                | 62 |
| 10.1.Berkomunikasi dengan User lain.....           | 62 |
| 10.1.1.write.....                                  | 62 |
| 10.1.2.talk.....                                   | 62 |

|  |    |
|--|----|
| 10.1.3.mail.....                             | 62 |
| 10.1.4.Mail dengan Evolution.....            | 63 |
| 10.2.Menjelajah Internet dengan Mozilla..... | 64 |
| 10.2.1.Menjalankan Mozilla.....              | 64 |

## Instalasi Linux Red Hat 9

### 1.1. Persiapan Sebelum Instalasi

Linux Red Hat 9 (RH 9) dapat diinstal dengan beberapa cara, dua diantaranya yaitu boot langsung dari CD RH 9 atau boot dari disket boot.

Jika akan menginstal melalui boot langsung dari CD RH 9, langkah pertama yang harus dilakukan adalah mengubah pilihan Boot Priority atau Boot Sequence pada BIOS setup agar boot pertama dilakukan lewat CD-ROM.

Jika akan menginstal RH 9 melalui boot dari disket boot, langkah pertama yang harus dilakukan adalah membuat disket boot dengan cara sebagai berikut :

1. Masukkan CD RH 9 ke CD-ROM drive.
2. Pindahkan drive aktif ke CD-ROM drive. Asumsi CD-ROM drive adalah drive D.
3. Masukkan sebuah disket kosong 1.44 MB ke drive A.
4. Ketikkan :  
`D:\>\dosutils\rawrite -f \images\bootdisk.img -d A -n [Enter]`
5. Setelah proses pembuatan disket boot selesai, disket siap digunakan.

### 1.2. Tahapan Instalasi

#### Langkah 1 : Persiapan Media Boot

Masukkan CD RH 9 ke CD-ROM drive. Jika akan melakukan instalasi dengan boot dari CD RH 9, boot komputer dan biarkan komputer boot dari CD RH 9. Jika akan melakukan instalasi dengan disket boot, masukkan disket boot ke drive A, dan biarkan komputer boot dari drive A. Cara manapun yang dipilih, proses selanjutnya adalah sama.

#### Langkah 2 : Pemilihan Bahasa untuk Proses Instalasi



**Gambar 2-1. Pemilihan Bahasa untuk Proses Instalasi**

Setelah proses boot, baik boot dari CD maupun disket boot, akan muncul tampilan seperti Gambar 2-1 untuk melakukan pemilihan bahasa yang akan digunakan dalam proses instalasi.



Sayangnya untuk distribusi Linux Red Hat belum tersedia pilihan bahasa Indonesia untuk instalasi. Pada daftar bahasa yang tersedia, pilihlah English (English), lalu klik tombol **Next**.

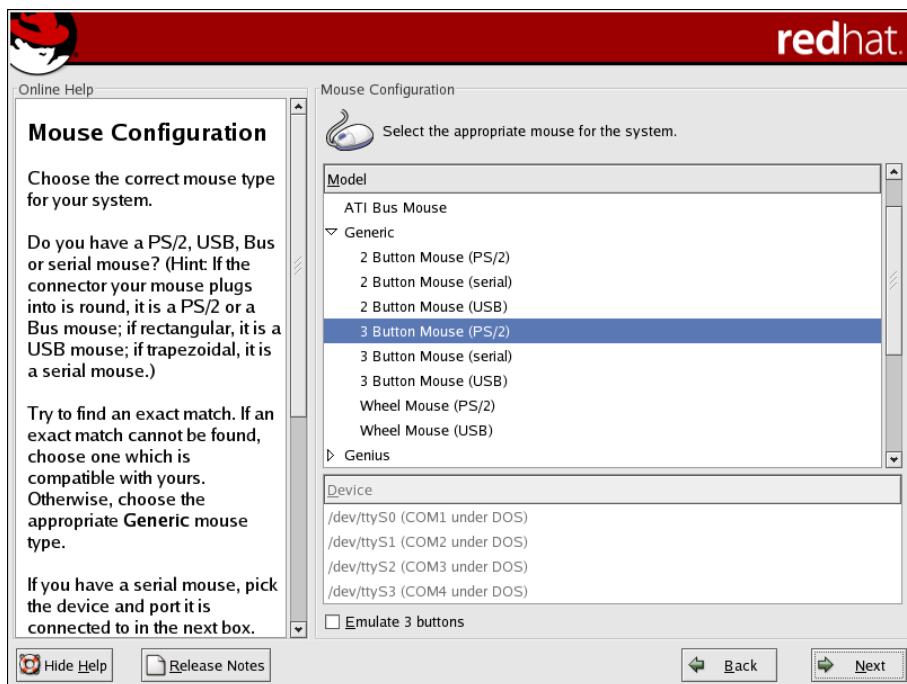
### Langkah 3 : Pemilihan Keyboard



Gambar 2-2. Pemilihan Keyboard

Pada pilihan jenis keyboard yang tersedia, pilihlah jenis U.S. English. Klik tombol **Next** untuk melanjutkan.

### Langkah 4 : Pemilihan Mouse

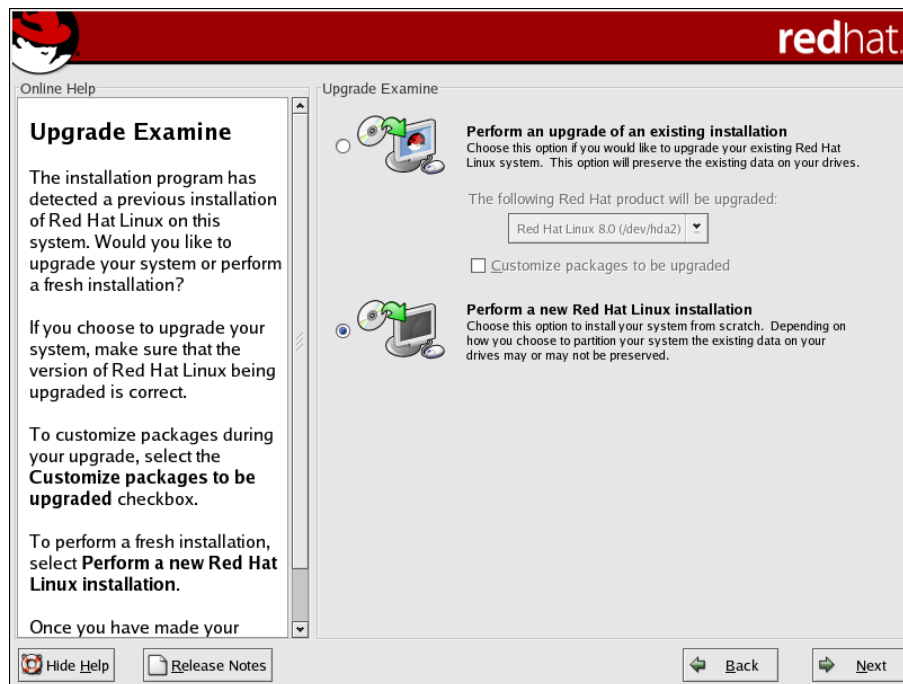


Gambar 2-3. Pemilihan Mouse

Pilihlah jenis mouse yang sesuai. Perhatikan jenis interface yang digunakan apakah serial, PS/2 atau USB. Sebaiknya pilih yang berjenis Generic agar kompatibel dengan semua mouse yang ada di pasaran.

Klik tombol **Next** untuk melanjutkan.

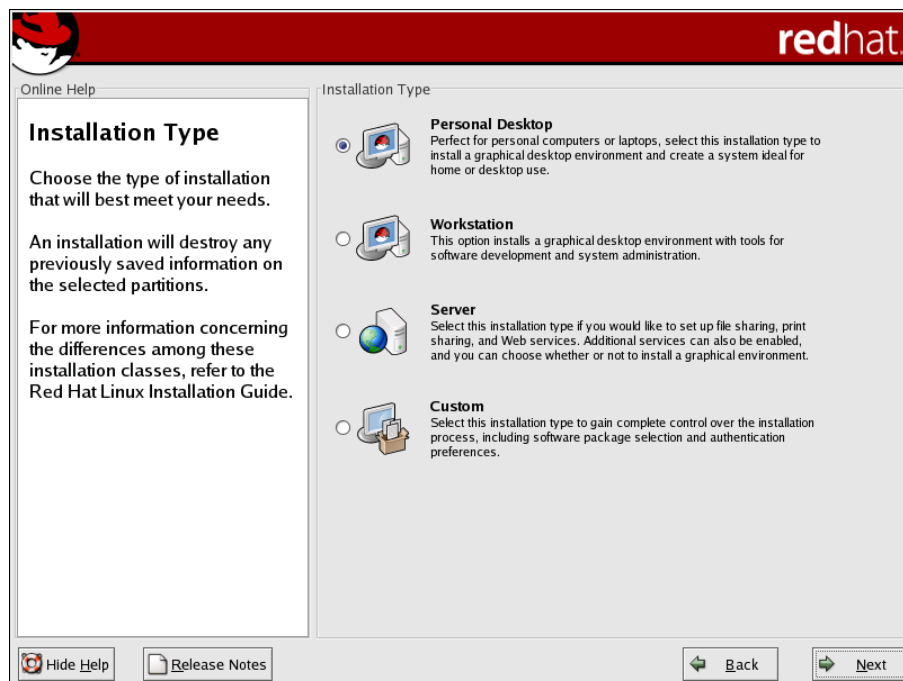
### Langkah 5 : Memilih Upgrade atau Install



Gambar 2-4. Memilih Upgrade atau Install

Program instalasi secara otomatis akan mendeteksi apabila sebelumnya sudah terinstal Linux Red Hat. Pada contoh kali ini pilih **Perform a new Red Hat Linux installation**. Klik tombol **Next** untuk melanjutkan.

### Langkah 6 : Pilih Jenis Instalasi



Gambar 2-5. Pilihan Jenis Instalasi

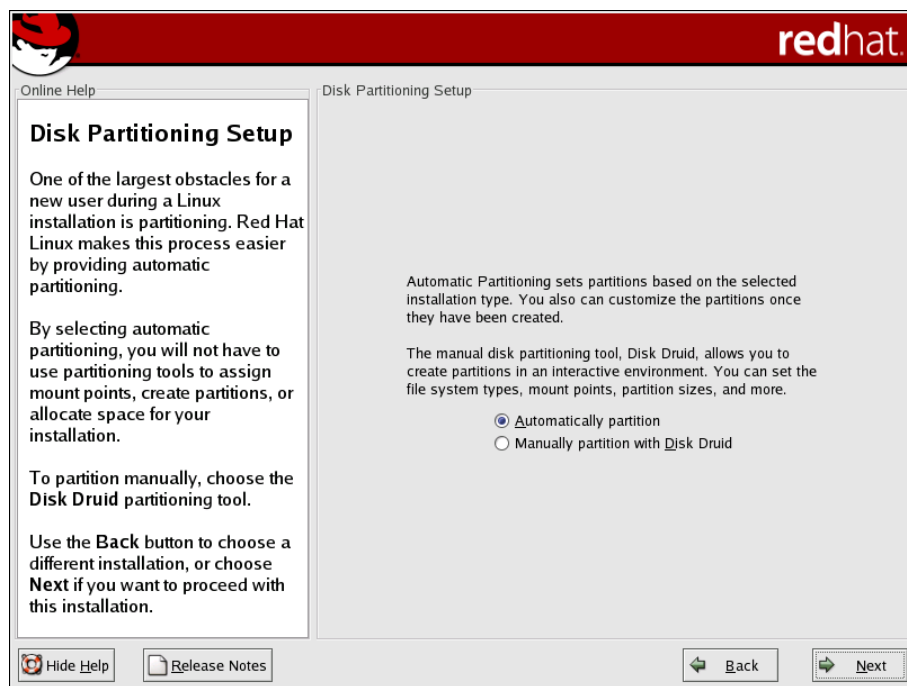
Beberapa jenis instalasi yang bisa dipilih adalah :

- **Personal Desktop** : Hanya akan menginstal paket-paket untuk desktop saja seperti KDE atau GNOME. Membutuhkan kapasitas hard disk sekitar 1,7 GB untuk KDE atau GNOME, atau sekitar 1,8 GB untuk menginstal keduanya.
- **Workstation** : Hanya akan menginstal paket-paket untuk desktop saja seperti KDE atau GNOME serta software development tools. Membutuhkan kapasitas hard disk sekitar 2,1 GB untuk KDE atau GNOME, atau sekitar 2,2 GB untuk menginstal keduanya.
- **Server** : Khusus untuk menginstal paket-paket server seperti email server, web server, samba server, squid, dan lain-lain. Membutuhkan kapasitas hard disk sekitar 850 MB sampai dengan 5 GB.
- **Custom** : Pilih jenis instalasi ini untuk memilih secara manual paket-paket yang akan diinstal. Paket minimal yang bisa diinstal akan menghabiskan kapasitas hard disk sekitar 475 MB, atau pilih semua paket yang akan menghabiskan kapasitas hard disk sekitar 5 GB.

Pada contoh instalasi kali ini, pilih **Custom** agar Anda tahu paket-paket apa saja yang terdapat pada RH 9.

Klik tombol **Next** untuk melanjutkan.

### Langkah 7 : Pilih Setup Partisi



**Gambar 2-6. Pilih Setup Partisi**

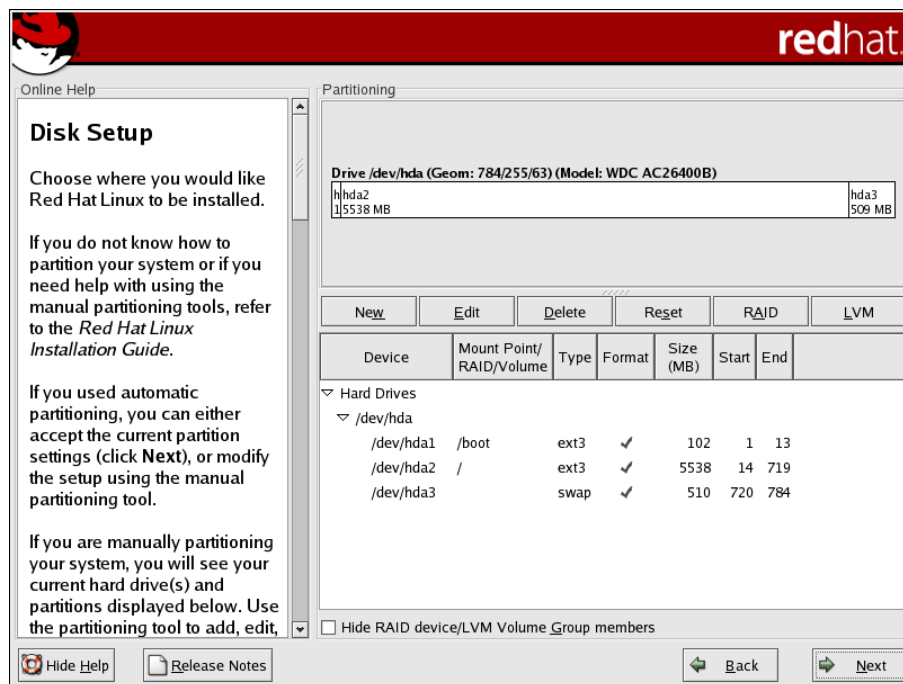
Ada dua jenis setup partisi yang bisa dipilih :

- **Automatically partition**: Pembuatan partisi akan dilakukan secara otomatis oleh program instalasi. Pilihan ini cocok untuk user awam. Jika sebelumnya sudah ada partisi Linux, seluruh data dalam partisi tersebut akan dihapus.
- **Manually partition with Disk Druid**: Pembuatan partisi akan dilakukan secara manual menggunakan Disk Druid yang mudah.

Pilih **Manually partition with Disk Druid** agar Anda tahu bagaimana caranya membuat partisi Linux secara manual.

Klik tombol **Next** untuk melanjutkan.

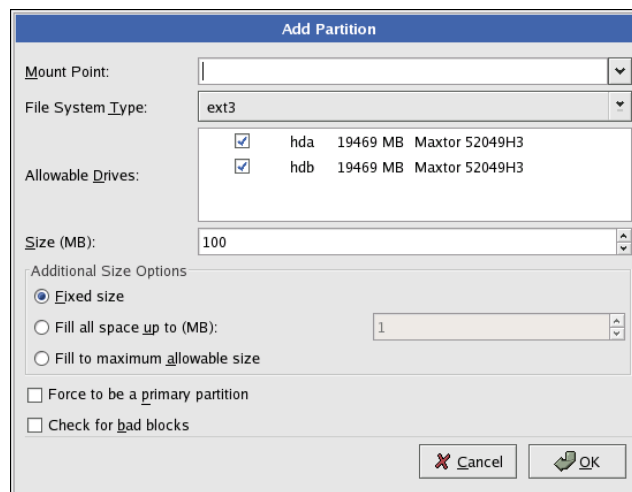
## Langkah 8 : Membuat Partisi



Gambar 2-7. Membuat Partisi

Pada Linux terdapat dua jenis partisi yaitu :

1. Partisi **Swap** yang digunakan untuk mendukung virtual memory. Besarnya minimal 32 MB. Besarnya partisi swap yang disarankan adalah dua kali jumlah memori yang terpasang pada komputer. Jika memori komputer Anda 1 GB atau kurang, buat partisi swap sebesar jumlah memori hingga dua kali jumlah memori terpasang. Jika memori komputer Anda lebih dari 1 GB, buat partisi swap sebesar 2 GB.
2. Partisi **Native**. Partisi native adalah partisi untuk sistem Linux itu sendiri. Jumlah partisi native bisa satu (hanya partisi root saja), atau bisa juga lebih dari satu. Partisi-partisi native yang disarankan untuk keperluan desktop adalah :
  - A. Partisi /boot. Buatlah partisi /boot sebesar 100 MB. Partisi ini digunakan untuk menyimpan berbagai file untuk keperluan boot Linux, termasuk didalamnya adalah kernel Linux itu sendiri.
  - B. Partisi root "/". Buatlah partisi root sesuai keperluan. Besar minimal yang disarankan adalah 1,7 GB untuk jenis instalasi Personal Desktop, atau 5 GB untuk instalasi seluruh paket.



Gambar 2-8. Menambah Partisi

Untuk membuat partisi sesuai dengan yang disarankan, ikuti langkah berikut :

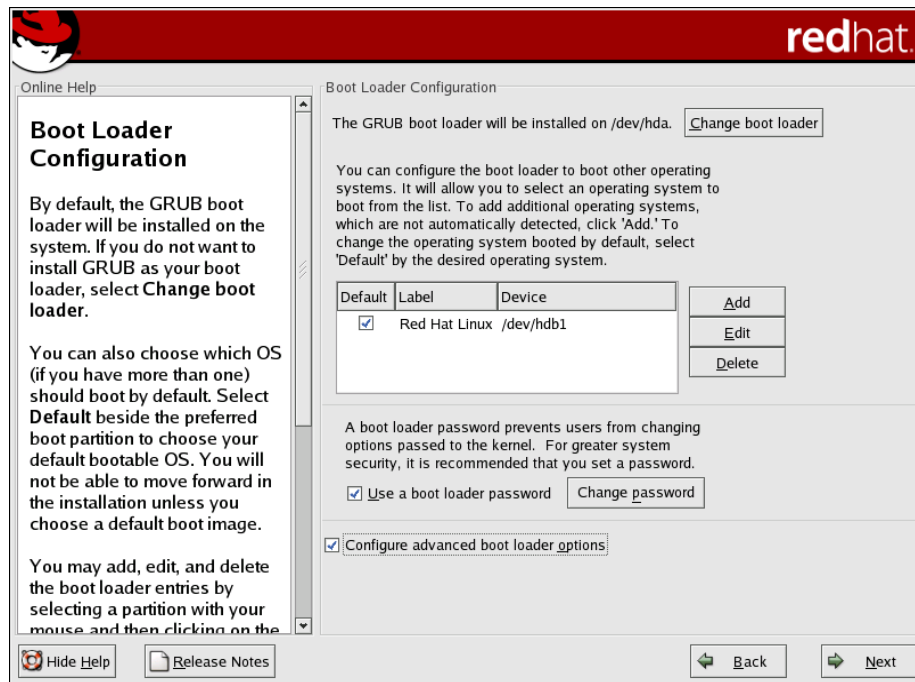
1. Buat partisi swap sebesar dua kali memori yang terpasang pada komputer, atau maksimal 2 GB, dengan cara sebagai berikut :
  - A. Klik tombol **New**. Kemudian akan muncul jendela untuk membuat partisi.
  - B. **Mount Point**, biarkan kosong.
  - C. **File System Type**, pilih Swap.
  - D. **Allowable Drives**, berisi daftar hard disk yang terinstal. Beri tanda cek pada harddisk yang mana partisi akan dibuat.
  - E. **Size (MB)**, masukkan besarnya partisi swap sesuai dengan yang telah dijelaskan sebelumnya.
  - F. **Additional Size Options**, pilih Fixed Size.
  - G. **Force to be a primary partition**, tidak perlu diberi tanda.
  - H. **Check for bad blocks**, beri tanda agar pada saat dilakukan format, block yang rusak akan ditandai sehingga dikemudian hari tidak digunakan.
  - I. Klik tombol **OK**.
2. Buat partisi /boot sebesar 100 MB dengan cara sebagai berikut :
  - A. Klik tombol **New**. Kemudian akan muncul jendela untuk membuat partisi.
  - B. **Mount Point**, isi dengan kata /boot.
  - C. **File System Type**, pilih ext3.
  - D. **Size (MB)**, isi dengan angka 100.
  - E. **Allowable Drives**, berisi daftar hard disk yang terinstal. Beri tanda cek pada harddisk yang mana partisi akan dibuat.
  - F. **Additional Size Options**, pilih Fixed Size.
  - G. **Force to be a primary partition**, beri tanda agar partisi yang dibuat disimpan di partisi primary.
  - H. **Check for bad blocks**, beri tanda agar pada saat dilakukan format, block yang rusak akan ditandai sehingga dikemudian hari tidak digunakan.
  - I. Klik tombol **OK**.
3. Buat partisi root "/" dengan cara sebagai berikut :
  - A. Klik tombol **New**. Akan muncul jendela untuk membuat partisi.
  - B. **Mount Point**, isi dengan tanda garis miring (/).
  - C. **File System Type**, pilih ext3.
  - D. **Size (MB)**. Masukkan besarnya partisi root minimal 1,7 GB (pada kotak ini masukkan nilai minimal 1700 karena satuannya adalah MB).
  - E. **Allowable Drives**, berisi daftar hard disk yang terinstal. Beri tanda cek pada harddisk yang mana partisi akan dibuat.
  - F. **Additional Size Options**, pilih **Fill to maximum allowable size** apabila ingin menggunakan semua kapasitas sisa hard disk yang ada untuk partisi root.
  - G. **Force to be a primary partition**, beri tanda agar partisi yang dibuat disimpan di partisi primary.
  - H. **Check for bad blocks**, beri tanda agar pada saat dilakukan format, block yang rusak akan ditandai sehingga dikemudian hari tidak digunakan.
  - I. Klik tombol **OK**.

Klik tombol **Next** jika pembuatan partisi sudah selesai.

## Langkah 9 : Konfigurasi Boot Loader

Boot loader adalah program yang pertama kali akan dijalankan ketika komputer dinyalakan. Program instalasi menyediakan dua program boot loader yaitu GRUB dan LILO.

GRUB (GRand Unified Bootloader), merupakan boot loader default sedangkan alternatifnya adalah LILO kependekan dari LInux LOader.



Gambar 2-9. Konfigurasi Boot Loader

GRUB atau LILO bisa dipasang pada :

- Master Boot Record (MBR). MBR adalah area khusus pada hard disk (letaknya di cylinder 0, head 0, sector 1) yang secara otomatis akan di-load oleh BIOS pada saat proses boot. Sangat disarankan untuk menempatkan GRUB atau LILO pada daerah ini, kecuali pada komputer yang telah terpasang boot loader lain seperti System Commander atau OS/2 Boot Loader atau NT Loader.
- First Sector of your root partition. Pilih area ini jika pada komputer telah terdapat boot loader lain yang tidak ingin diubah. Dengan memilih ini, maka boot loader tersebut akan di-load oleh BIOS, dan pertama kali muncul saat proses boot. Boot loader tersebut harus di-setting agar dapat me-load GRUB atau LILO yang diletakkan pada sektor pertama partisi root.

Klik tombol **Next** untuk melanjutkan.

## Langkah 10. Konfigurasi Network

Tampilan Gambar 2-10 hanya akan tampil jika pada komputer terdapat kartu jaringan (Network Interface Card) yang sudah dikenal dengan baik oleh RH 9.

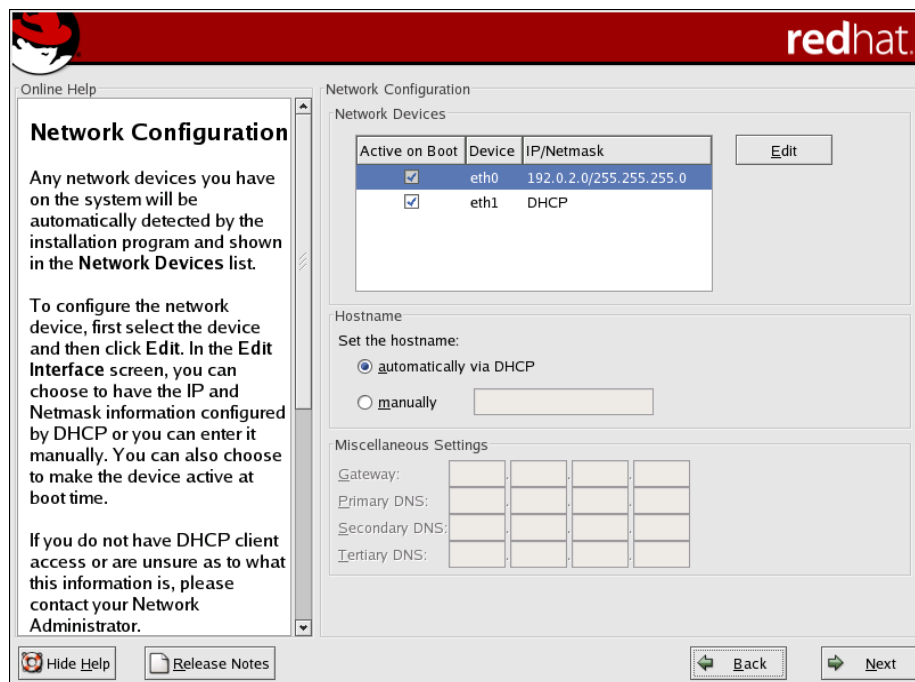
Secara otomatis alamat IP akan di set ke DHCP. Apabila pada jaringan tidak terdapat DHCP server dan anda ingin menentukan alamat IP secara manual klik tombol **Edit**, akan muncul jendela **Edit Interface**. Kosongkan kotak **Configure using DHCP** kemudian masukkan informasi alamat IP dan netmask pada kotak isian yang sesuai.

Beri tanda pada kotak cek **Activate on boot**.

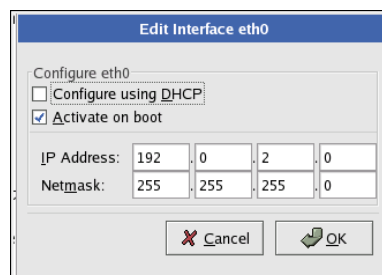
Klik tombol **OK**, untuk kembali ke jendela Konfigurasi Network.

Masukkan nama host, lengkap dengan nama domain-nya pada kotak isian **Hostname**. Hostname adalah nama komputer yang akan dipakai sebagai identitas. Jika kotak isian ini tidak diisi, secara otomatis nama komputer akan menjadi localhost.

Masukkan alamat Gateway dan Primary DNS pada kotak isian **Miscellaneous Settings**. Klik tombol **Next** untuk melanjutkan.



**Gambar 2-10. Konfigurasi Network**



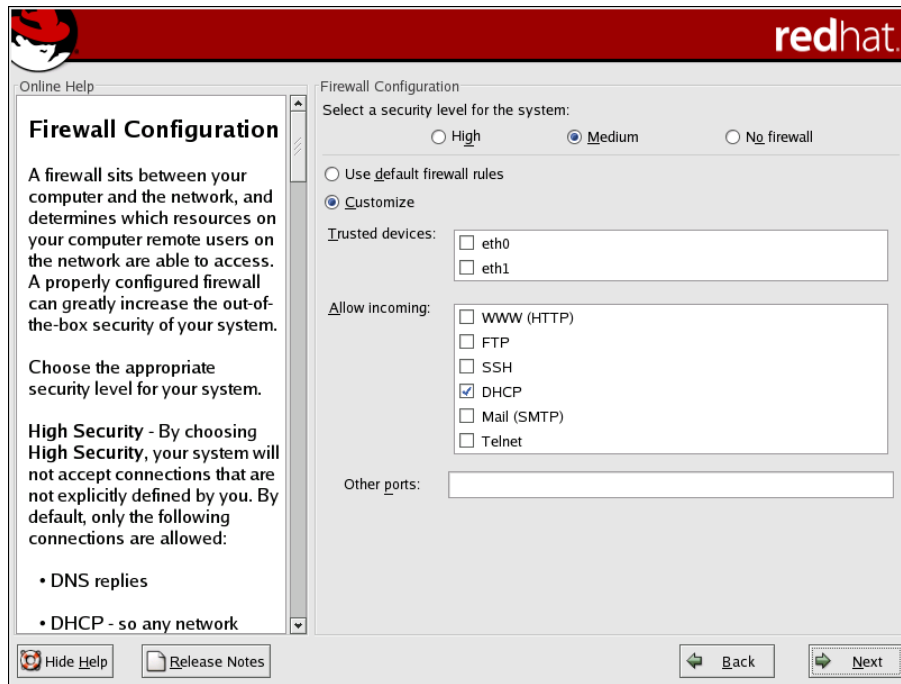
**Gambar 2-11. Jendela Edit Interface**

## Langkah 11. Konfigurasi Firewall

Lihat Gambar 2-12.

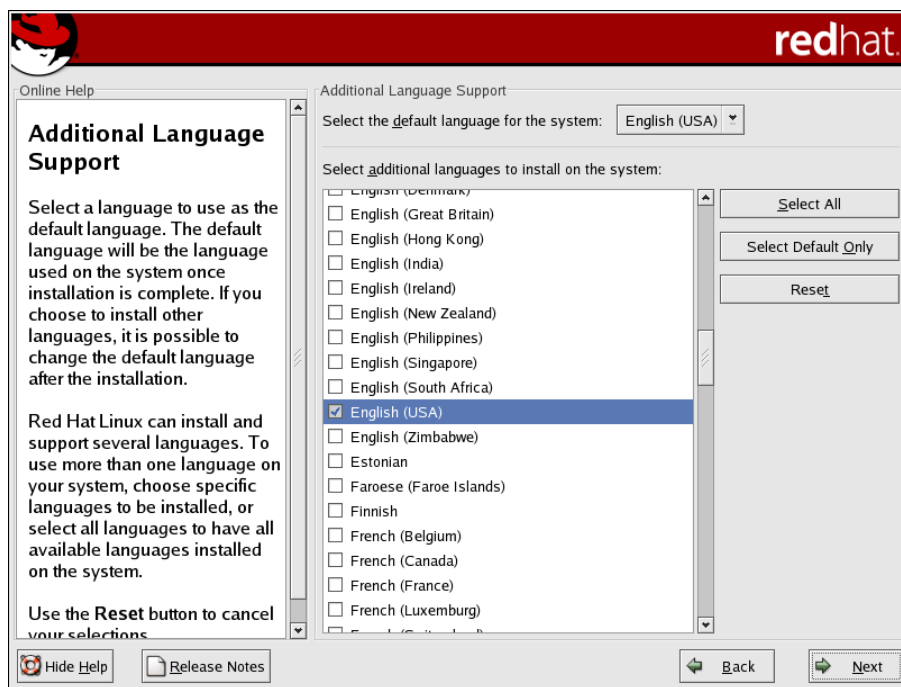
Firewall sangat diperlukan apabila komputer Anda terhubung ke jaringan luar. Firewall akan melindungi komputer Anda dari penyusupan oleh orang yang tidak berhak mempunyai akses.

Pilih level keamanan yang sesuai dengan kebutuhan. Pilih salah satu dari High, Medium, atau No Firewall. Jika bingung, pilih saja High, dan beri tanda pada kotak opsi Use default firewall rules. Klik tombol **Next** untuk melanjutkan.



Gambar 2-12. Konfigurasi Firewall

## Langkah 12. Pilih Bahasa yang Didukung



Gambar 2-13. Pilih Bahasa yang Didukung

Setidaknya harus dipilih satu bahasa. Secara default English (USA) sudah dipilih. Jika Anda memilih lebih dari satu bahasa, tentukan bahasa default yang akan digunakan pada kotak drop-down **Choose the default language**.

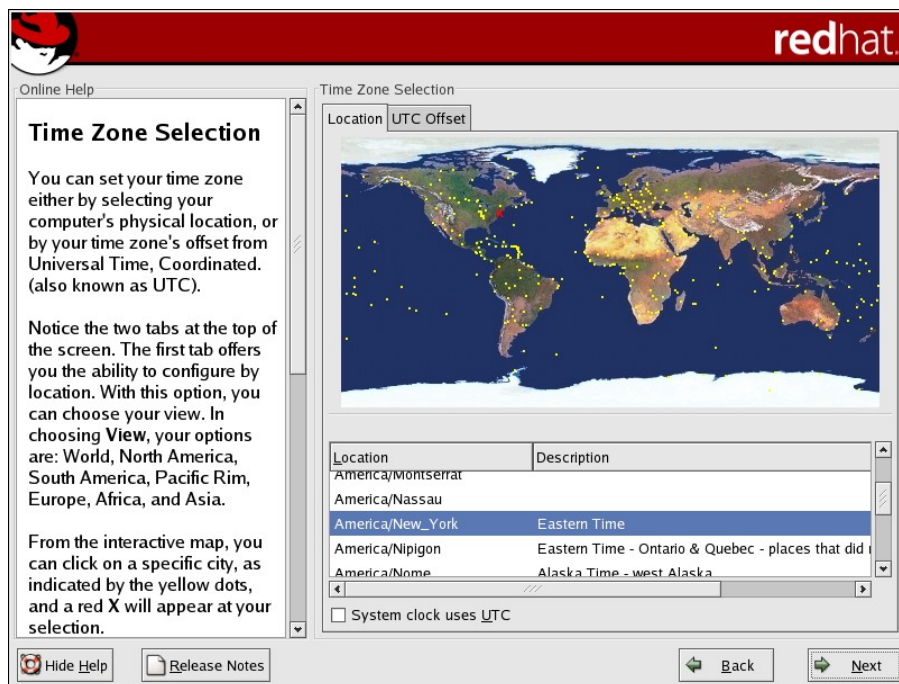
Klik tombol **Next** untuk melanjutkan.

## Langkah 13. Konfigurasi Time Zone

Lihat Gambar 2-14. Pilihlah time zone sesuai dengan lokasi komputer saat ini. Untuk user di Jakarta, pilih saja Asia/Jakarta.

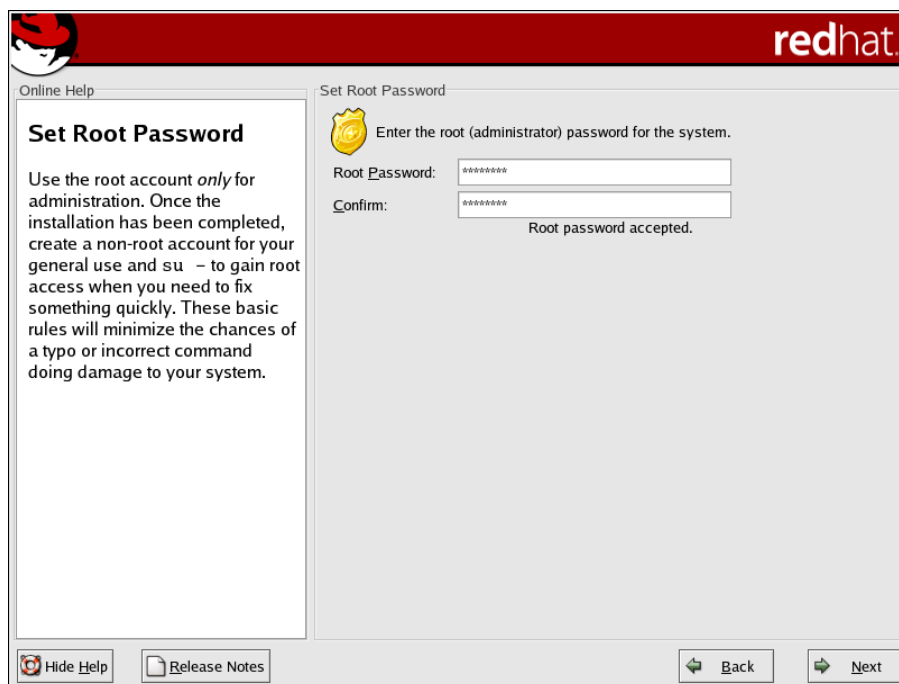
Klik tombol **Next** untuk melanjutkan.





Gambar 2-14. Konfigurasi Time Zone

#### Langkah 14. Isikan Password Root



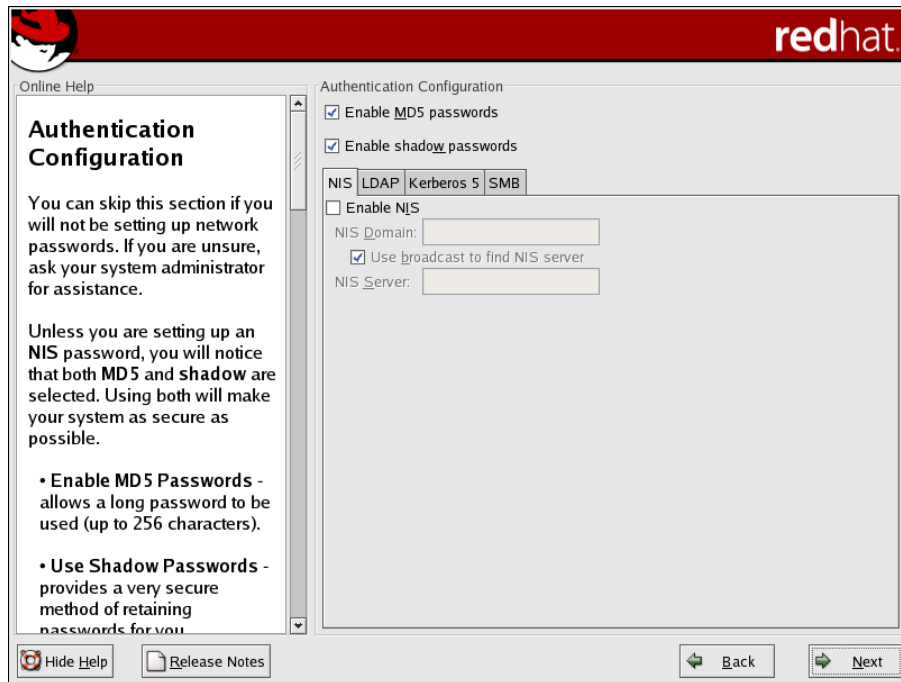
Gambar 2-15. Isikan Password Root

Pada kotak isian Root Password dan Confirm, masukkan password untuk user root. Jumlah karakter minimal untuk password user root adalah enam.

#### Langkah 15. Konfigurasi Authentication

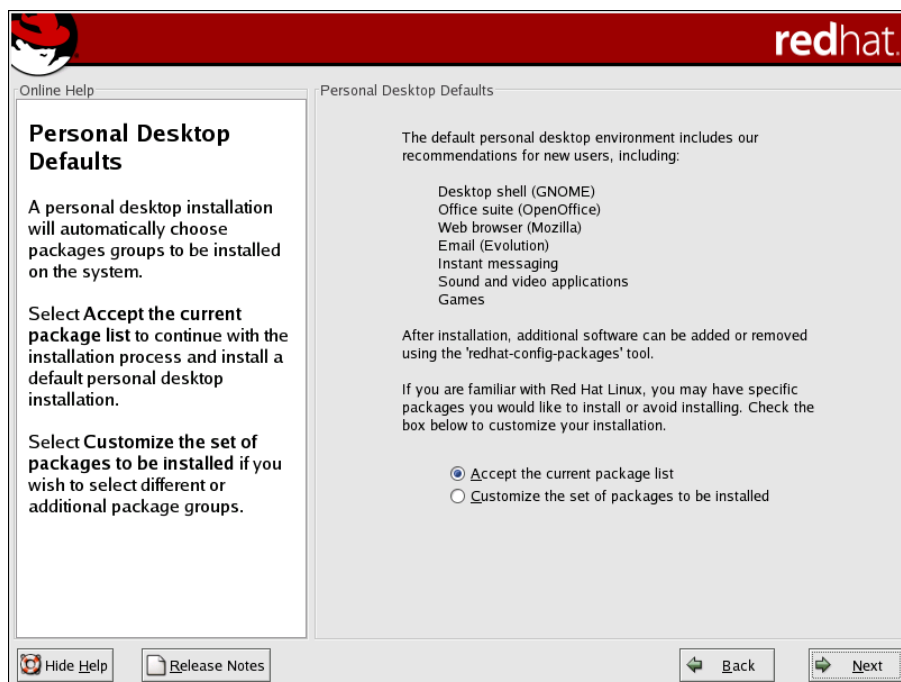
Lihat Gambar 2-16. Beri tanda pada kotak cek **Enable MD5 passwords** dan **Enable shadow passwords**.

Klik tombol **Next** untuk melanjutkan ke langkah selanjutnya.



Gambar 2-16. Konfigurasi Authentication

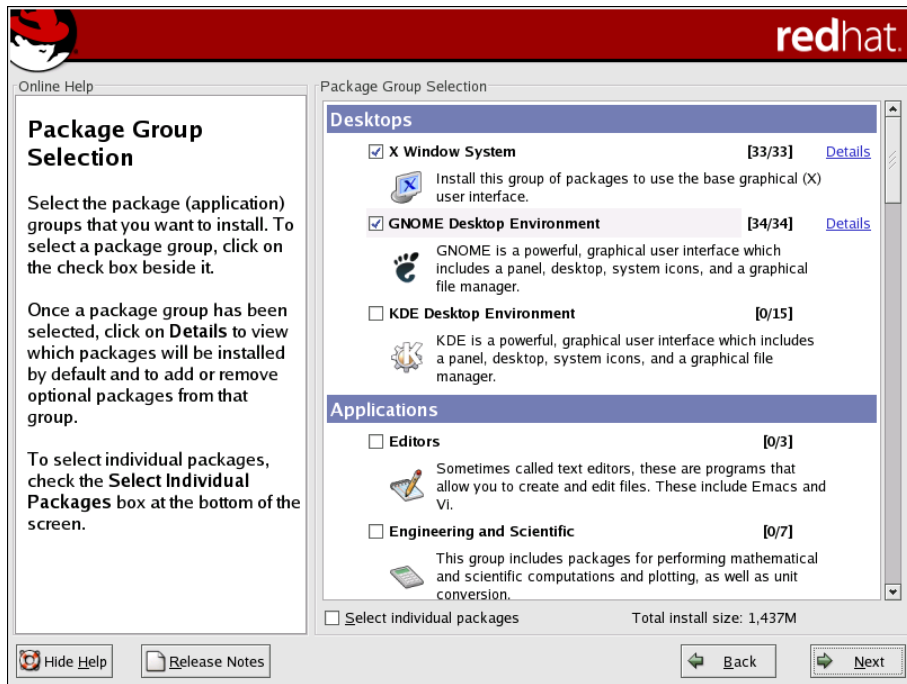
## Langkah 16. Pemilihan Grup Paket



Gambar 2-17. Paket Standar untuk Personal Desktop

Secara default, program instalasi sudah memilih beberapa paket standar yang harus diinstal kecuali pada instalasi jenis custom. Sebagai contoh apabila memilih instalasi jenis Personal Desktop, maka akan terlihat seperti Gambar 2-17.

Jika ingin memilih paket-paket diluar paket standar, pilih **Customize the set of packages to be installed**. Tampilannya akan muncul seperti pada Gambar 2-18.

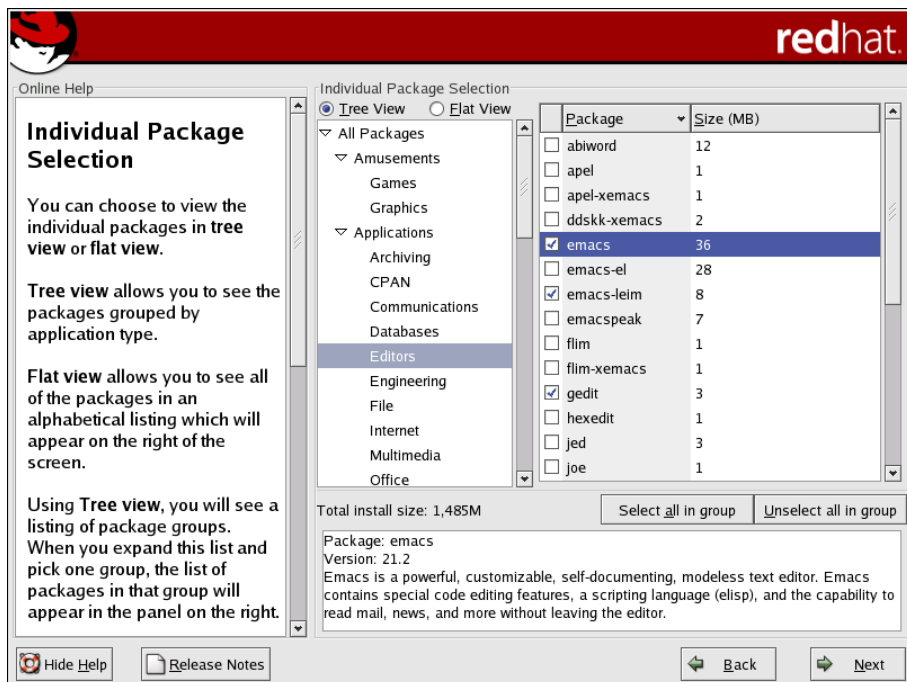


Gambar 2-18. Pemilihan Grup Paket

### Langkah 17. Pemilihan Paket Secara Individu

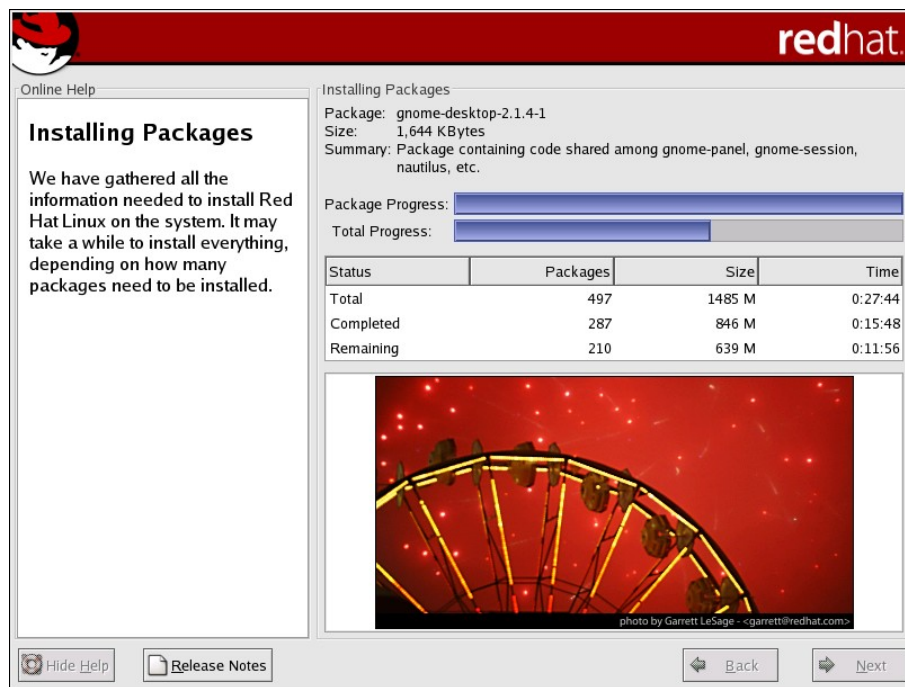
Tampilan pada Gambar 2-19 hanya akan muncul jika pada langkah sebelumnya memilih opsi **Select individual packages**.

Pilihlah paket-paket yang dikehendaki. Jika bingung, biarkan saja apa adanya, atau klik tombol **Back**, dan hilangkan tanda pada kotak cek **Select individual packages**, lalu klik tombol **Next** untuk melanjutkan ke langkah selanjutnya.



Gambar 2-19. Pemilihan Paket secara Individu

## Langkah 18. Instalasi Paket-paket

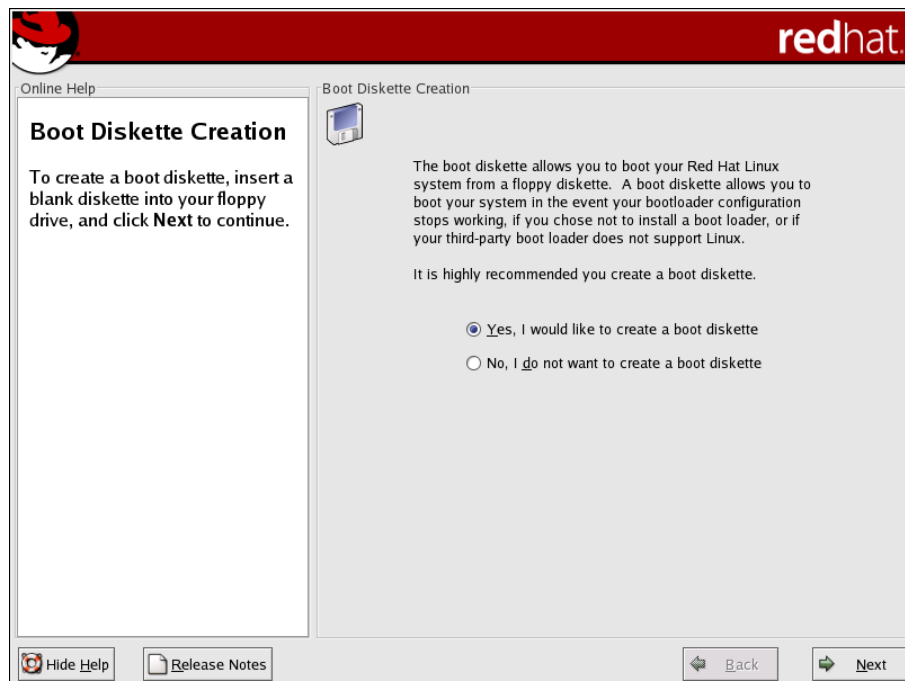


Gambar 2-20. Instalasi Paket

Tunggulah beberapa saat. Program instalasi akan melakukan persiapan proses instalasi dan melakukan instalasi paket-paket. Tidak ada yang harus dilakukan saat ini, kecuali menunggu program instalasi selesai melakukan instalasi paket.

CD instalasi RH 9 terdiri atas tiga CD, umumnya pada pertengahan proses instalasi, program instalasi akan meminta user untuk memasukkan CD 2 dan CD 3. Ikutilah petunjuk yang diberikan.

## Langkah 19. Membuat Disket Boot

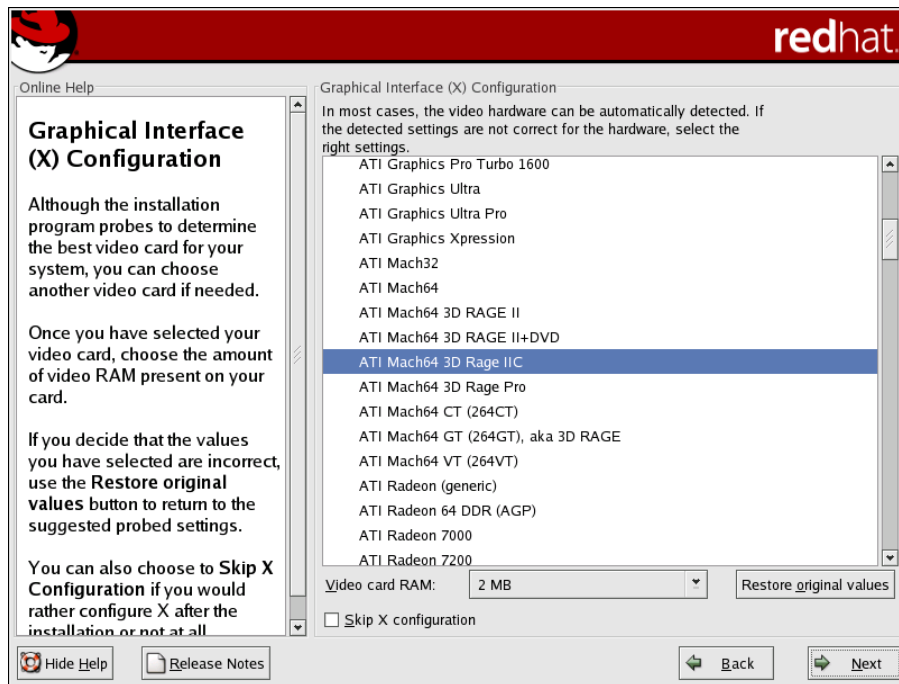


Gambar 2-21. Membuat Disket Boot

Disket boot biasanya digunakan pada saat *emergency*. misalnya ketika boot loader terhapus. Masukkan disket kosong ke drive A, lalu klik tombol **Next** untuk memulai pembuatan disket boot.

Jika tidak ingin membuat disket boot, pilih **No, I do not want to create a boot diskette** lalu klik tombol **Next**.

## Langkah 20. Konfigurasi Kartu Grafik



Gambar 2-22. Konfigurasi Kartu Grafik

Biasanya program instalasi sudah dapat mendeteksi jenis kartu grafik yang terpasang dan langsung memilih driver yang cocok. Jika tidak pilihlah jenis grafik yang sesuai dengan kartu grafik yang terpasang pada komputer.

Tekan tombol **Next** untuk melanjutkan ke langkah selanjutnya.

## Langkah 21. Pemilihan Monitor

Lihat Gambar 2-23. Biasanya program instalasi sudah dapat mendeteksi jenis monitor yang terpasang dan langsung memilih driver yang cocok.

Jika Anda tidak tahu, pilihlah jenis Generic yang non-interlace karena umumnya monitor yang beredar saat ini banyak yang non-interlace.

Klik tombol **Next** untuk melanjutkan.

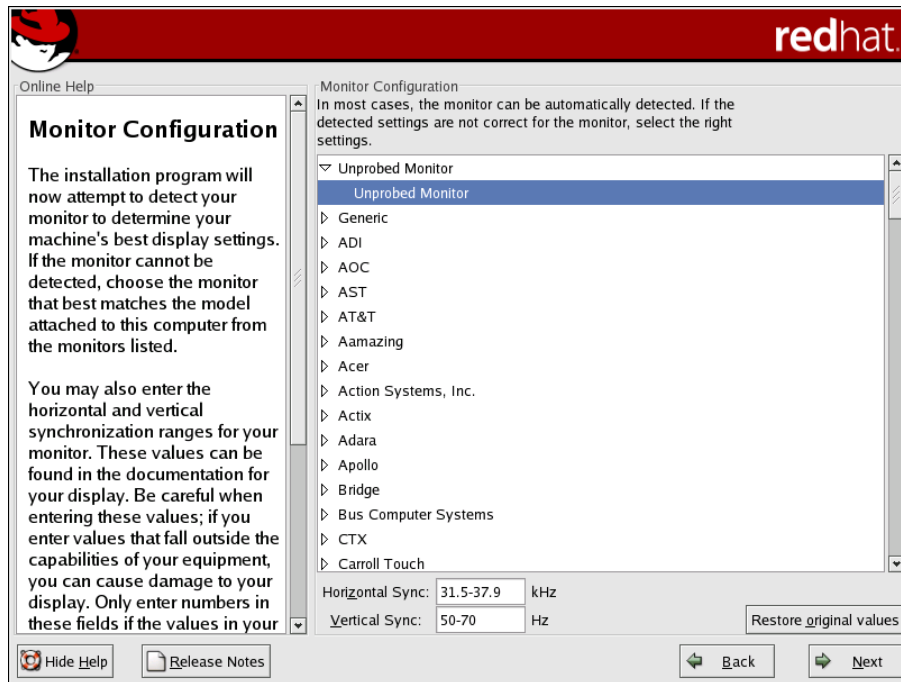
## Langkah 22. Pemilihan Resolusi dan Kedalaman Warna

Lihat Gambar 2-24. Pada kotak drop-down **Color Depth** tentukan kedalaman warna yang diinginkan. Untuk hasil tampilan yang bagus, pilih High Color 16 bit atau 24 bit.

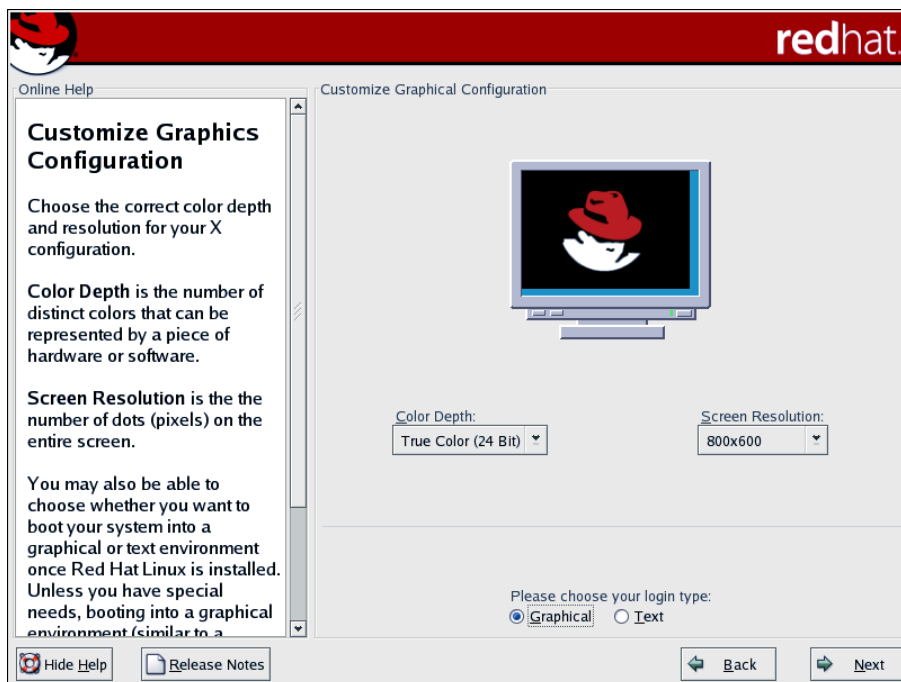
Pada kotak drop-down **Screen Resolution**. Pilih 1024 x 768 untuk hasil tampilan yang baik.

Pada bagian **Please choose your login type**, pilih jenis login yang diinginkan, apakah Text atau Graphical.

Klik tombol **Next** untuk melanjutkan ke langkah selanjutnya.



**Gambar 2-23. Pemilihan Monitor**



**Gambar 2-24. Pemilihan Resolusi dan Kedalaman Warna**

## Langkah 21. Instalasi Selesai

Instalasi telah selesai, keluarkan disket dan CD RH 9 dari tempatnya, dan boot-lah komputer sesuai petunjuk dari program instalasi.

## **2.1. Apakah Linux itu?**

Linux merupakan sebuah sistem operasi dengan kemampuan *multiuser* dan *multitasking* yang bersifat *free*. *Free* disini bukan diartikan sebagai gratis tetapi berarti kebebasan seperti yang tertuang dalam lisensi yang dimiliki oleh Linux yaitu GNU GPL (GNU is Not Unix General Public License). Perangkat lunak yang mempunyai lisensi GNU GPL akan menjamin 3 (tiga) kebebasan bagi para penggunanya, yaitu :

1. kebebasan untuk menjalankan perangkat lunak tersebut dengan tujuan apapun.
2. kebebasan untuk mengubah perangkat lunak tersebut agar sesuai dengan kebutuhan. Hal ini mengharuskan prasyarat tersedianya *source code* perangkat lunak tersebut.
3. kebebasan untuk mendistribusikan salinannya, baik secara gratis ataupun dengan biaya.

## **2.2. Sejarah Linux**

Pembuat sekaligus pemegang hak cipta Linux adalah Linus Bennedict Torvalds. Beliau pada saat itu adalah mahasiswa Universitas Helsinki, Finlandia yang terinspirasi oleh sebuah sistem operasi sejenis Unix yang dinamakan Minix. Walaupun pada awalnya pembuatan Linux dianggap suatu hobi tetapi setelah dilepas ke Internet ternyata mendapat tanggapan yang luar biasa dari programmer di seluruh dunia yang tergabung dalam suatu *newsgroup*. Selanjutnya Linux dikembangkan oleh Linus dibantu oleh programmer dari seluruh dunia melalui Internet.

## **2.3. Distribusi Linux**

Linux sendiri sebenarnya hanya sebuah kernel yang merupakan jantung dari suatu sistem operasi. Oleh karena itu kernel memerlukan program-program tambahan yang jalan diatasnya. Kernel Linux yang telah dilengkapi dengan program-program tambahan tersebut dikenal dengan sebutan distribusi atau distro. Distro-distro yang beredar dan terkenal di Indonesia diantaranya adalah Red Hat, Mandrake, Suse, Slackware, Debian dan sebagainya.

## **2.4. Penggunaan Linux**

Saat ini umumnya Linux lebih banyak digunakan sebagai suatu server baik itu sebagai mail server, web server, database server dan lain-lain. Selain dikenal lebih stabil juga keamanannya lebih terjamin. Disamping itu Linux bisa juga digunakan sebagai desktop untuk pekerjaan sehari-hari dengan aplikasi perkantoran yang biasanya sudah ada di setiap distribusinya.

## **2.5. Informasi tentang Linux**

Informasi tentang Linux umumnya banyak terdapat di Internet. Berikut beberapa diantaranya :

### **Informasi distro :**

<http://www.redhat.com>  
<http://www.mandrake.com>  
<http://www.suse.com>  
<http://www.slackware.com>  
<http://www.debian.org>

### **Belajar Linux:**

<http://www.linuxnewbie.org>  
<http://www.linuxdot.org>  
<http://www.linux.com>  
<http://pemula.linux.or.id>

### **Berita tentang Linux :**

<http://www.linuxtoday.com>  
<http://www.linuxnews.com>  
<http://www.linuxjournal.com>  
<http://www.linux.com>



## 2.6. Memulai Linux

Untuk dapat menggunakan Linux, user harus terlebih dahulu melakukan proses login. Proses login yaitu memasukkan nama login dan password yang sudah terlebih dahulu dibuat oleh administrator sistem. Proses login dapat dilakukan pada modus teks dan modus grafik.

### 2.6.1. Login pada Modus Teks

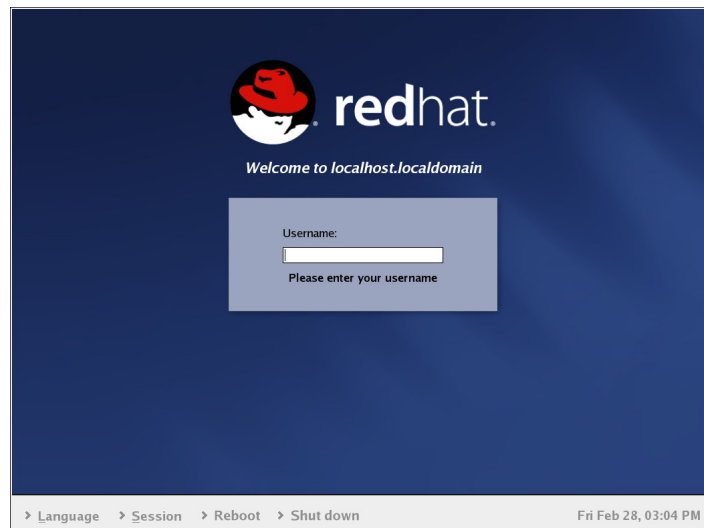
Tampilan login pada modus teks akan terlihat seperti dibawah ini.

```
Red Hat Linux release 9 (Shrike)
Kernel 2.4.20-8 on an i686

localhost login:
```

Masukkan nama login dan password pada tempat yang tersedia.

### 2.6.2. Login pada Modus Grafik



**Gambar 1-1. Login pada Modus Grafik**

Masukkan nama user pada kotak isian username, selanjutnya akan diminta untuk memasukkan password.

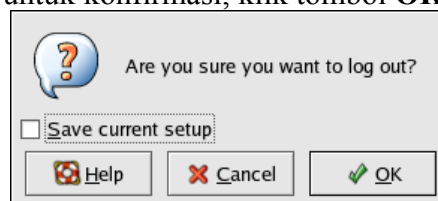
## 2.7. Keluar dari Sistem Linux

### 2.7.1. Logout pada Modus Teks

Untuk keluar dari sistem Linux pada modus teks, ketikkan `exit` pada **prompt** atau tekan tombol **[Ctrl] - [D]**.

### 2.7.2. Logout pada Modus Grafik

Untuk keluar dari sistem Linux pada modus grafik, klik **Main Menu** pada **Panel** kemudian pilih **Log Out**. Akan keluar jendela untuk konfirmasi, klik tombol **OK**.



**Gambar 1-2. Konfirmasi untuk Logout**



## 🐧 🖱️ Menggunakan Desktop

Red Hat Linux sudah memasukkan lingkungan grafis desktop ke dalam distribusinya sehingga sangat memudahkan penggunaan bagi pemula. Ada dua lingkungan desktop manager yang umum digunakan yaitu KDE dan GNOME yang merupakan desktop manager default di Red Hat Linux. Baik KDE maupun GNOME, di dalam distribusi Red Hat sudah diseragamkan tampilannya sehingga sedikit sekali perbedaannya.



Gambar 3-1. Desktop

### 3.1. Desktop

Contoh tampilan Desktop dapat dilihat pada Gambar 3-1.

Desktop terdiri dari **Panel**, **Desktop Icon** dan **Sistem Menu**. Di dalam desktop kita dapat melakukan *drag and drop* file atau aplikasi. Juga menambahkan **icon** untuk file atau aplikasi.

#### 3.1.1. Panel

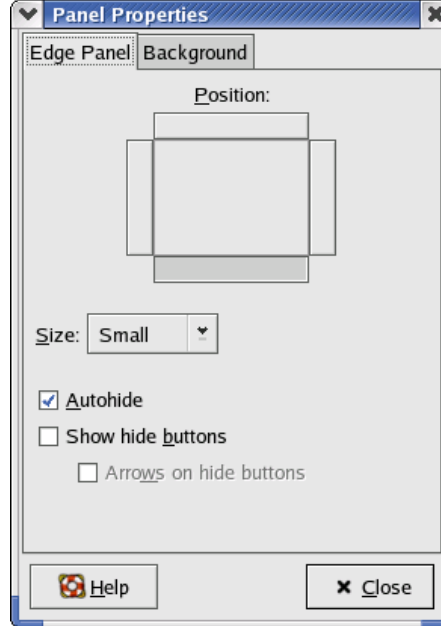


Gambar 3-2. Panel

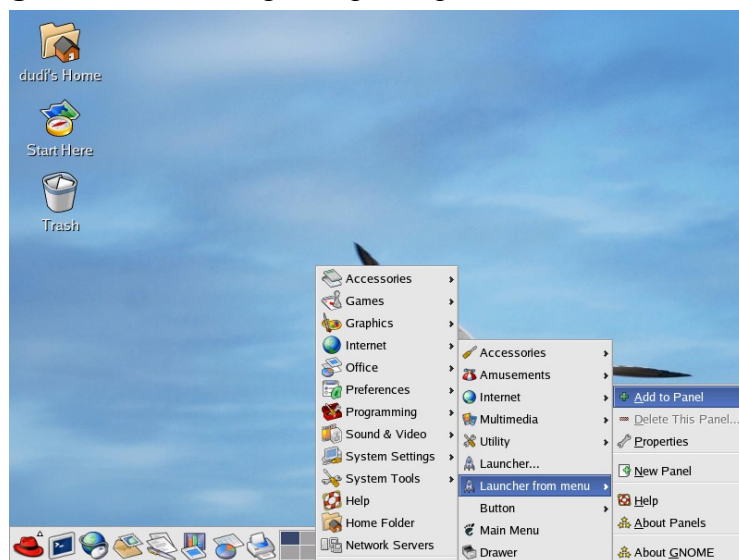
Panel merupakan kotak panjang kecil yang ada di bagian bawah desktop. Fungsinya untuk menyimpan **icon** dan aplikasi kecil yang disebut **applet** sehingga memudahkan penggunaan sistem.

Panel dapat disembunyikan secara manual atau otomatis, diubah ukuran, warna dan tampilannya. Untuk mengkonfigurasi Panel, ikuti langkah berikut:

1. Arahkan pointer mouse di daerah Panel yang kosong.
2. Klik tombol kanan mouse untuk menampilkan menu popup.
3. Pilih menu **Properties**. Akan muncul kotak dialog **Panel Properties** seperti Gambar 3-3.
4. Pilih:
  - tab **Edge Panel** untuk mengubah lokasi, ukuran dan tingkah laku Panel. Berikut penjelasan beberapa opsi yang terdapat pada tab **Edge Panel**:
    - **Position**. Untuk mengubah lokasi panel, klik lokasi yang diinginkan pada opsi ini.
    - **Size**. Untuk mengubah ukuran panel, klik menu drop down pada opsi ini kemudian klik ukuran yang diinginkan.
    - Agar panel bisa disembunyikan secara otomatis, beri tanda pada kotak cek **Autohide**.



- Apabila menginginkan panel disembunyikan secara manual, beri tanda pada kotak cek **Show hide buttons**.
- tab **Background** untuk mengubah penampilan Panel. Pada tab ini terdapat beberapa



opsi, yaitu:

- **Type**. Untuk menentukan tipe background. Tersedia pilihan **Default**, **Color**, **Image** dan **Transparent**.
- **Color**. Jika memilih **Color** pada opsi **Type**, tentukan warna yang diinginkan pada opsi ini.
- **Image**. Jika memilih **Image** pada opsi **Type**, tentukan image yang akan ditampilkan pada opsi ini.

**Gambar 3-3. Kotak Dialog Panel Properties**

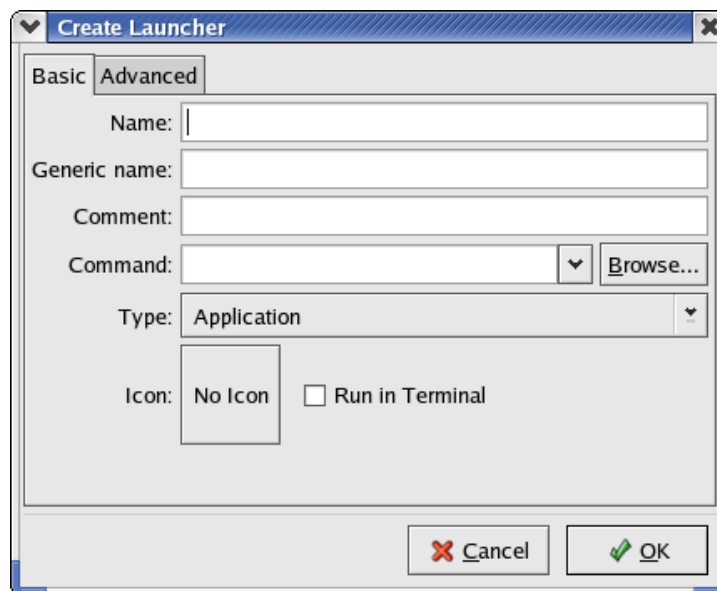
Untuk menambahkan icon ke Panel, ikuti langkah berikut :

1. Tempatkan pointer mouse di bagian Panel yang kosong.
2. Klik tombol kanan mouse untuk memunculkan menu popup.
3. Klik **Add to Panel > Launcher from Menu** > pilih aplikasi yang diinginkan.

**Gambar 3-4. Menambah icon ke Panel**

### 3.1.2. Desktop Icon

Icon-icon yang ada di desktop biasanya berupa *shortcut* ke suatu folder, aplikasi atau *shortcut* ke removable device seperti CD-ROM atau Floppy ketika telah di mount. Untuk membuka folder atau menjalankan aplikasi, klik icon yang bersangkutan.



Untuk merapikan icon pada desktop, ikuti langkah berikut:


1. Tempatkan pointer mouse di bagian desktop yang kosong.
2. Klik tombol kanan mouse untuk memunculkan menu popup.
3. Pada menu popup, pilih **Clean up by Name** untuk merapikan icon secara otomatis.

Untuk menambahkan icon ke desktop, ikuti langkah berikut :

1. Tempatkan pointer mouse di bagian desktop yang kosong.
2. Klik tombol kanan mouse untuk memunculkan menu popup.
3. Klik **New Launcher**, akan muncul kotak dialog **Create Launcher**.
4. Pada bagian **Name** masukkan nama aplikasi.
5. Pada bagian **Command** isikan nama file aplikasi lengkap dengan path-nya.
6. Pada bagian **Icon** klik tombol **No Icon** untuk menambahkan icon.
7. Klik tombol **OK** apabila telah selesai.

**Gambar 3-5. Kotak dialog Create Launcher**

### 3.1.3. Sistem Menu

Sistem menu dapat ditampilkan dengan melakukan klik tombol **Main Menu**  . Hampir semua aplikasi, perintah dan tool konfigurasi dapat diakses dari **Main Menu**.

## 3.2. Background Desktop

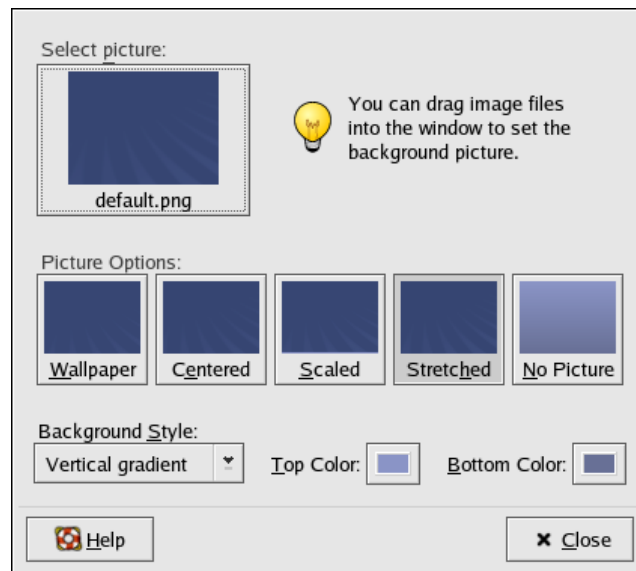
Background Desktop dapat diubah dengan menggunakan tool **Background Preferences**. Lihat Gambar 3-4. Untuk memunculkan jendela **Background Preferences**, dapat melakukan salah satu cara berikut ini:

- Klik **Main Menu > Preferences > Background**
- Klik kanan mouse di daerah desktop, kemudian pilih **Change Desktop Background** dari menu.

Pada jendela **Background Preferences** tersedia pilihan sebagai berikut :

- **Select picture.** Pilih image yang akan ditampilkan sebagai background. Untuk memilih image, klik tombol **Picture**, akan muncul pilihan image yang secara default disimpan di direktori `/usr/share/backgrounds/images`.
- **Picture options.** Untuk menentukan bagaimana image ditampilkan, biasanya berguna apabila image berukuran kecil. Pilih salah satu dari yang berikut ini:
  - **Wallpaper:** image akan ditampilkan secara berulang hingga menutupi seluruh desktop.

- **Centered**: image akan ditampilkan tepat di tengah desktop.
- **Scaled**: image akan diperbesar hingga mencapai tepi layar secara proporsional.
- **Strechted**: image akan diperbesar hingga menutupi seluruh layar.
- **No Picture**: tidak menampilkan image.
- **Background Style**: digunakan apabila ingin mendesain warna sendiri dan memilih **No Picture** pada **Picture options**.
- Klik tombol **close** untuk menyimpan dan keluar dari tool **Background Preferences**.



**Gambar 3-6. Background Preferences**

### 3.3. Lock Screen

Lock Screen (mengunci layar) berguna apabila anda ingin meninggalkan desktop dan tidak ingin ada orang lain yang menggunakan desktop.

Untuk mengaktifkan lock screen caranya :

1. Klik **Main Menu > Lock Screen**
2. Apabila lock screen aktif maka akan mengaktifkan screensaver. Untuk menon-aktifkan lock screen, masukkan username dan password pada dialog lock screen kemudian tekan tombol **[Enter]**.

### 3.4. Screensaver

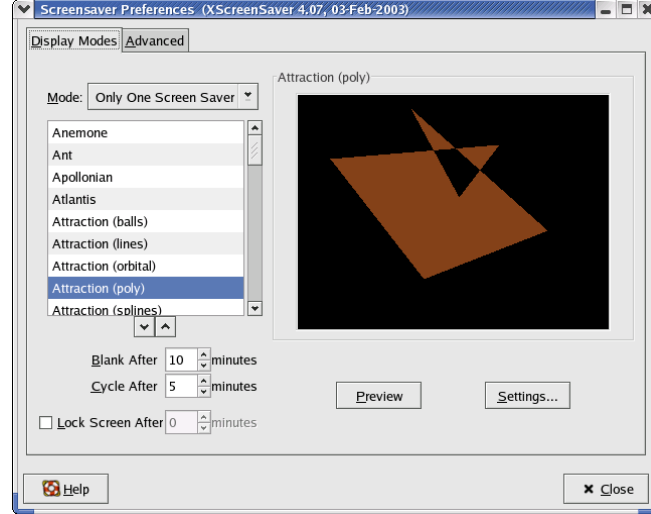
Screensaver adalah aplikasi yang akan ditampilkan apabila layar tidak digunakan selama waktu tertentu. Untuk mengkonfigurasi screensaver, ikuti langkah berikut :

1. Klik **Main Menu > Preferences > Screensaver**. Akan muncul kotak dialog **Screensaver Preferences**.
2. Pilih tab **Display Modes**.

Berikut penjelasan opsi yang ada pada tab **Display Modes**.

**Mode**. Ada empat pilihan mode screensaver, yaitu:

- **Disable Screen Saver**, pilih mode ini untuk menonaktifkan screensaver.
- **Blank Screen Only**, pilih mode ini untuk mengaktifkan screensaver dengan mode blank screen.
- **Only One Screen Saver**, pilih mode ini untuk mengaktifkan sebuah screensaver kemudian pilih aplikasi screensaver yang diinginkan pada daftar pilihan.
- **Random Screen Saver**, pilih mode ini untuk mengaktifkan lebih dari satu screensaver kemudian pilih aplikasi screensaver yang diinginkan pada daftar pilihan.



**Gambar 3-7. Kotak Dialog Screensaver Preferences**

## Shell Prompt

Shell berfungsi sebagai *command interpreter*, yaitu menterjemahkan perintah yang diberikan dari prompt dan juga dapat digunakan sebagai bahasa pemrograman, yaitu melaksanakan sekumpulan perintah yang dibaca dari suatu file. Sebagian pengguna Linux masih banyak yang menyukai bekerja di lingkungan shell prompt dikarenakan lebih cepat dibandingkan bekerja di lingkungan grafis (GUI).

### 4.1. Mengakses Shell Prompt

Untuk mengakses shell prompt caranya adalah klik **Main Menu > System Tools > Terminal**.

### 4.2. Shell bash

Shell default di sistem Linux adalah bash yang merupakan singkatan dari Bourne Again Shell. Prompt untuk shell bash ditandai dengan simbol `$`. Shell bash menggabungkan fitur-fitur dari shell yang ada di sistem Unix yaitu sh, csh dan ksh. Fitur-fitur yang dimiliki oleh shell bash diantaranya adalah :

#### 1. alias

alias biasanya digunakan untuk menyingkat suatu perintah yang panjang.

Sintaks : `alias nama_alias=perintah`

Contoh :

```
$ alias cls=clear
$ alias ls="ls -l"
```

#### 2. history

Anda dapat menampilkan kembali perintah-perintah yang sudah diketikkan dari prompt dengan cara menekan tombol panah atas atau tombol panah bawah.

#### 3. command line editing

Apabila anda salah mengetikkan suatu perintah maka perintah tersebut masih dapat diedit dengan menggunakan tombol panah kiri atau tombol panah kanan.

#### 4. command line completion

Fungsi ini berguna untuk mempermudah pengetikan perintah yang panjang.

Caranya dengan menekan tombol **[TAB]** pada saat mengetikkan perintah.

### 4.3. File Startup

File startup adalah file-file yang akan dieksekusi ketika shell bash dijalankan. File-file itu adalah:

- `/etc/profile` : File ini akan dieksekusi pada urutan pertama. Isinya sebagian besar berupa pendefinisian variabel lingkungan dan berlaku global untuk seluruh sistem.
- `~/.bash_profile` : File ini berada di direktori home user. Fungsinya sama dengan file `/etc/profile` hanya berlaku untuk user yang bersangkutan.
- `~/.bashrc` : File ini berada di direktori home user. Isinya berupa pendefinisian alias dan fungsi.
- `~/.bash_logout` : File ini berada di direktori home user. Isi file ini akan dieksekusi pada saat keluar dari shell (logout).

### 4.4. Perintah Shell

Sintaks perintah secara umum yaitu :

`perintah [-pilihan...] [argumen...]`

Keterangan :

- antara perintah dan pilihan harus ada spasi.
- antara pilihan dan argumen harus ada spasi.
- tanda kurung siku ([ ]) berarti bahwa pilihan dan argumen tidak harus selalu ada.
- tanda titik tiga ( . . . ) berarti bahwa pilihan dan argumen bisa lebih dari satu.
- setiap perintah di Linux membedakan huruf besar dan kecil (*case sensitive*).
- setiap pengetikan perintah selalu diakhiri dengan menekan tombol **[Enter]**.

#### 4.4.1. **ls - list**

Fungsi : untuk menampilkan isi sebuah direktori.

Pilihan yang sering digunakan :

- a menampilkan seluruh isi direktori termasuk file yang hidden
- d menampilkan atribut direktori bukan isinya
- l menampilkan isi direktori lengkap dengan atributnya
- R menampilkan isi subdirektori

Contoh :

```
$ ls
$ ls -l
$ ls -la
```

#### 4.4.2. **passwd**

Fungsi : untuk mengubah password user

Contoh :

```
$ passwd [Enter]
Changing password for dudi
(current) UNIX password:
New UNIX password:
Retype new UNIX password:
passwd: all authentications tokens updated successfully
```

**Catatan : Password yang anda ketikkan tidak akan ditampilkan.**

#### 4.4.3. **more**

Fungsi : sebagai pager, menampilkan *output* per layar

Navigasi tombol pada perintah more :

- [Enter] scroll down per baris
- Space Bar scroll down per layar
- q keluar dari more

Contoh :

```
$ ls -l /etc | more
```

#### 4.4.4. **less**

Fungsi : sama dengan more

Navigasi tombol pada perintah less :

- Panah Bawah scroll down per baris
- Panah Atas scroll up per baris
- [Enter] scroll down per baris
- Space Bar scroll down per layar
- q keluar dari less

Contoh :

```
$ ls -l /etc | less
```

#### 4.4.5. **man - manual page**

Fungsi : menampilkan dokumentasi atau manual suatu perintah

Contoh :

```
$ man ls
```

#### 4.4.6. **w dan who**

Fungsi : menampilkan user yang sedang login saat ini

Contoh :

```
$ w
$ who
```

#### 4.4.7. **cat - concatenate**

Fungsi :

1. membuat file sederhana

Contoh :

```
$ cat > latihan
saya sedang belajar
[Ctrl]+[D]
```

2. melihat isi suatu file

Contoh :

```
$ cat latihan
saya sedang belajar
```

3. menambah isi suatu file

Contoh :

```
$ cat >> latihan
Linux Fundamental
[Ctrl]+[D]
$ cat latihan
saya sedang belajar
Linux Fundamental
```

#### 4.4.8. **pwd - print working directory**

Fungsi : menampilkan posisi direktori kerja saat ini

Contoh :

```
$ pwd
/home/dudi
```

#### 4.4.9. **cd - change directory**

Fungsi : untuk melakukan pindah direktori

Catatan :

Jika perintah **cd** dijalankan tanpa argumen, maka akan mengembalikan posisi direktori ke direktori home user.

Contoh :

```
$ cd /etc
$ pwd
/etc
```



```
$ cd
$ pwd
/home/dudi
```

#### 4.4.10. **mkdir** - make directory

Fungsi : membuat direktori

Contoh :

```
$ mkdir data
```

#### 4.4.11. **cp** - copy

Fungsi : membuat salinan suatu file

Contoh :

```
$ cp latih coba
```

#### 4.4.12. **mv** - move

Fungsi :

1. merubah nama file (*rename*)

Contoh :

```
$ mv coba uji
```

2. memindahkan file ke direktori

Catatan : argumen kedua harus berupa suatu direktori

Contoh :

```
$ mv latih data
```

#### 4.4.13. **rm** - remove

Fungsi :

1. menghapus file

Contoh :

```
$ rm uji
```

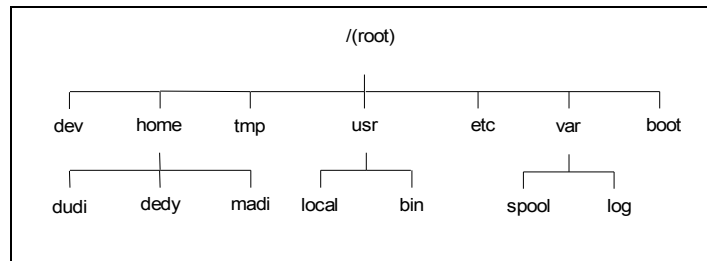
2. menghapus direktori

Contoh :

```
$ rm -r data
```

## File dan Direktori

Struktur direktori pada sistem Linux sama dengan struktur direktori pada Windows yaitu menerapkan sistem pohon (*tree*) terbalik. Akar direktori atau direktori root yang dilambangkan dengan garis miring (/) merupakan tingkat direktori tertinggi (*top level directory*). Contoh struktur direktori sistem Linux dapat dilihat pada Gambar 5-1.



**Gambar 5-1. Struktur Direktori Linux**

### 5.1. Direktori Utama Sistem Linux

| <i>Direktori</i> | <i>Penjelasan</i>   |
|------------------|---|
| /                | Direktori root. Merupakan direktori tertinggi pada susunan direktori Linux. Seluruh direktori di Linux berada di bawah direktori root ini.<br>Pada Windows, root direktori dilambangkan dengan C:\>.  |
| /bin             | Direktori root tidak ada hubungannya dengan user root. Pada Linux memang terdapat dua istilah root yang berbeda, yang pertama adalah user root, dan yang kedua adalah direktori root.<br>Direktori tempat perintah-perintah dasar Linux yang bisa dijalankan oleh semua user.                           |
| /boot            | Direktori tempat file yang digunakan untuk proses boot, termasuk kernel Linux itu sendiri..   |
| /dev             | Direktori tempat semua file device. File device adalah file yang mewakili peralatan tertentu pada komputer. Harap diingat bahwa di Linux segala sesuatu dianggap sebagai file.  |
| /etc             | Direktori tempat file-file konfigurasi Linux yang sangat penting. File konfigurasi Linux umumnya berbentuk file teks.   |
| /home            | Direktori utama dari direktori home. Direktori home adalah direktori pribadi user kecuali user root. Setiap user bisa memiliki direktori pribadi di bawah direktori /home. Umumnya nama direktori home sesuai dengan nama user bersangkutan. Contoh: user dudi akan memiliki direktori home /home/dudi. |
| /root            | Direktori home milik user root.   |
| /lib             | Direktori tempat program-program library.   |
| /lost+found      | Direktori tempat menyimpan file-file hasil perbaikan saat sistem bermasalah (crash atau masalah lain). Di Windows, umumnya file-file seperti ini diberi perluasan (ekstensi) *.chk.   |
| /opt             | Direktori tempat menyimpan software-software tambahan yang berukuran besar, seperti StarOffice.   |
| /proc            | Direktori ini berisi file-file yang merepresentasikan berbagai aspek dari sistem komputer seperti prosesor, memori dan lain-lain.   |
| /sbin            | Direktori tempat berbagai utiliti atau program yang hanya bisa dijalankan oleh user root atau superuser.  |
| /tmp             | Direktori tempat menyimpan file-file sementara yang digunakan oleh berbagai   |

## Direktori

## Penjelasan

|      |   |
|------|---|
|      | macam program.  |
| /usr | Direktori ini berisi banyak direktori yang sangat penting.  |
| /var | Direktori tempat file-file yang digunakan untuk administrasi seperti system logs dan data yang sering kali berubah. |

### 5.2. Home Directory

Merupakan direktori tertentu yang diberikan khusus kepada pemakai sebagai tempat penyimpanan file-file kepunyaan pemakai tersebut. Struktur direktori dibawah *home directory* merupakan wewenang dari pemakai yang bersangkutan.

### 5.3. Current & Parent Directory

*Current directory* (.) adalah direktori dimana kita berada pada saat ini sedangkan *Parent directory* (..) adalah direktori satu tingkat di atas *current directory*. Setiap kali direktori baru tercipta, otomatis akan dibuat *current directory* dan *parent directory*

### 5.4. Absolute Path dan Relative Path

Absolute path merupakan penulisan path yang mengacu ke *root directory* (/) jadi selalu diawali dengan tanda /.

Contoh :

```
/home/dudi/latih.txt  
/var/log/messages
```

Relative path merupakan penulisan path yang mengacu ke *current directory*.

Contoh :

```
./latih.txt  
../../etc/passwd
```

### 5.5. Penamaan Suatu File/Direktori

- Maksimum 256 karakter
- Alfanyumerik
- Tanda titik yang mengawali nama file mempunyai arti bahwa file tersebut hanya akan ditampilkan jika menggunakan option -a pada perintah ls (*hidden file*)
- Semua karakter bisa digunakan sebagai nama file tetapi sebaiknya jangan menggunakan karakter-karakter di bawah ini :

& ; | ? ` " ' ~ [ ] ( ) \$ < > { } ^ # \ /

- Penamaan file pada sistem Linux bersifat *case sensitive*

### 5.6. Atribut File/Direktori

|                 |           |       |       |               |         |       |       |
|-----------------|-----------|-------|-------|---------------|---------|-------|-------|
| -rw-rw-r--      | 1         | dudi  | dudi  | 24343         | Dec 03  | 09:44 | latih |
| Tipe File       | Hard Link | Owner | Group | Ukuran (byte) | Tanggal | Waktu | Nama  |
| Permission File |           |       |       |               |         |       |       |

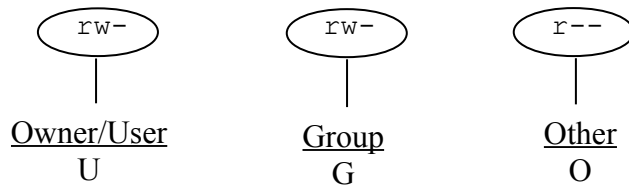
#### 5.6.1. Tipe File

Tipe file terdiri dari :

| Simbol | Arti |
|--------|------|
|--------|------|

|   |                         |
|---|-------------------------|
| - | File biasa              |
| d | Direktori               |
| l | Symbolic Link           |
| b | <i>Block Device</i>     |
| c | <i>Character Device</i> |

### 5.6.2. Permission File / Ijin akses



Tiga digit pertama digunakan untuk mengatur ijin akses bagi owner. Tiga digit kedua digunakan untuk mengatur ijin akses bagi user lain yang masih satu group dengan owner sedangkan tiga digit ketiga digunakan untuk mengatur ijin akses bagi user lain diluar group owner.

#### Ijin Akses untuk File

r : read : ijin untuk membaca/melihat isi suatu file (contoh: \$ cat latihan)  
w : write : ijin untuk mengubah isi suatu file (contoh: \$ cat >> latihan)  
x : execute : ijin untuk mengeksekusi/menjalankan suatu file dimana jenis file berupa *script/program* (contoh: \$ ./latih)

#### Ijin Akses untuk Direktori

r : read : ijin untuk melihat isi suatu direktori (contoh: \$ ls -l data)  
w : write : ijin untuk membuat/menghapus file dalam direktori (contoh: \$ rm data/latih)  
x : execute : ijin untuk pindah ke dalam direktori (contoh: \$ cd data)

### Merubah *Permission File* - **chmod**

Perintah **chmod** memiliki dua format :

#### 1. Format Huruf

Sintaks : **chmod** [augo] [+==] [rwx] file(s)

Keterangan :

a : all = ugo  
u : user  
g : group  
o : other  
+ : menambahkan permission  
- : menghilangkan permission  
= : set

Contoh :

```
$ ls -l berkasY
-rw-rw-r-- 1 boim rumpi 24343 Dec 03 09:44 berkasY
$ chmod o+w berkasY
$ ls -l berkasY
-rw-rw-rw- 1 boim rumpi 24343 Dec 03 09:44 berkasY
$ chmod u=rwx,g=r,o=r berkasY
$ ls -l berkasY
-rwxr--r-- 1 boim rumpi 24343 Dec 03 09:44 berkasY
```

## 2. Format Angka

Sintaks : `chmod [kode_oktal] file(s)`

|     |          |     |   |          |
|-----|----------|-----|---|----------|
|     | read (r) | =   | 4 | Notasi : |
|     | write    | =   | 2 |          |
| rw- | (w)      | rw- |   | r--      |
| 420 | execute  | 420 | 1 | 400      |
| 6   | (x)      | 6   |   | 4        |

Contoh :

```
$ ls -l berkasY <Enter>
-rw-rw-r-- 1 boim rumpi 24343      Dec 03  09:44  berkasY
$ chmod 751 berkasY <Enter>
$ ls -l berkasY <Enter>
-rwxr-x--x 1 boim rumpi 24343      Dec 03  09:44  berkasY
```

### File Creation Mask - `umask`

Setiap kali file/direktori dibuat, maka permission-nya sesuai dengan *mask* yang ditentukan.

Khusus untuk file, *mask* untuk x tidak ada

Sintaks : `umask [kode_oktal]`

Kode oktal untuk File :

|         |     |         |     |
|---------|-----|---------|-----|
| 0 = rw- | 000 | 4 = -w- | 100 |
| 1 = rw- | 001 | 5 = -w- | 101 |
| 2 = r-- | 010 | 6 = --- | 110 |
| 3 = r-- | 011 | 7 = --- | 111 |

Keterangan :

Untuk file *mask* untuk x tidak ada.

Kode oktal untuk Direktori :

|         |     |         |     |
|---------|-----|---------|-----|
| 0 = rwx | 000 | 4 = -wx | 100 |
| 1 = rw- | 001 | 5 = -w- | 101 |
| 2 = r-x | 010 | 6 = --x | 110 |
| 3 = r-- | 011 | 7 = --- | 111 |

Contoh :

```
$ umask
022
$ umask 077
$ touch latih
$ ls -l latih
-rw----- 1 boim rumpi 24343      Dec 03  09:44  latih
$ mkdir data
$ ls -ld data
drw----- 1 boim rumpi 24343      Dec 03  09:44  data
```

### 5.6.3. Hard Link

Kolom ini menunjukkan jumlah hard link. Untuk file biasa jumlah linknya satu sedangkan untuk direktori tanpa subdirektori jumlah link-nya dua. Setiap membuat subdirektori maka jumlah link-nya akan bertambah satu.

#### 5.6.4. Owner

Menunjukkan owner atau pemilik yaitu yang membuat file tersebut. Untuk mengubah owner suatu file digunakan perintah `chown`.

Sintaks : `chown owner nama_file`

Contoh :

```
# chown dodo latihan
```

Keterangan :

Yang bisa menjalankan perintah `chown` hanya user root.

#### 5.6.5. Group

Menunjukkan ke dalam group apa user tersebut dikelompokkan. Untuk mengubah group suatu file digunakan perintah `chgrp`.

Sintaks : `chgrp group nama_file`

Contoh :

```
# chgrp staf latihan
```

#### 5.6.6. Ukuran

Menunjukkan besarnya ukuran file dalam satuan byte.

#### 5.6.7. Tanggal

Menunjukkan tanggal terakhir file tersebut diakses.

#### 5.6.8. Jam

Menunjukkan jam terakhir file tersebut diakses.

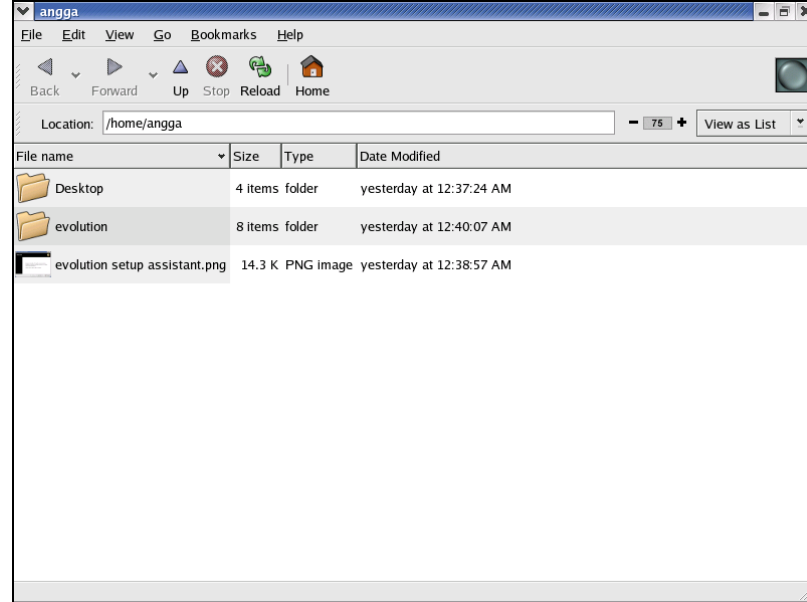
#### 5.6.9. Nama

Menunjukkan nama file yang bersangkutan.

### 5.7. Nautilus File Manager

#### 5.7.1. Menjalankan Nautilus

Untuk menjalankan **Nautilus**, klik icon **Home directory** pada Desktop atau klik **Main Menu > Home Folder**. Tampilan **Nautilus** dapat Anda lihat pada Gambar 5-2.



**Gambar 5-3. Tampilan Nautilus**

### 5.7.2. Mengubah Tampilan Nautilus

**Nautilus** menyediakan dua buah bentuk tampilan file dan direktori, yaitu : Icon View dan List View. Untuk mengubah bentuk tampilan file dan direktori, pilih menu **View**, lalu pilih salah satu dari **View as Icons** atau **View as List**.

### 5.7.3. Membuat Direktori

Untuk membuat direktori baru, ikuti langkah berikut :

1. Pilih direktori dimana direktori baru akan dibuat.
2. Pilih menu **File > New Folder**, akan muncul direktori baru dengan nama **untitled folder**.
3. Ketikkan nama untuk direktori yang baru lalu tekan tombol **[Enter]**.

### 5.7.4. Mengubah Nama File atau Direktori

Untuk mengubah nama file atau direktori, ikuti langkah berikut :

1. Pilih file atau direktori yang akan diubah namanya.
2. Pilih menu **Edit > Rename** atau klik kanan tombol mouse lalu pada menu popup pilih **rename**.
3. Ketikkan nama file atau direktori yang baru kemudian tekan tombol **[Enter]**.

### 5.7.5. Memindahkan File atau Direktori

Untuk memindahkan file atau direktori, ikuti langkah berikut :

1. Pilih file atau direktori yang akan dipindahkan.
2. Pilih menu **Edit > Cut File** atau klik kanan tombol mouse lalu pada menu popup pilih **Cut File**.
3. Pilih direktori yang akan menjadi tempat untuk file atau direktori yang dipindahkan.
4. Pilih menu **Edit > Paste Files** atau klik kanan tombol mouse lalu pada menu popup pilih **Paste Files**.

### 5.7.6. Menyalin File atau Direktori

Untuk menyalin file atau direktori, ikuti langkah berikut :

1. Pilih file atau direktori yang akan disalin.
2. Pilih menu **Edit > Copy File** atau klik kanan tombol mouse lalu pada menu popup pilih **Copy File**.
3. Pilih direktori yang akan menjadi tempat untuk file atau direktori yang disalin.
4. Pilih menu **Edit > Paste Files** atau klik kanan tombol mouse lalu pada menu popup pilih **Paste Files**.

### 5.7.7. Menghapus File atau Direktori

Untuk menghapus file atau direktori, ikuti langkah berikut :

1. Pilih file atau direktori yang akan dihapus.
2. Pilih menu **Edit > Delete** atau klik kanan tombol mouse lalu pada menu popup pilih **Delete**. Menu **Delete** akan muncul apabila Anda memilih pilihan **Include a Delete command that bypasses Trash** pada kotak dialog **Preferences**.

#### **5.7.8. Memindahkan File dan Direktori ke Direktori Trash**

Untuk memindahkan file atau direktori ke direktori Trash, ikuti langkah berikut :

1. Pilih file atau direktori yang akan dipindahkan ke direktori **Trash**.
2. Pilih menu **Edit > Move to Trash** atau klik kanan tombol mouse lalu pada menu popup pilih **Move to trash**.



## Editor Teks vi

Pada suatu sistem operasi editor teks mempunyai peranan yang cukup penting diantaranya digunakan untuk mengedit file konfigurasi, membuat dokumentasi dan membuat program. *vi* merupakan editor teks standar yang digunakan di lingkungan Linux dan Unix.

### 6.1. Memulai vi

Untuk memulai *vi* ketikkan salah satu perintah berikut :

```
$ vi                : Memulai vi tanpa membuka file
$ vi nama_file      : Memulai vi dengan membuka file nama_file
$ vi -r nama_file   : Memulai vi dengan membuka file recover setelah terjadi crash
```

### 6.2. Modus Operasi

Modus operasi pada editor *vi* terdiri dari dua macam, yaitu :

1. Modus instruksi : Pada modus ini setiap tombol yang ditekan akan diterjemahkan sebagai suatu perintah. Untuk masuk ke modus ini tekan tombol [Esc].
2. Modus edit : Digunakan pada waktu penulisan teks. Untuk masuk ke modus ini tekan salah satu tombol : i, I, a, A, o atau O.

### 6.3. Pergerakan Kursor

Untuk menggerakkan kursor selain menggunakan tombol panah juga dapat menggunakan perintah-perintah sebagai berikut:

```
h : Digunakan untuk menggerakkan kursor ke kiri.
j : Digunakan untuk menggerakkan kursor ke bawah.
k : Digunakan untuk menggerakkan kursor ke atas.
l : Digunakan untuk menggerakkan kursor ke kanan.
```

### 6.4. Menyisipkan Teks

Pada waktu pertama kali membuka sebuah file, maka anda akan berada pada modus instruksi. Untuk memasuki modus edit dapat menggunakan tombol-tombol sebagai berikut :

```
i : Digunakan untuk menyisipkan teks mulai dari posisi kursor.
I : Digunakan untuk menyisipkan teks di awal baris.
a : Digunakan untuk menyambung teks setelah posisi kursor.
A : Digunakan untuk menyambung teks pada akhir baris.
o : Digunakan untuk menyisipkan baris kosong di bawah posisi kursor.
O : Digunakan untuk menyisipkan baris kosong pada posisi kursor.
```

### 6.5. Menghapus Teks

```
nx : Menghapus n karakter mulai dari posisi kursor. Fungsinya sama dengan tombol [Del].
nX : Menghapus n karakter di sebelah kiri kursor. Fungsinya sama dengan tombol [Backspace].
ndw : Menghapus sebanyak n kata dari posisi kursor.
ndd : Menghapus sebanyak n baris dari posisi kursor.
```

### 6.6. Menyimpan File dan Keluar dari vi.

Untuk menyimpan file dapat digunakan perintah-perintah sebagai berikut :

```
:w : Digunakan untuk menyimpan file.
:w! : Digunakan untuk menyimpan file walaupun filenya sudah ada.
```

- `:wq` : Digunakan untuk menyimpan file lalu keluar dari `vi`.
- `:x` : Digunakan untuk menyimpan file lalu keluar dari `vi`.
- `:q!` : Digunakan untuk keluar dari `vi` tanpa menyimpan file.

## 6.7. Copy, Cut, Paste

Untuk melakukan **cut & paste** atau **copy & paste**, anda dapat melakukannya dengan perintah berikut:

- `nyy` : Digunakan untuk menyalin `n` baris pada posisi kursor ke buffer.
- `nyw` : Digunakan untuk menyalin `n` kata pada posisi kursor ke buffer.
- `P` : Digunakan untuk menyalin data yang ada di buffer (termasuk teks yang baru dihapus dengan menggunakan perintah `x`, `dw`, dan `dd`) pada posisi kursor.
- `p` : Digunakan untuk menyalin data yang ada di buffer (termasuk teks yang baru dihapus dengan menggunakan perintah `x`, `dw`, dan `dd`) setelah posisi kursor.

## 6.8. Penggantian Teks

Untuk mengganti teks anda dapat menggunakan perintah-perintah sebagai berikut:

- `r` : Digunakan untuk mengganti karakter pada posisi kursor.
- `R` : Digunakan untuk mengganti karakter pada posisi kursor sampai tombol `[Esc]` ditekan.
- `ncw` : Digunakan untuk mengganti teks per kata.
- `ncc` : Digunakan untuk mengganti teks per baris.

## 6.9. Membatalkan perintah sebelumnya - undo

Setelah melakukan perintah pada modus instruksi anda dapat membatalkannya dengan menekan tombol:

- `u` : Digunakan untuk membatalkan satu perintah yang telah kita lakukan sebelumnya.
- `2u` : Digunakan untuk membatalkan dua perintah yang telah kita lakukan sebelumnya.

## 6.10. Membuka dokumen lain sewaktu mengedit file

Untuk membuka dokumen lain pada waktu anda mengedit suatu dokumen dapat menggunakan perintah (hanya dapat membuka satu dokumen):

- `:e nama_file` : Digunakan untuk membuka file *nama\_file* sewaktu sedang mengedit dokumen lain (hanya dapat digunakan jika file yang anda edit telah di simpan).
- `:e! nama_file` : Digunakan untuk membuka file *nama\_file* sewaktu sedang mengedit dokumen lain tanpa menyimpan perubahan yang anda lakukan.

## 6.11. Keluar sementara dari vi dan mengeksekusi perintah

Pada waktu mengedit dokumen, anda dapat melakukan beberapa perintah Linux yang lain dengan menggunakan perintah sebagai berikut:

- `!:bash` : Digunakan untuk keluar sementara dari `vi` ke shell, lalu kembali ke `vi` dengan mengetikkan `exit`.
- `!:ls` : Digunakan untuk keluar sementara dari `vi`, mengeksekusi perintah `ls`, dan kembali ke `vi`.

## 6.12. Pencarian String dalam vi

### 6.12.1. Pencarian String

Perintah-perintah yang dapat kita dilakukan adalah :

- `/string` : Digunakan untuk mencari "*string*" dengan arah pencarian maju (*forward*).

? : Digunakan untuk mencari "*string*" dengan arah pencarian mundur (*backward*).  
*string*  
n Mengulangi pencarian dengan arah yang sama dengan sebelumnya.  
N Mengulangi pencarian dengan arah kebalikan dari sebelumnya.

### 6.12.2. Pattern Matching dalam Pencarian String

| Pola  | Arti  | Contoh  | Keterangan  |
|-------|---|---------|---|
| .     | <i>Match</i> satu karakter                              | /p./    | Mencari karakter yang didahului huruf p   |
| ^     | <i>Match</i> awal baris                                 | /^ini/  | Mencari kata "ini" pada awal baris  |
| \$    | <i>Match</i> akhir baris                                | /ini\$/ | Mencari kata "ini" pada akhir baris   |
| \<    | <i>Match</i> awal kata                                  | /\<i/   | Mencari kata yang diawali huruf "i"   |
| \>    | <i>Match</i> akhir kata                                 | /u\>/   | Mencari kata yang diakhiri huruf "u".   |
| [...] | <i>Match</i> pilihan karakter                           | /[0-9]  | Mencari numerik   |
| *     | <i>Match</i> setiap karakter (termasuk karakter kosong) | /ada*   | Mencari kata yang ada string "ad" ( <u>jad</u> wal, <u>ada</u> , <u>pada</u> , <u>adalah</u> , <u>adapun</u> ). |

### 6.13. Mengedit Teks dengan Menggunakan Perintah Baris Akhir

Untuk melakukan perintah baris akhir anda harus ada dalam modus instruksi dan selalu diawali dengan mengetikkan karakter titik dua (:). Contoh-contoh penggunaannya antara lain :

|                       |   |   |
|-----------------------|---|---|
| :s/linux/LINUX/       | : | Mengganti sebuah kata linux dengan LINUX pada posisi kursor             |
| :s/linux/LINUX/g      | : | Mengganti semua kata linux dengan LINUX pada posisi kursor              |
| :1,\$ s/linux/LINUX/g | : | Mengganti semua kata linux dengan LINUX di seluruh dokumen              |
| :s/[Ll]nux/LINUX/     | : | Mengganti sebuah kata Linux atau linux dengan LINUX pada posisi kursor. |
| :1,3w test.txt        | : | Menyimpan baris 1 sampai 3 ke file test.txt                             |
| :2 r test.txt         | : | Menyisipkan isi file test.txt di bawah baris ke 2                       |
| :1,\$co \$            | : | Meng-copy semua isi file ke baris terakhir                              |
| :4m 1                 | : | Memindahkan baris ke 4 di bawah baris pertama.                          |
| :/^\$/d               | : | Menghapus baris kosong  |
| :g/LINUX/d            | : | Menghapus seluruh baris yang mengandung kata LINUX                      |
| :1,\$ s/linux//g      | : | Menghapus seluruh kata linux  |
| :1,\$ s/[^0-9]//g     | : | Menghapus seluruh alphabet  |
| :1,\$ s/[A-Z]/\l&/g   | : | Mengganti semua huruf besar menjadi huruf kecil                         |
| :1,\$ s/[a-z]/\u&/g   | : | Mengganti semua huruf kecil menjadi huruf besar                         |
| :1,\$ s/^/>>/g        | : | Menyisipkan >> pada setiap awal baris                                   |
| :1,\$ s/\$/<</g       | : | Menyisipkan << pada setiap akhir baris                                  |

## Pemrosesan String

Beberapa perintah Linux dapat digunakan untuk melakukan pemrosesan file teks seperti manipulasi file, pengurutan dan pencarian file dan string.

### 7.1. **wc - word count**

Fungsi : menghitung jumlah baris, kata dan karakter dalam suatu file

Sintaks : `wc [-lwc] file`

Keterangan :

- l : *line*/baris
- w : *word*/kata
- c : *character*/karakter

Contoh :

1. Menampilkan jumlah baris, kata dan karakter dari file `latih`

```
$ wc latihan
  24   133   861 latihan
```

2. Menampilkan jumlah baris dari file `latih`

```
$ wc -l latihan
 24 latihan
```

### 7.2. **cut**

Fungsi : mengambil bagian tertentu dari suatu file dan menampilkannya ke layar

Sintaks : `cut [-dchar] [-fc] file`

Keterangan :

- d : *delimiter*/pemisah antar *field*/kolom, digunakan jika menggunakan *option -f*  
Default = TABULASI
- f : *field*/kolom
- c : *character*/karakter

Contoh :

1. Menampilkan field pertama dan ketiga dari file `/etc/passwd`

```
$ cut -d: -f1,3 /etc/passwd
```

2. Menampilkan karakter ke-1 sampai dengan ke-8 dan karakter ke-25 sampai dengan ke-27 dari file `/etc/passwd`

```
$ cut -c1-8,25-27 /etc/passwd
```

3. Menampilkan karakter ke-12 sampai dengan ke-16 dari hasil keluaran perintah `date`

```
$ date | cut -c12-16
```

### 7.3. **head**

Fungsi : menampilkan n baris pertama isi suatu file

Sintaks : `head [-n] file`

Keterangan :

- n : jumlah baris yang akan ditampilkan  
Default = 10

Contoh :

1. Menampilkan 10 baris pertama file `/etc/passwd`

```
$ head /etc/passwd
```

2. Menampilkan 5 baris pertama file `/etc/passwd`

```
$ head -5 /etc/passwd
```

## 7.4. tail

Fungsi : menampilkan n baris terakhir isi suatu file

Sintaks : `tail [-n] file`

Keterangan :

`-n` : jumlah baris yang akan ditampilkan  
Default = 10

Contoh :

1. Menampilkan 10 baris terakhir file `/etc/passwd`

```
$ tail /etc/passwd
```

2. Menampilkan 5 baris terakhir file `/etc/passwd`

```
$ tail -5 /etc/passwd
```

3. Menampilkan mulai dari baris ke 5 sampai baris terakhir file `/etc/passwd`

```
$ tail +5 /etc/passwd
```

## 7.5. sort

Fungsi : mengurutkan isi suatu file berdasarkan *key* tertentu

Sintaks : `sort [-tchar] [+key] [-r] file`

Keterangan :

`-tchar` : *delimiter*/karakter pemisah antar *field*/kolom  
`+key` : nomor kolom sebagai *key* pengurutan, dimulai dari nol  
`-r` : reverse - mengurutkan secara descending

Contoh :

1. Mengurutkan isi file `/etc/passwd` berdasarkan kolom ke-1 secara *ascending*

```
$ sort /etc/passwd
```

2. Mengurutkan isi file `/etc/passwd` berdasarkan kolom ke-4 secara *ascending*

```
$ sort -t: +3 /etc/passwd
```

3. Mengurutkan isi file `/etc/passwd` berdasarkan kolom ke-4 secara *descending*

```
$ sort -t: +3 -r /etc/passwd
```

## 7.6. uniq

Fungsi : menghilangkan duplikasi data, syarat : data harus terurut

Sintaks : `uniq file`

Contoh :

```
$ cat > juice
apel
apel
melon
jeruk
apel
melon
[Ctrl]+[D]
$ uniq juice
apel
melon
jeruk
apel
melon
$ sort juice | uniq
apel
jeruk
melon
```

## 7.7. tr

Fungsi : untuk mengubah suatu karakter menjadi karakter lain

Sintaks : `tr "string1" "string2"`

Contoh :

1. Mengubah semua karakter l menjadi G, 3 menjadi &, dan s menjadi n

```
$ tr "l3s" "G&n" < latih
```

2. Mengubah semua huruf kecil menjadi huruf besar

```
$ tr "[a-z]" "[A-Z]" < latih
$ cat latih | tr "[a-z]" "[A-Z]"
```

## 7.8. grep

Fungsi : mencari string di dalam suatu file

Sintaks : `grep [-pilihan] string file`

Keterangan :

Pilihan yang sering digunakan :

- q : tidak menampilkan hasil keluaran ke layar
- v : menampilkan baris string yang tidak cocok

Contoh :

1. Menampilkan baris yang ada string Linux di file latih

```
$ grep Linux latih
```

2. Menampilkan seluruh file di *current directory* yang mempunyai string linux

```
$ grep linux *
```

3. Menampilkan baris yang tidak mempunyai string Linux di file latih

```
$ grep -v Linux latih
```

## 7.9. find

Fungsi : mencari file dengan kriteria tertentu

Sintaks : `find nama_path -pilihan`

Keterangan :

nama\_path = direktori awal pencarian

Pilihan yang sering digunakan :

- name *nama\_file* : cari file dengan nama *nama\_file*
- user *owner* : cari file dengan pemilik *owner*
- type *tipe\_file* : cari file dengan tipe *tipe\_file* (f=file,d=direktori,dll)
- perm *permission* : cari file dengan izin akses=*permission* dalam format angka
- size *kap\_size* : cari file dengan kapasitas = *kap\_size*
- ok *perintah* : eksekusi perintah dengan konfirmasi
- exec *perintah* : eksekusi perintah tanpa konfirmasi

Contoh :

1. Mencari file dengan nama latih mulai dari *current directory* ke bawah

```
$ find . -name latih
```

2. Mencari file yang dimiliki oleh user dudi mulai dari direktori / ke bawah

```
$ find / -user dudi
```

3. Mencari seluruh file dengan tipe direktori mulai dari direktori /home/dudi ke bawah

```
$ find /home/dudi -type d
```

4. Mencari seluruh file yang bernama core kemudian dihapus dengan konfirmasi mulai dari *current directory* ke bawah

```
$ find . -name core -ok rm {} \;
```

## Latihan Pemrosesan String

Buatlah file dengan nama `data_nama` dengan isi sebagai berikut :

|    |       |   |           |          |
|----|-------|---|-----------|----------|
| 10 | Habil | B | Bandung   | 555-9136 |
| 20 | Amir  | M | Jakarta   | 666-8888 |
| 50 | Yusuf | S | Depok     | 591-1191 |
| 60 | Siti  | Z | Tangerang | 511-1972 |
| 70 | Budi  | H | Bekasi    | 317-6512 |
| 80 | Yudi  | W | Bogor     | 777-2222 |

1. Hitung jumlah karakter dan baris dari file `data_nama`.
2. Urutkanlah file `data_nama` berdasarkan kolom ke 4.
3. Tampilkan isi file `data_nama` mulai dari baris ke-2 sampai baris ke-5.
4. Tampilkan baris yang terdapat karakter `D` atau `d` di file `data_nama`.
5. Tampilkanlah semua user yang sedang login pada sebuah sistem Linux.  
(hanya nama user login yang ditampilkan)
6. Carilah file yang mempunyai nama yang sama dengan login user mulai direktori `/var`.
7. Carilah file yang memiliki ijin akses `rw-r--r--` pada *home directory* user.
8. Tampilkan user yang login antara jam 11.00 sampai dengan jam 11:59, menggunakan perintah `who` dan `grep`.

# 🐧 📁 Shell Script I

*Shell script* adalah suatu *file* yang berisi perintah-perintah yang harus dikerjakan oleh Shell. Perintah dalam suatu *script* bisa berupa perintah yang dapat diberikan dari *prompt* atau perintah kendali alir (*Flow Control Commands*). Perintah kendali alir adalah perintah untuk mengatur urutan eksekusi perintah-perintah dalam suatu *script*. Perintah kendali alir terbagi dalam dua kelompok, yaitu pencabangan dan pengulangan.

## 8.1. Pengantar *Shell Script*

### 8.1.1. Membuat *Shell Script*

File script dapat dibuat dengan menggunakan teks editor seperti misalnya vi, pico, joe, gedit, kedit dll. Setiap file script biasanya diawali dengan baris seperti ini : `#!/bin/bash` yang artinya script di bawah baris tersebut akan dijalankan dengan interpreter `/bin/bash`. Untuk menandai baris komentar gunakan tanda `"#"` pada awal baris.

Contoh :

```
$ cat > myscript
#!/bin/bash
# ini adalah program pertama saya
# baris yang diawali dengan "#" adalah baris komentar
# dan tidak akan dieksekusi oleh shell
who
ls -l myscript
[Ctrl]+[D]
```

### 8.1.2. Menjalankan *Shell Script*

*Script* dijalankan seperti *executable file* dengan cara mengetikkan nama file pada *prompt*. Ada dua hal yang harus diperhatikan dalam menjalankan script, yaitu :

1. Direktori tempat script berada harus termasuk dalam *search PATH*  
Gunakan perintah `echo $PATH` untuk melihat *search PATH* yang berlaku saat ini
2. *File script* harus memiliki ijin akses (permission) r dan x.  
Ubah dengan perintah `chmod` jika perlu !

Contoh :

```
$ echo $PATH
/usr/local/bin:/bin:/usr/bin
$ cd
$ pwd
/home/badu
$ cat myscript
who
ls -l myscript
$ ls -l myscript
-rw-r--r--  1 badu      imt      16 Feb 23 11:29 myscript
$ myscript
bash : myscript : command not found
$ PATH=$PATH:$HOME

$ echo $PATH
/usr/local/bin:/bin:/usr/bin:/home/badu
$ myscript
bash : /home/badu/myscript : Permission denied
```

*file myscript tidak dapat ditemukan oleh shell memasukkan direktori tempat myscript berada ke dalam PATH*

*file myscript tidak bisa dieksekusi*



```
$ chmod +x myscript
$ ls -l myscript
-rwxr-xr-x  1 badu      imt      16 Feb 23 11:29 myscript
$ myscript
badu      ttyt0      Mar 01 12:47
-rwxr-xr-x  1 badu      imt      16 Feb 23 11:29 myscript
```

ditambahkan hak eksekusi

## 8.2. Variabel

Variabel merupakan nama yang mewakili suatu harga *string*. Nama variabel harus diawali dengan alfabet (a..z, A..Z) atau garis bawah/*underscore* (`_`) dan tidak boleh diawali dengan angka.

Contoh :

1. Penamaan variabel yang benar  
Nama  
hari  
\_jumlah
2. Penamaan variabel yang salah  
5hari  
2jum

### 8.2.1. Menyimpan Harga Variabel

Bentuk umum: Nama\_Var=Harga

Keterangan:

Tidak boleh ada spasi di kiri dan kanan "=".

Contoh :

```
$ jumlah=1
$ temp=/usr/unyil/temp
```

### 8.2.2. Menampilkan Harga Variabel

Perintah `echo` digunakan untuk menampilkan nilai yang disimpan dalam suatu variabel.

Bentuk umum : `echo $Nama_Var`

Keterangan:

\$ merupakan karakter khusus bagi *shell* yang menyatakan substitusi variabel.

\$Nama\_Var berarti substitusi harga yang disimpan dalam variabel bernama Nama\_Var pada posisi tersebut.

Contoh :

```
$ echo $jumlah
1
$ echo $temp
/usr/unyil/temp
```

### 8.2.3. Menghapus Variabel

Caranya : isi dengan NULL

Contoh :

```
$ nama= atau
$ nama='' atau
$ nama="" atau
$ unset nama
```

## 8.3. Parameter

### 8.3.1. Parameter Posisi (Positional Parameter)

Variabel yang diketikkan pada baris perintah.

Parameter posisi : \$0, \$1, \$2 ... \$9  
\$ perintah parameter1 parameter2 ...

|     |     |     |
|-----|-----|-----|
|     |     |     |
| \$0 | \$1 | \$2 |

Contoh :

```
$ cat tetangga
who | grep $1
$ who
unyii    ttyp03    Feb    16    08:37
usro     ttyp09    Feb    16    09:13
$ tetangga unyii
unyii    ttyp03    Feb    16    08:37
```

*variabel \$1 akan disubstitusi oleh "unyii" sehingga program tetangga menjadi who | grep unyii*

### 8.3.2. Parameter Khusus (Special Parameter)

Variabel yang mempunyai arti khusus bagi *shell* dan sudah ditentukan oleh *shell*.

Yang termasuk dalam parameter khusus diantaranya : \$#, \$\*, @\$

#### 1. \$#

Digunakan untuk menyimpan jumlah parameter yang diketikkan pada baris perintah.

Contoh :

```
$ cat > arg
echo Nama program = $0
echo Jumlah argumen = $#
echo Argumen1=:$1: Argumen2=:$2: Argumen3=:$3:
[Ctrl]+[D]
$ arg a b c
Nama program = arg
Jumlah argumen = 3
Argumen1=:a: Argumen2=:b: Argumen3=:c:
$ arg
Nama program = arg
Jumlah argumen = 0
Argumen1=: Argumen2=: Argumen3=:
$ arg "a b c"
Nama program = arg
Jumlah argumen = 1
Argumen1=:a b c: Argumen2=: Argumen3=:
```

#### 2. \$\*

Digunakan untuk menyimpan nilai semua parameter yang diketikkan pada baris perintah.

Contoh :

```
$ cat > arg2
echo Jumlah argumen = $#
echo yaitu :$*:
[Ctrl]+[D]
$ arg2 a b c
Jumlah argumen = 3
yaitu :a b c:
$ arg2
Jumlah argumen = 0
yaitu ::
```

#### 3. @\$

Fungsinya sama dengan `$*` kecuali bila dalam kutip ganda ("`$@"`") akan menempatkan setiap argumen posisi dalam kutip ganda.

Contoh:

```
$ cat > arg3
echo Jumlah argumen = $#
for arg in "$*"
do
    echo $arg
done
[Ctrl]+[D]
$ arg3 a b c
Jumlah argumen = 3
a b c
$ arg3 "a b" c
Jumlah argumen = 2
a
b
c
```

#### 4. Mengganti "`$*`" dengan "`$@"`"

Contoh :

```
$ cat > arg4
echo Jumlah argumen = $#
for arg in "$@"
do
    echo $arg
done
[Ctrl]+[D]
$ arg4 a b c
Jumlah argumen = 3
a
b
c
$ arg4 "a b" c
Jumlah argumen = 2
a b
c
```

### 8.4. Pengutipan

Karakter khusus adalah karakter yang mempunyai fungsi khusus bagi shell.

Contoh karakter khusus :

|                 |   |                      |
|-----------------|---|----------------------|
| <code>?</code>  | : | substitusi nama file |
| <code>*</code>  | : | substitusi nama file |
| <code>`</code>  | : | substitusi perintah  |
| <code>\$</code> | : | substitusi variabel  |

Pengutipan (*quoting*) berfungsi untuk mengabaikan fungsi dari karakter khusus.

Terdapat tiga macam pengutipan :

1. *Backslash* (`\`)

Contoh :

```
$ echo >
bash: syntax error near unexpected token '>'
$ echo \>                > tidak lagi dianggap sebagai karakter pembelokan
```

## 2. Kutip Tunggal ('')

Fungsi : mengabaikan semua karakter khusus di dalamnya, seperti *newline*, *tabs*, spasi

Contoh :

```
$ echo satu      dua      tiga
satu dua tiga
$ echo 'satu      dua      tiga'
satu      dua      tiga
```

## 3. Kutip Ganda ("")

Fungsi : mirip dengan kutip tunggal kecuali tidak mengabaikan :

\$ (substitusi variabel)

*Back quote*/kutip balik (`)

*Backslash* (\)

Contoh :

```
$ x=satu
$ echo $x
satu
$ echo '$x'
$x
$ echo "$x"
satu
```

## 8.5. Kutip Balik

Fungsinya untuk menjalankan perintah yang ada di dalamnya serta menyisipkan *standard output*-nya pada posisi tersebut (*command substitution*).

Bentuk umum : `perintah`

Contoh :

```
$ echo Hari ini : `date`
Hari ini : Tue Feb 16 09:34:40 JAVT 1997
```

## 8.6. Substitusi Nama File

| Karakter | Arti                   |
|----------|------------------------|
| ?        | Match satu karakter    |
| *        | Match semua karakter   |
| [ ... ]  | Match pilihan karakter |

Contoh :

```
$ ls -l
total 10
-rw-r--r--  1 badu      imt           20 Mar 12  1999 Test
-rw-r--r--  1 badu      imt           38 Mar 12  1999 cetak
-rw-r--r--  1 badu      imt           18 Mar 12  1999 test
-rw-r--r--  1 badu      imt           18 Mar 12  1999 test1
-rw-r--r--  1 badu      imt           20 Mar 12  1999 test2
$ ls test?
test1      test2
$ ls *
```

```
Test      cetak      test      test1      test2
$ ls [Tt]est*
Test      test      test1      test2
```

## 8.7. Input dan Output

### 8.7.1. read

Fungsi : untuk membaca data dari keyboard atau file.

Bentuk umum :

```
read var1 var2 ... varn
```

Keterangan :

kata pertama di-assign ke var<sub>1</sub>, kata kedua ke var<sub>2</sub> dan seterusnya.

jika jumlah variabel dalam list lebih sedikit dari jumlah data yang dibaca maka sisa baris disimpan dalam variabel terakhir (var<sub>n</sub>).

Contoh :

1. Membaca data nama dan alamat dari keyboard

```
$ read nama alamat
dudi depok
```

2. Menampilkan isi variabel nama

```
$ echo $nama
dudi
```

3. Menampilkan isi variabel alamat

```
$ echo $alamat
depok
```

### 8.7.2. echo

Fungsi : untuk menampilkan string ke *standard output* (layar).

Bentuk umum :

```
echo [-pilihan] arg1 arg2 ... argn
```

Keterangan :

Setiap argumen dipisahkan oleh satu spasi.

Pilihan yang sering digunakan :

- n : tidak menampilkan baris baru (**newline**)
- e : untuk menampilkan backslash-escaped characters

Backslash-escaped characters :

| Karakter | Hasil yang dicetak                                     |
|----------|--|
| \b       | backspace  |
| \c       | seluruh argumen tanpa new line                         |
| \f       | formfeed   |
| \n       | new line   |
| \r       | carriage return  |
| \\       | karakter <i>backslash</i>                              |
| \nnn     | ASCII karakter dengan nnn=3 digit oktal yang diawali 0 |

Contoh :

1. Menampilkan isi direktori

```
$ echo *
bin mail misc
```

2. Menampilkan baris kosong

```
$ echo
```

3. Menampilkan tabulasi

```
$ echo -e "x\ty"
x      y
```

4. Menampilkan baris baru

```
$ echo -e "\n\nlompati 2 baris"

lompati 2 baris
```

5. Membuat *script* untuk menerima input - tanpa *escape character*

```
$ cat > masukan1
echo "Nama Anda: "
read nama
echo "Nama anda adalah $nama"
[Ctrl]+[D]
$ masukan1
Nama Anda:
Unyil
Nama anda adalah Unyil
```

6. Membuat *script* untuk menerima input - dengan *escape character*

```
$ cat > masukan2
echo -e "Nama Anda: \c"
read nama
echo "Nama anda adalah $nama"
[Ctrl]+[D]
$ masukan2
Nama Anda: Unyil
Nama anda adalah Unyil
```

## 8.8. Pencabangan dengan If

### 8.8.1. if

Fungsi : untuk mengambil keputusan perintah mana yang akan dijalankan dari beberapa pilihan yang ada

Bentuk umum :

|            |            |
|------------|------------|
| if kondisi | if kondisi |
| then       | then       |
| perintah   | perintah   |
| perintah   | perintah   |
| :          | :          |
| fi         | else       |
|            | perintah   |
|            | perintah   |
|            | :          |
|            | fi         |

Keterangan :

Kondisi diperoleh sebagai hasil eksekusi program (berupa *exit status*).

*Exit status* = 0 maka perintah-perintah antara *then* dan *fi* dilaksanakan.

*Exit status*  $\neq$  0

- Perintah antara *then* dan *fi* dilompati.
- Untuk bentuk umum dengan *else* perintah-perintah antara *else* dan *fi* dilaksanakan.

### 8.8.2. Exit Status

Suatu angka yang menunjukkan sukses atau tidaknya suatu program atau perintah dijalankan.

*Exit status* = 0 berarti sukses.

*Exit status*  $\neq$  0 berarti gagal.

*Exit status* dari suatu perintah *pipeline* adalah *exit status* dari perintah paling akhir dalam *pipeline*.

### 8.8.3. Variabel \$?

\$? merupakan variabel penyimpanan *exit status* dari perintah terakhir.

Contoh :

|                                      |                             |
|--------------------------------------|-----------------------------|
| \$ <b>cp file1 file2</b>             |                             |
| \$ <b>echo \$?</b>                   |                             |
| 0                                    | SUKSES                      |
| \$ <b>cp nihil backup</b>            |                             |
| cp: nihil: No such file or directory | asumsi file nihil tidak ada |
| \$ <b>echo \$?</b>                   |                             |
| 1                                    | GAGAL                       |

Contoh :

|                                       |
|---------------------------------------|
| \$ <b>cat &gt; on</b>                 |
| #                                     |
| # menentukan user yang sedang login   |
| #                                     |
| user="\$1"                            |
| if who   grep "\$user"                |
| then                                  |
| echo "\$user sedang login"            |
| else                                  |
| echo "\$user tidak login"             |
| fi                                    |
| [Ctrl]+[D]                            |
| \$ <b>who</b>                         |
| unyil    ttyp03    Feb    16    08:37 |
| usro    ttyp09    Feb    16    09:13  |
| \$ <b>on unyil</b>                    |
| unyil    ttyp03    Feb    16    08:37 |
| unyil sedang login                    |
| \$ <b>on slamet</b>                   |
| slamet tidak login                    |

### 8.8.4. Bentuk else if

Bentuk *else if* digunakan jika pilihan yang harus dilaksanakan lebih dari dua.

Bentuk umum :

|                         |                           |
|-------------------------|---------------------------|
| if kondisi <sub>1</sub> | if kondisi <sub>1</sub>   |
| then                    | then                      |
| perintah                | perintah                  |
| perintah                | perintah                  |
| :                       | :                         |
| else                    | elif kondisi <sub>2</sub> |
| if kondisi <sub>2</sub> | then                      |
| then                    | perintah                  |
| perintah                | perintah                  |
| perintah                | :                         |
| :                       | else                      |
| else                    | perintah                  |
| perintah                | perintah                  |
| perintah                | :                         |

```

fi
fi
fi

```

## 8.9. Evaluasi Kondisi

Fungsi : melakukan *testing* satu kondisi atau lebih

### 8.9.1. test

Bentuk umum : `test ekspresi` atau `[ ekspresi ]` atau `[[ ekspresi ]]`

Keterangan :

ekspresi adalah kondisi yang akan *ditest*

untuk bentuk `[ ekspresi ]` harus ada spasi setelah tanda "[" dan sebelum tanda "]"

test mengembalikan :

- *exit status* = 0 jika ekspresi bernilai *true*.
- *exit status*  $\neq$  0 jika ekspresi bernilai *false*.

### 8.9.2. Operator String

| Operator                        | Menghasilkan <i>exit status</i> =0 jika                       |
|---------------------------------|---|
| <code>string1 = string2</code>  | string1 identik dengan string2                                |
| <code>string1 != string2</code> | string1 tidak identik dengan string2                          |
| <code>string</code>             | string tidak null (test untuk string dengan panjang $\neq$ 0) |
| <code>-n string</code>          | string tidak null (test untuk string dengan panjang $\neq$ 0) |
| <code>-z string</code>          | string null (test untuk string dengan panjang = 0)            |

Contoh :

|                                      |              |
|--------------------------------------|--------------|
| <code>\$ nama="unyii"</code>         |              |
| <code>\$ [ "\$nama" = unyii ]</code> |              |
| <code>\$ echo \$?</code>             |              |
| 0                                    | <i>True</i>  |
| <code>\$ hari="kamis"</code>         |              |
| <code>\$ [ "\$hari" = rabu ]</code>  |              |
| <code>\$ echo \$?</code>             |              |
| 1                                    | <i>False</i> |

### 8.9.3. Operator Integer

| Operator                   | Menghasilkan <i>exit status</i> =0 jika |
|----------------------------|---|
| <code>int1 -eq int2</code> | int1 sama dengan int2                   |
| <code>int1 -ge int2</code> | int1 lebih besar atau sama dengan int2  |
| <code>int1 -gt int2</code> | int1 lebih besar dari int2              |
| <code>int1 -le int2</code> | int1 lebih kecil atau sama dengan int2  |
| <code>int1 -lt int2</code> | int1 lebih kecil dari int2              |
| <code>int1 -ne int2</code> | int1 tidak sama dengan int2             |

Keterangan :

*Shell* tidak akan membedakan tipe harga yang disimpan dalam variabel tetapi *test* akan memperlakukan suatu harga sebagai integer jika dibandingkan dengan operator integer.

Contoh :

|                                  |                             |
|----------------------------------|-----------------------------|
| <code>\$ x1="005"</code>         |                             |
| <code>\$ x2=" 10"</code>         |                             |
| <code>\$ [ "\$x1" = 5 ]</code>   | <i>Perbandingan string</i>  |
| <code>\$ echo \$?</code>         |                             |
| 1                                | <i>False</i>                |
| <code>\$ [ "\$x1" -eq 5 ]</code> | <i>Perbandingan integer</i> |
| <code>\$ echo \$?</code>         |                             |
|                                  | <i>True</i>                 |



```

0
$ [ "$x2" = 10 ]
$ echo $?
1
$ [ "$x2" -eq 10 ]
$ echo $?
0

```

Perbandingan string  
False  
Perbandingan integer  
True

#### 8.9.4. Operator File

| Operator | Menghasilkan exit status=0 jika                |
|----------|--|
| -d file  | file adalah direktori                          |
| -f file  | file adalah file biasa ( <i>ordinary</i> file) |
| -L file  | file adalah symbolic link                      |
| -r file  | file dapat dibaca oleh proses                  |
| -s file  | panjang file > nol                             |
| -w file  | file dapat ditulis oleh proses                 |
| -x file  | file dapat dijalankan atau dieksekusi          |

Contoh :

```

$ ls -l
total 6
drwxr-xr-x  2 badu      imt          32 Mar 12  1999 data
-rw-r--rw-  1 badu      imt          18 Mar 12  1999 hasil
-rw-r--r--  1 badu      imt          20 Mar 12  1999 test
$ [ -f hasil ]
$ echo $?
0
$ [ -d data ]
$ echo $?
0

```

True  
True

#### 8.9.5. Operator Logika

##### 1. Operator negasi (!)

Fungsi : me-negasi-kan hasil evaluasi kondisi/ekspresi

Contoh :

```

$ ls -l
total 6
drwxr-xr-x  2 badu      imt          32 Mar 12  1999 data
-rw-r--rw-  1 badu      imt          18 Mar 12  1999 hasil
-rw-r--r--  1 badu      imt          20 Mar 12  1999 test
$ [ ! -f hasil ]
$ echo $?
1

```

False

[ ! "\$x1" = "\$x2" ]    sama hasilnya dengan  
[ "\$x1" != "\$x2" ]    yaitu akan *true* (*exit status* = 0) jika \$x1 tidak identik dengan \$x2

##### 2. Operator and (-a)

Fungsi :

melakukan fungsi logika *and* terhadap 2 ekspresi  
mengembalikan hasil *true* bila kedua ekspresi benar

Bentuk umum : [ ekspresi<sub>1</sub> -a ekspresi<sub>2</sub> ]

Keterangan :

Memiliki prioritas lebih rendah dibandingkan dengan operator *string*, *integer* dan *file*.

Contoh :

```
$ ls -l
total 6
drwxr-xr-x  2 badu      imt          32 Mar 12  1999 data
-rw-r--rw-  1 badu      imt          18 Mar 12  1999 hasil
-rw-r--r--  1 badu      imt          20 Mar 12  1999 test
$ [ -f hasil -a -d data ]
$ echo $?
0                                True
```

### 8.9.6. Operator or (-o)

Fungsi :

melakukan fungsi logika *or* terhadap 2 ekspresi

mengembalikan hasil *true* bila salah satu ekspresi benar

Bentuk umum : [ ekspresi<sub>1</sub> -o ekspresi<sub>2</sub> ]

Keterangan :

Memiliki prioritas lebih rendah dibandingkan dengan operator -a.

Contoh :

```
$ ls -l
total 6
drwxr-xr-x  2 badu      imt          32 Mar 12  1999 data
-rw-r--rw-  1 badu      imt          18 Mar 12  1999 hasil
-rw-r--r--  1 badu      imt          20 Mar 12  1999 test
$ [ -d hasil -o -d data ]
$ echo $?
0                                True
```

### 8.10. Pengulangan dengan for

Berfungsi untuk menjalankan sekelompok perintah secara berulang-ulang sesuai dengan jumlah yang sudah ditentukan.

Bentuk umum :

- for dengan list

```
for var in word1 word2 ... wordn
do
    perintah
perintah
:
done
```
- for tanpa list

```
for var
do
    perintah
perintah
:
done
```

Keterangan :

Perintah-perintah antara *do* dan *done* disebut badan Iterasi (*Body Loop*)

Pertama kali *loop* dijalankan, *var* disubstitusi dengan *word<sub>1</sub>* (atau \$1 untuk *for* tanpa list)

lalu badan iterasi dijalankan.

Berikutnya `var` disubstitusikan dengan `word2` (atau `$2` untuk `for` tanpa list) kemudian badan iterasi dieksekusi lagi dan seterusnya.

Badan iterasi ini dijalankan berulang-ulang dari atas ke bawah sebanyak:

- Jumlah *word* dalam list setelah `in`
- Jumlah argumen pada *command line* (untuk `for` tanpa list)

Contoh :

1. For dengan list

```
$ cat > loop1
for i in 1 2 3
do
    echo $i
done
[Ctrl]+[D]
$ loop1
1
2
3
```

2. For tanpa list

```
$ cat > loop2
for i
do
    echo $i
done
[Ctrl]+[D]
$ loop2 1 2 3
1
2
3
```

## Latihan Shell Script I

1. Buatlah program dengan nama `isi_nama` yang digunakan untuk memeriksa apakah seseorang memasukkan input nama atau tidak.

Contoh Eksekusi :

```
$ isi_nama
Nama : boim [Enter]
Nama anda adalah boim
$ isi_nama
Nama : [Enter]
Anda tidak memasukkan nama
```

2. Buatlah program dengan nama `cek_tipe` yang digunakan untuk memeriksa tipe suatu file.

Contoh Eksekusi :

```
$ ls -l
total 6
drwxr-xr-x  2 badu      imt           32 Mar 12  1999 data
-rw-r--rw-  1 badu      imt           18 Mar 12  1999 hasil
lrwxrwxrwx  1 badu      imt              5 Mar 12  1999 test ->
hasil
$ cek_tipe test
test is a symbolic link
$ cek_tipe hasil
hasil is a file
$ cek_tipe data
data is a directory
```

3. Buatlah program dengan nama `cek_dir` yang digunakan untuk memeriksa tipe semua file yang ada di *current directory*.

Contoh Eksekusi :

```
$ ls -l
total 6
drwxr-xr-x  2 badu      imt           32 Mar 12  1999 data
-rw-r--rw-  1 badu      imt           18 Mar 12  1999 hasil
lrwxrwxrwx  1 badu      imt              5 Mar 12  1999 test ->
hasil
$ cek_dir
data is a directory
hasil is a file
test is a symbolic link
```

## 9.1. Pencabangan dengan case

### 9.1.1. case

case berfungsi untuk membandingkan satu harga dengan sejumlah pola.

Bentuk umum :

```
case harga in
    pola1 ) perintah
        perintah
        :
        perintah ;;
    pola2 ) perintah
        perintah
        :
        perintah ;;
    polan ) perintah
        perintah
        :
        perintah ;;
esac
```

Keterangan :

case membandingkan harga dengan pola-pola yang ada dari atas ke bawah.

jika ditemukan pola yang tepat maka perintah-perintah untuk pola tersebut dijalankan (dibatasi ;;).

jika tidak ditemukan maka tidak ada perintah-perintah dalam case yang dijalankan.

case dapat memanfaatkan sejumlah *pattern matching character* seperti :

- \*
- ?
- [ ... ]

Contoh :

```
$ cat > ctype
if [ $# -ne 1 ]
then
    echo usage : ctype char
    exit 1
fi
char=$1
case "$char" in
    [0-9] ) echo digit ;;
    [a-z] ) echo huruf kecil ;;
    [A-Z] ) echo huruf besar ;;
    ? ) echo karakter khusus ;;
    * ) echo ketikkan 1 karakter saja ;;
esac
[Ctrl]+[D]
$ ctype u
huruf kecil
$ ctype '>'
karakter khusus
$ ctype xx
```

ketikkan satu karakter saja

### 9.1.2. Operasi Logika or Antar Dua Pola

Bentuk umum:  $pola_1 \mid pola_2$

Keterangan : harga dapat diterima apabila memenuhi salah satu pola.

Contoh :

```
$ cat > greetings
hour=`date +%H`
case "$hour"
in
  0? | 1[01] ) echo "Good morning" ;;
  1[2-7]      ) echo "Good afternoon" ;;
  *           ) echo "Good evening" ;;
esac
[Ctrl]+[D]
$ date
Tue Mar  7 10:12:40 JAVT 2000
$ greetings
Good morning
```

Keterangan :

Perintah `date +%H` hanya akan menampilkan bagian jam-nya saja dari perintah `date`.

### 9.1.3. Bentuk && (and-if)

Fungsi : mengatur eksekusi perintah berdasarkan *exit status* perintah sebelumnya.

Bentuk umum :  $perintah_1 \ \&\& \ perintah_2$

Keterangan :

$perintah_2$  dikerjakan bila  $perintah_1$  menghasilkan *exit status* = 0, jika tidak maka  $perintah_2$  dilompati (melakukan operasi logika *and*)

$perintah_1 \ \&\& \ perintah_2$

hasilnya sama

```
if perintah1
then
    perintah2
fi
```

Contoh :

```
$ cat > cari
user="$1"
who | grep "$user" && echo "$user sedang login"
[Ctrl]+[D]
$ who
unyil  tty01  Feb 15   14:00
badu   tty01  Feb 15   14:01
$ cari unyil
unyil sedang login
$ cari usro
```

### 9.1.4. Bentuk || (or-if)

Bentuk umum :  $perintah_1 \ || \ perintah_2$

Keterangan :

perintah<sub>2</sub> dikerjakan hanya jika perintah<sub>1</sub> menghasilkan *exit status*  $\neq 0$  (melakukan operasi logika *or*)

```
perintah1 || perintah2
```

**hasilnya sama**

```
if perintah1
then
:
else
    perintah2
fi
```

Keterangan :

Tanda : berarti tidak melakukan apa-apa

Contoh :

```
$ cat > cari2
user="$1"
who | grep "$user" || echo "$user tidak login"
[Ctrl]+[D]
$ who
unyil  tty01  Feb 15   14:00
badu   tty02  Feb 15   14:01
$ cari2 unyil
unyil  tty01  Feb 15   14:00
$ cari2 usro
usro tidak login
```

## 9.2. Ekspresi Aritmetika

Operator yang digunakan dalam ekspresi aritmatika :

|   |   |                          |
|---|---|--------------------------|
| + | : | penjumlahan              |
| - | : | pengurangan              |
| / | : | pembagian                |
| * | : | perkalian                |
| % | : | modula (siswa pembagian) |

Keterangan :

"+" dan "-" memiliki prioritas yang lebih rendah dibanding tiga operator lain.

setiap operator didahului dan diikuti oleh spasi.

operator "\*" harus dikutip agar tidak rancu dengan substitusi file.

### 9.2.1. expr

Bentuk umum : `expr ekspresi`

Contoh :

```
$ expr 1 + 2
3
$ expr 17 * 6
expr: syntax error
$ expr 17 "*" 6
102
$ expr 17 '*' 6
102
$ expr 17 \* 6
102
```

Catatan :

Bila hasil dari perintah `expr` akan disimpan ke dalam suatu variabel maka perintah `expr` harus berada di dalam tanda kutip balik.

Contoh :

```
$ i=1
$ i=`expr $i + 1`
$ echo $i
2
```

### 9.2.2. `let`

Bentuk umum : `let ekspresi`

Contoh :

```
$ i=1
$ let i=i+1
$ echo $i
2
```

### 9.2.3. `(( ))`

Bentuk umum : `(( ekspresi ))`

```
$ i=1
$ ((i=i+1))
$ echo $i
2
```

## 9.3. Pengulangan dengan `while` dan `until`

### 9.3.1. `while`

Bentuk umum :

`while kondisi`

`do`

`perintah`

`perintah`

`:`

`done`

Keterangan :

periksa kondisi

jika *exit status* = 0 jalankan perintah-perintah antara `do` dan `done`

ulangi langkah pertama

jika *exit status*  $\neq$  0 **stop**

Contoh :

```
$ cat > selama
i=1
while [ $i -le 4 ]
do
    echo $i
    let i=i+1
done
[Ctrl]+[D]
$ selama
1
2
3
4
```



### 9.3.2. true

Perintah `true` digunakan dalam `while` apabila menginginkan pengulangan secara terus menerus.

Contoh :

```
$ cat > ulang
while true
do
    echo "Hallo"
done
[Ctrl]+[D]
$ ulang
Hallo
Hallo
Hallo
[Ctrl]+[C]
```

*Tekan Ctrl-C untuk menghentikan program*

### 9.3.3. until

Bentuk umum :

```
until kondisi
do
    perintah
    perintah
    :
done
```

Keterangan :

- periksa kondisi
- jika *exit status*  $\neq 0$  jalankan perintah-perintah antara `do` dan `done`
- ulangi langkah pertama
- jika *exit status* = 0 **stop**

Contoh :

```
$ cat > hingga
i=1
until [ $i -gt 4 ]
do
    echo $i
    let i=i+1
done
[Ctrl]+[D]
$ hingga
1
2
3
4
```

### 9.3.4. false

Perintah `false` digunakan dalam `until` apabila menginginkan pengulangan secara terus menerus.

Contoh :

```
$ cat > ulang2
until false
do
    echo "Hallo"
done
```

```
[Ctrl]+[D]
$ ulang2
Hallo
Hallo
Hallo
[Ctrl]+[C]
```

*Tekan Ctrl-C untuk menghentikan program*

### 9.3.5. break

Fungsi : keluar dari pengulangan

Bentuk umum : `break n`

Keterangan :

keluar dari `n loop` terdalam

Contoh :

```
$ cat > potong
i=0
while [ $i -le 4 ]
do
    let i=i+1
    if [ $i = 3 ]
    then
        break
    fi
    echo $i
done
[Ctrl]+[D]
$ potong
1
2
```

### 9.3.6. continue

Fungsi : melompati sisa perintah dalam *loop* untuk kembali melanjutkan *loop* dari atas (kembali ke pemeriksaan kondisi)

Bentuk umum : `continue n`

Keterangan :

`n loop` terdalam dilompati

Contoh :

```
$ cat > lewat
i=0
while [ $i -le 4 ]
do
    let i=i+1
    if [ $i = 3 ]
    then
        continue
    fi
    echo $i
done
[Ctrl]+[D]
$ lewat
1
2
4
```

### 9.3.1. I/O Redirection dalam Pengulangan

Contoh :

```
$ cat > selama2
i=1
while [ $i -le 4 ]
do
    echo $i
    let i=i+1
done > hasil
[Ctrl]+[D]
$ selama2
$ cat hasil
1
2
3
4
```

## 9.4. Fungsi

Merupakan kumpulan perintah yang dapat dipanggil berulang kali dalam suatu *shell script*.

Fungsi dieksekusi seperti perintah biasa.

Bentuk Umum :

```
function nama_fungsi () { perintah; ... perintah; }
```

atau

```
nama_fungsi() { perintah; ... perintah; }
```

Keterangan :

Setelah tanda "{" harus diikuti spasi dan sebelum tanda "}" juga harus didahului spasi

Contoh :

```
$ cat > fungsi
nu () { who | wc -l; }
[Ctrl]+[D]
$ . fungsi
$ who
unyl      tty01   Feb 15   14:00
badu      tty02   Feb 15   14:01
usro      tty03   Feb 15   14:03
ucrit     tty04   Feb 15   14:05
$ nu
4
```

### 9.4.1. return

Digunakan untuk menghentikan eksekusi suatu fungsi dan kembali ke program yang memanggilnya

Bentuk umum : `return n`

Keterangan :

`n` = *exit status* yang ingin dikembalikan

## Latihan Shell Script II

1. Buatlah program dengan nama `nilai` yang digunakan untuk menampilkan jumlah nilai yang dimasukkan dari **keyboard** sebanyak tiga kali.

Contoh Eksekusi :

```
$ nilai
Nilai ke-1 = 8
Nilai ke-2 = 7
Nilai ke-3 = 5
Total Nilai = 20
```

2. Buatlah program dengan nama `tot_nilai` yang digunakan untuk menampilkan jumlah nilai yang dimasukkan dari **keyboard**. Program akan meminta masukan sampai user hanya menekan tombol [Enter].

Contoh Eksekusi :

```
$ tot_nilai
Nilai ke-1 = 8
Nilai ke-2 = 7
Nilai ke-3 = 5
Nilai ke-4 = [Enter]
Total Nilai = 20
```

3. Buatlah program dengan nama `tot_nilai2` yang digunakan untuk menampilkan jumlah nilai yang dimasukkan dari **keyboard**. Program akan meminta masukan sampai user hanya menekan [Enter]. Masukan yang diterima adalah 1 digit numerik, selain itu program akan menampilkan pesan kesalahan dan mengulangi permintaan.

Contoh Eksekusi :

```
$ tot_nilai2
Nilai ke-1 = 8
Nilai ke-2 = n
Masukan angka
Nilai ke-2 = 20
Masukan 1 digit
Nilai ke-2 = 7
Nilai ke-3 = 5
Nilai ke-4 = [Enter]
Total Nilai = 20
```



### 10.1. Berkomunikasi dengan User lain

#### 10.1.1. write

Dengan write user bisa berkomunikasi dengan user lain yang sedang login dengan metode seperti HT (*Handy Talky*). Terminal yang akan dihubungi harus dalam posisi siap menerima pesan.

Gunakan perintah :

- `mesg y` agar terminal anda bisa menerima pesan
- `mesg n` agar terminal anda tidak bisa menerima pesan (tidak ingin diganggu)

Contoh :

```
$ who
dudi      tty1      Sep 25 15:13
agus      tty2      Sep 25 15:14
$ whoami
dudi
$ write agus
Hallo, lagi ngapain?
[Ctrl]+[D]
```

#### 10.1.2. talk

Perintah `talk` adalah salah satu layanan TCP/IP dan dapat dipakai untuk berkomunikasi antar komputer. Metode komunikasinya seperti telepon. Hanya dapat dipakai untuk berkomunikasi antara dua user. Terminal pengirim dan penerima harus dalam posisi siap menerima pesan (`mesg y`). Agar terjadi komunikasi, user yang dihubungi harus memberi respon dengan perintah `talk` juga.

Contoh :

```
$ who

$ whoami
dudi
$ talk dedi
```

#### 10.1.3. mail

mail merupakan metode komunikasi secara off-line (user yang dihubungi tidak perlu dalam posisi login). Mail dapat dikirim kepada satu atau beberapa user sekaligus.

Sintaks: `mail [user-1] [user-2] ... [user-n]`

### 1. Mengirim mail

Contoh :

```
$ mail agus
Subject: rapat rutin
Jangan lupa besok Rabu rapat rutin jam 13:00
Dudi
[Ctrl]+[D]
Cc:
```

### 2. Membaca mail

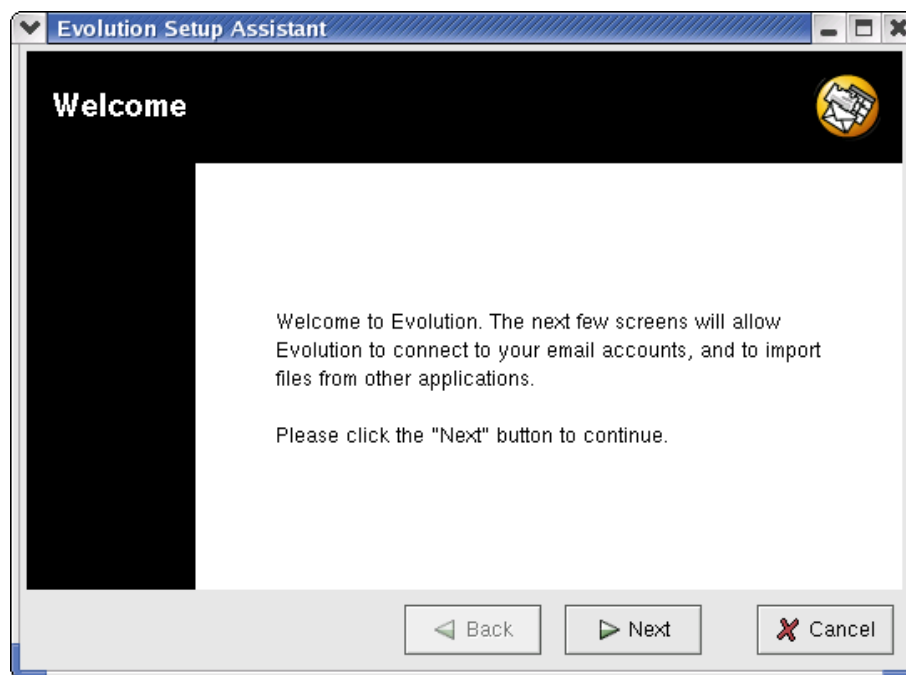
```
$ mail
Mail version 8.1 6/6/93. Type ? for help.
"/var/spool/mail/agus": 1 messages 1 new
>N 1 dudi@jaguar. Thu Sep 25 15:16 16/666 "rapat rutin"
```

### 3. Membalas mail

```
$ mail
Mail version 8.1 6/6/93.  Type ? for help.
"/var/spool/mail/agus": 1 messages 1 new
>N 1 dudi@jaguar.  Thu Sep 25 15:16  16/666  "rapat rutin"
& r
```

#### 10.1.4. Mail dengan Evolution

Evolution merupakan salah satu aplikasi untuk membaca e-mail dari GNOME.



#### 1. Menjalankan Evolution

Untuk menjalankan Evolution ikuti salah satu cara berikut :

- Klik **Mail Menu > Internet > Evolution Email**.
- Klik tombol **Evolution Email** yang terdapat pada **Panel**.

Pada saat pertama kali menjalankan **Evolution**, akan muncul kotak dialog **Evolution Setup Assistant** yang akan membantu Anda untuk membuat account email. Lihat Gambar 10-1.

Berikut ini panduannya :

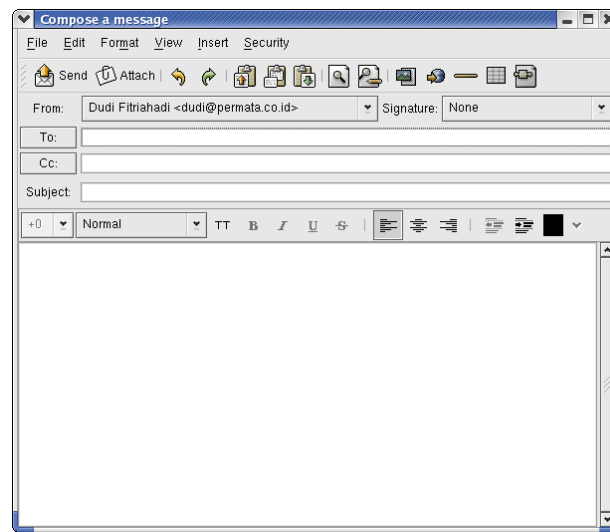
1. Pada bagian **Identity** isikan nama lengkap dan alamat email anda lalu klik tombol **Next**.
2. Pada bagian **Receiving Email** pilih **Local delivery** untuk **Server Type** dan untuk **Path** biarkan sesuai default. Klik tombol **Next**.
3. Bagian ini masih lanjutan dari bagian **Receiving Email**. Pada bagian ini bisa dilewatkan dengan menekan tombol **Next**.
4. Pada bagian **Sending Email** pilih **Sendmail** untuk **Server Type**. Klik tombol **Next**.
5. Pada bagian **Account Management** tuliskan nama untuk account yang baru saja dibuat. Klik tombol **Next**.
6. Pada bagian **Timezone** pilih **Asia/Jakarta**. Klik tombol **Next**.
7. Pada bagian akhir ini Anda tinggal klik tombol **Finish**.

**Gambar 10-1. Evolution Setup Assistant**

## 2. Mengirim Email

Untuk mengirim email, ikuti langkah berikut :

1. Pilih menu **File > New > Mail Message** atau klik tombol **New** pada toolbar. Akan muncul jendela **Compose a message** seperti terlihat pada Gambar 10-2.
2. Pada kotak isian **To**, masukkan alamat email tujuan.
3. Jika ingin memberikan tembusan email kepada seseorang, masukkan alamat email orang yang dimaksud pada kotak isian **Cc**.
4. Masukkan judul email pada kotak isian **Subject**.
5. Tuliskan pesan yang akan disampaikan pada kotak body message.
6. Email bisa dikirim langsung dengan menekan tombol **Send** pada toolbar atau memilih menu **File > Send** atau menekan tombol **[Ctrl]+[Enter]**.



Gambar 10-2. Mengirim Email

## 3. Menerima dan Membaca Email

Untuk menerima dan membaca email, ikuti langkah berikut :

1. Pilih menu **Actions > Send / Receive** atau klik tombol **Send / Receive** pada toolbar atau dengan menekan tombol **[F9]**.
2. Jika ada email baru maka pada shortcut **Inbox** akan muncul angka yang menunjukkan jumlah email yang masuk.
3. Untuk membaca email, klik shortcut **Inbox** dan klik email yang akan dibaca.

## 10.2. Menjelajah Internet dengan Mozilla

Mozilla merupakan browser default pada Linux Red Hat.

### 10.2.1. Menjalankan Mozilla

Untuk menjalankan Mozilla, ikuti salah satu cara berikut :

1. Klik **Main Menu > Internet > Mozilla Web Browser**.
2. Klik tombol **Mozilla Web Browser** yang terdapat pada **Panel**.

Untuk menjelajah internet, masukan alamat URL situs yang akan dikumjungi pada kotak isian yang tersedia, lalu tekan tombol **[Enter]**. Lihat Gambar 10-3.



**Gambar 10-3. Menjelajah Internet dengan Mozilla**