

Pertemuan ke-13

PEMROGRAMAN SHELL SCRIPT (PENYELEKSIAN)

A. TUJUAN

Mampu mengimplementasikan pemrograman *Shell Script* dalam sistem operasi Linux menggunakan konstruksi **if-then-else** dan **case**.

B. TEORI SINGKAT

Shell script adalah program yang eksekusinya tidak memerlukan *compiler/interpreter* khusus, akan tetapi cukup diterjemahkan oleh *shell* tersebut. Dalam sistem Linux, terdapat banyak pilihan untuk *Shell*, salah satunya adalah *Bourne-Again Shell (bash)*. *Bash script* memiliki struktur seperti bahasa C dan beberapa fungsi internal untuk beragam tugas. Selain fungsi internal, *bash* secara otomatis akan menjalankan perintah external apabila perintah tersebut tidak dikenal (sebagai perintah internal). Demikian pula pengguna dapat mendefinisikan sendiri fungsi-fungsi pribadi. *Bash* akan menjalankan perintah yang ditulis di baris perintah (*command line*) atau yang terdapat pada *file text*. Dalam praktikum ini semua perintah akan ditulis pada file, dengan asumsi praktikan mampu menggunakan *text-editor*. Keterangan lengkap mengenai fungsi-fungsi serta sintaks penulisan dapat dilihat pada *Bash User Manual* (man bash).

Status Exit

Setiap program, yang selesai dieksekusi akan memberikan informasi melalui variable spesial `$?`. Adapun indikasi yang diberikan adalah :

- Bila program berakhir dengan sukses, `$? = 0`.
- Bila program berakhir dengan error, `$? <> 0`.

Nilai dari status exit ini, dapat dilihat melalui instruksi `echo $?`

Operator Bilangan Bulat untuk Test

Untuk membandingkan 2 buah bilangan, *test* memerlukan operator yang berbeda dengan string.

Operator	Bernilai True, Jika
-eq	Bilangannya sama
-ge	Lebih besar atau sama dengan
-gt	Lebih besar
-le	Lebih kecil atau sama dengan
-lt	Lebih kecil
-ne	Bilangan tidak sama

Statemen IF

1. **if then**
 if kondisi (instruksi awal)
 then
 instruksi1
 instruksi2

 fi
 if akan mengeksekusi instruksi-awal dan **Status Exit** dari instruksi tersebut akan mempunyai 2 kemungkinan. Bila 0, maka instruksi selanjutnya masuk ke dalam blok **then**, bila `<> 0`, maka alur program diteruskan setelah kata kunci **fi**.
2. **if then else**
 if kondisi
 then
 instruksi1.1
 instruksi1.2

 else
 instruksi2.1
 instruksi2.2

 fi
 Bila **Status Exit** = 0, maka instruksi yang di jalankan adalah instruksi sebelum **else** dan bila `<> 0`, maka kondisi menjadi false dan instruksi yang di jalankan adalah instruksi setelah **else**.
3. **if then elif then else**
 if kondisi1
 then
 instruksi1.1

```

        instruksi1.2
        -----
elif kondisi2
then
    instruksi2.1
    instruksi2.2
    -----
else
    instruksi3.1
    instruksi3.2
    -----
fi

```

Bila **Status Exit** = 0, maka instruksi yang di jalankan adalah instruksi sebelum **else** dan bila $\neq 0$, maka kondisi menjadi false dan instruksi yang di jalankan adalah instruksi setelah **else**.

Statemen Case

Digunakan untuk menyederhanakan pemakaian statemen **If** yang berantai, sehingga dengan **Case**, kondisi dapat dikelompokkan secara logis dengan lebih jelas dan mudah untuk ditulis.

```

case variable in
match1)
    instruksi1.1
    instruksi1.2
    -----
;;
match2)
    instruksi2.1
    instruksi2.2
    -----
;;
*)
    instruksi3.1
    instruksi3.2
    -----
;;
esac

```

Case diakhiri dengan **esac** dan pada setiap kelompok instruksi diakhiri dengan **;;**. Pada akhir pilihan yaitu ***)** yang berarti adalah default, bila kondisi tidak ada yang memenuhi.

C. PRAKTIK

1. Memahami **Status Exit**

a. Jalan perintah-perintah berikut secara berurutan :

```

mkdir program
echo $?

```

Catatlah hasilnya! Pastikan pula bahwa direktori bersangkutan telah tercipta. Bagaimana cara membuktikannya ?

b. Sekarang ulangi kedua perintah diatas. Apakah perintah **mkdir program** gagal melaksanakan tugasnya ? Bagaimana pula hasil perintah **echo \$?** !

c. Bandingkan dengan hasil sebelumnya. Apa kesimpulannya ?

2. Memahami kembali **Status Exit**

a. Pastikan dulu bahwa file bernama **catatan** tak ada pada direktori kerja sekarang. Bagaimana anda memastikannya ?

b. Cobalah berikan perintah **ls catatan**. Perhatikan terlebih dulu apa ada suatu kesalahan ditampilkan oleh perintah tersebut ! Kemudian berikan perintah **echo \$?**.

tesBerapa nilai **Status Exit**-nya ? Apa artinya ?

c. Sekarang berikan perintah :

```

ls
echo $?

```

Mengapa nilai **Status Exit ls** berupa 0 ?

3. Membuat *Shell Script*

a. Buatlah file bernama **tes** dengan isinya sebagai berikut :

```
clear
echo "Tanggal Sekarang"
date
```

- b. Jalankan file **tes** dengan memberikan perintah :

```
./tes
```

Apa yang terjadi ?

Penyebabnya adalah hak *execute* belum diberikan pada file **tes** . Buktikanlah dengan memberikan perintah **ls -l**. Catatlah hak akses untuk user, group, dan other !

- c. Agar **tes** bisa dijalankan, maka hak *execute* perlu di tambahkan. Salah satu caranya adalah dengan memberikan perintah :

```
chmod +x tes
```

Setelah perintah diatas diberikan, catatlah hak akses (user, group, dan other) pada file tersebut.

- d. Jalankan kembali file **tes** dengan memberikan perintah :

```
./tes
```

Apa yang terjadi ?

4. Sekarang marilah untuk mempraktekkan penggunaan statement **if**.

Buatlah file bernama **bonus** dan isinya sebagai berikut :

```
echo "Total Pembelian : "
read total
if test $total -lt 50000
then
echo "Bonus : Tidak Ada"
elif [ $total -lt 100000 ]
then
echo "Bonus : Voucher Rp.5000"
elif [ $total -lt 500000 ]
then
echo "Bonus : Setrika Listrik"
elif [ $total -lt 1000000 ]
then
echo "Bonus : Radio Mini"
else
echo "Bonus : Kipas Angin Kotak"
fi
```

- a. Setelah itu aturlah agar file tersebut bisa dijalankan dengan perintah : **./bonus**

- b. Kemudian ujlilah dan catatlah hasilnya untuk Total pembelian berikut ini :

- 20.000
- 65.000
- 200.000
- 720.000
- 1.300.000

Setelah mencoba perintah-perintah diatas, jelaskan proses dari *script* diatas

5. Mempraktekkan penggunaan statemen **Case**.

Buatlah file bernama **Pilih** dan isinya sebagai berikut :

```
echo "1. Siapa yang aktif di Serverstudent"
echo "2. Tanggal hari ini"
echo "3. Kalender bulan sekarang"
echo -n "Pilihan : "
read pil
case $pil in
1)
echo "User login Anda adalah : "
who
;;
2)
echo "Tanggal hari ini"
date
;;
3)
echo "Kalender bulan sekarang"
cal
```

```
;;
*)
    echo "Salah pilih ...!"
;;
esac
```

c. Aturlah agar file tersebut bisa dijalankan dengan perintah : **./Pilih**

d. Kemudian ujilah dan catatlah hasilnya untuk pilihan berikut ini :

- 1
- 2
- 3
- 4
- 8

c. Setelah mencoba perintah-perintah diatas, jelaskan proses dari *script* diatas.

D. LATIHAN

- Akan disampaikan oleh Dosen Pengampu saat praktikum.

E. TUGAS

- Akan disampaikan oleh Dosen Pengampu saat praktikum.