

Praktikum 3	Pemrograman Input/Output (I/O)
-----------------------	-----------------------------------

A. TUJUAN

1. Menggunakan statement echo dan printf untuk menampilkan data pada standar output (layar monitor)
2. Menggunakan statment read untuk memproses data masukan yang berasal dari standard input (keyboard).
3. Menggunakan konstanta ANSI untuk keluaran dengan pengaturan warna dan cursor

B. DASAR TEORI

Proses I/O

Sebuah proses memerlukan Input dan Output.



Instruksi (command) yang diberikan pada Linux melalui Shell disebut sebagai eksekusi program yang selanjutnya disebut proses. Setiap kali instruksi diberikan, maka Linux kernel akan menciptakan sebuah proses dengan memberikan nomor PID (Process Identity). Proses dalam Linux selalu membutuhkan Input dan menghasilkan suatu Output.

Dalam konteks Linux input/output adalah :

- Keyboard (input)
- Layar (output)
- Files
- Struktur data kernel

Simple I/O

I/O merupakan hal yang mendasar dari kerja komputer karena kapasitas inilah yang membuat komputer begitu berdayaguna. I/O yang dimaksud adalah device yang menangani masukan dan keluaran, baik itu berupa keyboard, floppy, layar monitor, dsb. Penggunaan proses I/O ini pada contoh-contoh sebelumnya seperti statement *echo* yang digunakan untuk menampilkan teks atau informasi ke layar, atau operasi redirect ke file. Selain *echo*, bash menyediakan perintah builtin *printf* untuk mengalihkan keluaran ke output standard, baik ke layar ataupun ke file dengan format tertentu, mirip statement *printf* kepunyaan *bahasa C* atau *perl*. berikut contohnya:

• Output dengan *echo*

Perintah yang digunakan adalah : *echo*.

Kalimat yang mengikuti *echo* akan ditampilkan di layar, kemudian kursor pindah baris berikutnya tanda (opsi) **-n** antara *echo* dengan kalimat digunakan agar kursor tidak pindah ke baris berikutnya, Contoh :

```
#!/bin/bash
#myinfo
#membersihkan tampilan layar
clear
#menampilkan informasi
echo -n "Waktu system    ":"; date
echo -n "Anda           ":"; whoami
echo -n "Banyak pemakai ":"; who | wc -l
```

tentunya layout diatas akan disesuaikan dengan system yang digunakan statement *echo* dengan opsi *-n* akan membuat posisi kursor untuk tidak berpindah ke baris baru karena secara default statement *echo* akan mengakhiri proses pencetakan ke standar output dengan karakter baris baru (newline), dapat juga dicoba tanpa menggunakan opsi *-n*, dan lihat perbedaannya. opsi lain yang dapat digunakan adalah *-e* (enable), memungkinkan penggunaan backslash karakter atau karakter sekuen seperti pada bahasa C atau perl, misalkan :

```
echo -e "\abunyikan bell"
```

jika dijalankan akan mengeluarkan bunyi bell, informasi opsi pada statement *echo* dan backslash karakter selengkapnya dapat dilihat via man di prompt shell. **\$man echo**

- **Output dengan *printf***

```
#!/bin/bash
#pr1
url="pemula.linux.or.id";
angka=32;
printf "Hi, Pake printf ala C\n\t\ta di bash\n";
printf "My url %s\n %d decimal = %o octal\n" $url
    $angka $angka;
printf "%d decimal dalam float = %.2f\n" $angka
    $angka
```

untuk menggunakan format kontrol disertakan simbol %, bash akan mensubtitusikan format tsb dengan isi variabel yang berada di posisi kanan sesuai dengan urutannya jika lebih dari satu variabel, %n %t %a adalah karakter sekuen lepas newline, tab, dan bell, seperti terlihat pada tabel berikut:

Tabel 1. Format Control

Format control	keterangan
%d	untuk format data integer
%o	octal
%f	float atau decimal
%x	Hexadecimal

pada script diatas %.2f akan mencetak 2 angka dibelakang koma, defaultnya 6 angka, informasi lebih lanjut dapat dilihat via man printf

- **Output dengan Konstanta ANSI**

1. Pengaturan Warna

Untuk pewarnaan tampilan dilayar dapat menggunakan konstanta ANSI (salah satu badan nasional amerika yang mengurus standarisasi).

syntaxnya: `\033[warnam`

Dimana:

m menandakan setting color

contohnya:

```
$echo -e "\033[31m HELLO\033[0m"
HELLO
```

konstanta **31m** adalah warna merah, **0m** untuk kembali ke warna normal (none), untuk mengatur tampilan prompt shell, contohnya: `$PS1="\033[34m"`

Daftar warna yang digunakan :

foreground

None	0m		
Black	0;30	Dark Gray	1;30
Red	0;31	Light Red	1;31
Green	0;32	Light Green	1;32
Brown	0;33	Yellow	1;33
Blue	0;34	Light Blue	1;34
Purple	0;35	Light Purple	1;35
Cyan	0;36	Light Cyan	1;36
Light Gray	0;37	White	1;37

background

dimulai dengan 40 untuk BLACK,41 RED,dst

lain-lain

4 underscore,5 blink, 7 inverse

2. Pengaturan posisi kursor

Sedangkan untuk penempatan posisi kursor, dapat digunakan salah satu cara dibawah.

- Menentukan posisi baris dan kolom kursor: `\033[baris;kolomH`
- Pindahkan kursor keatas N baris: `\033[NA`
- Pindahkan kursor kebawah N baris: `\033[NB`
- Pindahkan kursor kedepan N kolom: `\033[NC`
- Pindahkan kursor kebelakang N kolom: `\033[ND`

Contohnya:

```
#!/bin/bash

SETMYCOLOR="\033[42;1;37m"
GOTOYX="\033[6;35H"
clear
echo -e "\033[3;20H INI DIBARIS 3, KOLOM 20"
echo -e "\033[44;1;33;5m\033[5;35H HELLO\033[0m" ;
echo -e "$SETMYCOLOR$GOTOYX ANDA LIHAT INI\033[0m"
```

Menggunakan utility *tput* untuk penempatan posisi kursor

Penempatan posisi kursor di layar dapat diatur dengan memanfaatkan utility *tput*.

syntaxnya:

tput cup baris kolom

contohnya:

```
#!/bin/bash

clear
tput cup 5 10
echo "HELLO"
tput cup 6 10
echo "PAKE TPUT"
```

jika dijalankan akan didapatkan string *HELLO* di koordinat baris 5 kolom 10, dan string *PAKE TPUT* dibaris 6 kolom 10. informasi selengkapnya tentang *tput* gunakan man *tput*, atau info *tput*.

- **Perintah Input**

Perintah yang digunakan adalah : ***read***

Membaca data yang dimasukkan dari input standar (keyboard)

syntaknya :

read -opsi [nama_variabel...]

opsi

-t(TIMEOUT), ***-p*** (PROMPT), ***-s***(SILENT), ***-n*** (NCHAR) dan ***-d***(DELIM)

Tabel 1. Contoh Opsi untuk Read

Opsi	Keterangan
-p	memungkinkan kita membuat prompt sebagai informasi pengisian
-s	membuat input yang dimasukkan tidak di echo ke layar (seperti layaknya password di linux)
-n	menentukan banyak karakter yang diinput
-d	menentukan karakter pembatas masukan

C. PROSEDUR

1. Masuk ke sistem operasi Linux.
2. Login sebagai stD3XXYYY.
3. Gunakan perintah-perintah input dan output pada shell : echo , printf, tput, dan read.
4. Catatlah hasil dari setiap perintah pada prosedur no.3 sebagai laporan sementara.
5. Kerjakan soal latihan yang telah disediakan pada modul praktikum.

D. PERCOBAAN

Percobaan 1 : Output

1. Menampilkan kata *Hello World* dengan perintah *echo*

```
$vi out.sh
#!/bin/sh
# This is a comment!
```

```

echo "Hello      World"
echo "Hello World"
echo "Hello * World"
echo Hello * World
echo Hello      World
echo "Hello" World
echo Hello "      " World
echo "Hello \"*\ " World"
echo `hello` world
echo 'hello' world

```

2. Menampilkan kalimat pada baris yang berbeda

```

$vi out1.sh

#!/bin/bash
#Ini adalah program pertama
echo "Selamat belajar program bash"
echo "Mudah dan menyenangkan.....!"

$chmod u+x out1.sh
$./out1.sh

```

3. Menampilkan kalimat pada baris yang sama

```

$vi out2.sh

#!/bin/bash
#Ini adalah program kedua
echo -n "selamat"
echo "belajar program bash"

```

4. Menampilkan informasi waktu system, tentang user dan jumlah pemakai yang sedang login di system

```

$vi out3.sh

#!/bin/bash
#myinfo
#membersihkan tampilan layar
clear
#menampilkan informasi
echo -n "Waktu system    :"; date
echo -n "Anda           :"; whoami
echo -n "Banyak pemakai :"; who | wc -l

```

5. Menampilkan kalimat, tipe shell dengan echo langsung dari prompt

```

$echo Hello World
$echo '#!/bin/bash' > my-script.sh
$echo 'echo Hello World' >> my-script.sh
$chmod 755 my-script.sh
$./my-script.sh
$echo $SHELL

```

6. Menampilkan beberapa informasi dengan perintah *printf*

```
$vi out4.sh

#!/bin/bash
#pr1
url="pemula.linux.or.id";
angka=32;
printf "Hi, Pake printf ala C\n\t\ a di bash\n";
printf "My url %s\n %d decimal = %o octal\n" $url
    $angka $angka;
printf "%d decimal dalam float = %.2f\n" $angka
    $angka
```

7. Output dengan konstanta ANSI untuk pengaturan warna

```
$vi out5.sh

$echo -e "\033[31m HELLO\033[0m"

$PS1="\033[34m"
$echo -e "\033[31;1;33m Bash and ansi color\033[0m"
```

8. Output dengan konstanta ANSI untuk pengaturan posisi kursor

```
$vi out6.sh

#!/bin/bash
SETMYCOLOR="\033[42;1;37m"
GOTOYX="\033[6;35H"
clear
echo -e "\033[3;20H INI DIBARIS 3, KOLOM 20"
echo -e "\033[44;1;33;5m\033[5;35H HELLO\033[0m";
echo -e "$SETMYCOLOR$GOTOYX ANDA LIHAT INI\033[0m"
```

9. Pengaturan posisi kursor dengan perintah *tput*

```
$vi out7.sh

#!/bin/bash
clear
tput cup 5 10
echo "HELLO"
tput cup 6 10
echo "PAKE TPUT"
```

Percobaan 2 : Input

1. Menampilkan nama yang berasal dari keyboard

```
$vi inp1.sh

#!/bin/bash
echo -n "Tuliskan nama : "
read nama
echo "Hallo .....$nama"

$chmod 777 inp1.sh
$./inp1.sh
```


2. Proses pembacaan data input dengan read menggunakan opsi

```
$vi inp2.sh
#!/bin/bash
read -p "User Name : " user
echo -e "Password 10 karakter,\njika dalam 6 second
        tidak dimasukkan pengisian password diakhiri"
read -s -n 10 -t 6 pass
echo    "kesan anda selama pake linux,
        _underscore=>selesai"
read -d _ kesan
echo    "User = $user"
echo    "Password = $pass"
echo    "Kesan selama pake linux = $kesan"
```

3. Proses pembacaan data input dengan read dan variable *reply*

```
$vi inp3.sh
#!/bin/bash
#rd1
echo -n "Nama anda : "
read nama;
echo    "Hi $nama, apa khabarmu";
echo    "Pesan dan kesan : ";
read
echo    "kata $nama, $REPLY";
```

E. Latihan

1. Buatlah program bash script untuk menampilkan tulisan sebagai berikut :

```
what is your name ?
Andi                      ;(diketik lewat keyboard)
Hello Andi
I will create you a file called Andi_file
```

- gunakan perintah read untuk membaca input dari keyboard dan perintah echo, printf untuk mencetak. File dengan nama *Andi_file* tersimpan secara otomatis pada direktori aktif ketika eksekusi program telah dilaksanakan.

- periksalah apakah file *Andi_file* sudah tersimpan pada direktori aktif

2. Buatlah program bash script untuk mencetak tulisan sebagai berikut :

```
A man, a plan, a canal, Panama.
```

3. Buatlah program bash script untuk menampilkan tulisan sebagai berikut :

```
I will guess your user name.
It is: ericfj
```

Keterangan : ericj adalah login name user yang tercetak secara otomatis ketika program dijalankan.

4. Buatlah program bash script untuk menampilkan tulisan sebagai berikut :

Hmm...determining operating system...

This computer runs: Linux

Keterangan : linux adalah identitas mesin (informasi sistem kernel) yang tercetak secara otomatis ketika program dijalankan.