

# Bab 3

# Bash Shell

Program D3 PJJ PENS ITS

# Tujuan

- Setelah mempelajari materi dalam bab ini, mahasiswa diharapkan mampu:
  - Memahami bagaimana menggunakan baris perintah (*command line*).
  - Mengerti penggunaan baris perintah lebih mendalam.
  - Mengetahui bagaimana menggunakan history dan cara pengeditannya.

# Materi

- Shell
- Pengenalan Bash
- Penggunaan Baris Perintah dengan Cepat
- Lebih Lanjut Tentang Baris Perintah

# Shell

- Shell adalah *Command executive*, artinya program yang menunggu instruksi dari pemakai, memeriksa sintak dari instruksi yang diberikan, kemudian mengeksekusi perintah tersebut.
- Untuk pemakai menggunakan prompt \$ dan untuk superuser menggunakan promp #.

# Beberapa Macam Shell

- /bin/sh
  - Bourne shell, dirancang oleh Steve Bourne dari AT&T
- /bin/csh
  - Dikembangkan oleh UNIX Berkeley yang dikenal dengan C-Shell
- /bin/bash
  - Kompatibel dengan Bourne Shell dan juga mengadaptasi kemampuan Korn-Shell.

# Bash Shell

- Kepanjangan dari "*Bourne Again Shell*" yang merupakan kelanjutan dari sh (shell dari original UNIX).
- Dikembangkan untuk proyek GNU dan secara de facto merupakan shell standar Linux.
- Mengimplementasikan beberapa fitur dari shell sebelumnya yaitu sh, csh, ksh, tcsh.

# Penggunaan Baris Perintah

- Wildcard
- Kunci Tab
- History

# Wildcard

- Perlu memanggil/menggunakan perintah yang sama untuk beberapa file pada waktu yang sama.
- Memungkinkan untuk memetakan beberapa nama file dengan proses yang disebut *globbing*.
- Beberapa wildcard antara lain:
  - \* : satu atau lebih karakter yang sesuai
  - ? : single karakter yang sesuai
  - [a-z] : satu jangkauan karakter yang sesuai
  - [^a-z] : semua karakter yang sesuai kecuali jangkauan tersebut
- Contoh, jika satu direktori terdiri dari beberapa file: joshua.txt alex.txt angelo.txt gong.mp3 zonk.mp3.
- Bila diketikkan perintah :
  - \$ rm \*.mp3            sama dengan mengetikkan
  - \$ rm gong.mp3 zonk.mp3



# Kunci Tab

- Digunakan untuk meneruskan perintah baris dengan lengkap.
- Untuk melengkapi nama file, ketik perintah dan dilanjutkan dengan ketik nama file.
- Contoh :

```
$ ls myf<Tab>
```

- Jika hanya ada satu file yang dimulai dengan myf yaitu myfile.txt maka nama file ini akan muncul di layar sebagai berikut

```
$ ls myfile.txt
```

# History

- Catatan dari semua instruksi yang sejauh ini telah dilakukan.
- Dapat dilihat sebagai history, dapat dipilih kembali, diedit dan dieksekusi.
- Navigasi pada daftar history menggunakan karakter kontrol sebagai berikut :
  - ^P (Ctrl-P) melihat instruksi sebelumnya
  - ^N (Ctrl-N) melihat instruksi berikutnya
  - !! eksekusi kembali instruksi sebelumnya
  - !! -3 3 instruksi sebelumnya akan diulang
  - !!88 ulangi instruksi no 88
- Beberapa trik tentang history :
  - Gunakan kunci panah atas dan bawah untuk melihat perintah sebelumnya.
  - Gunakan *Ctrl-r* untuk mencari perintah pada history
  - Untuk memanggil kembali argumen terakhir dari perintah sebelumnya gunakan kunci *Esc* (kunci escape dilanjutkan dengan periode) dan kunci *Alt-* (tekan kunci Alt dilanjutkan periode)

# Baris Perintah (1)

- Fitur *tilde* (~)

```
$ cat ~/.bash_profile
```

Tanda ~ mereferensi ke *home directory*.

- Parameter atau variabel \$ menggantikan nilai variabel dalam baris perintah.

```
$ cd $HOME/public_html
```

Tanda \$HOME mereferensi pada *home directory*.

# Baris Perintah (2)

- Tanda kurung kurawal (`{ }`) membuat string untuk setiap pola di dalam kurung kurawal jika terdapat file.

```
$ echo {a,b}
```

akan menghasilkan string a dan b.

- Perintah output `` `` atau `$( )` disebut perintah pengganti.

```
$ echo "Hostname: `hostname`"
```

# Baris Perintah (3)

- Perintah aritmatika `$ [ ]` menggantikan hasil dari ekspresi matematika ke dalam baris perintah.
- Penggunaan perintah `expr` diikuti dengan *backslash* sebelum tanda asterik untuk perkalian harus diperhatikan.
- Aritmatika dasar yang dikenali terdiri dari :
  - + penambahan
  - pengurangan
  - \* perkalian
  - / pembagian
  - \*\* pangkat
  - % modulo
- Tanda backslash (`\`) menyebabkan karakter berikutnya dikenal  
`$ echo Your cost: \$5.00`

# Baris Perintah (4)

- Tanda backslash juga digunakan untuk melanjutkan perintah pada baris berikutnya.

```
$ echo "This long string will be echoed \  
> back as one long line
```

- Tanda petik satu (‘) membatasi semua ekspansi, tanda petik dua (“) membatasi semua ekspansi kecuali:

```
$ ekspansi variabel
```

- mengganti perintah

```
\ ekspansi satu karakter
```

```
! mengganti history
```