

LAPORAN RESMI
PRAKTIKUM 7
“ Pemrograman Shell 1 ”



NAMA : FERI AFRIANTO
KELAS : D4 Teknik Informatika A
NRP : 2110191007

Politeknik Elektronika Negeri Surabaya

PRAKTIKUM 7

“ Pemrograman Shell 1 ”

1. PERCOBAAN

1. Percobaan 1.

- 1) Buatlah file prog01.sh dengan editor vi

```
$ vi prog01.sh
#!/bin/sh
# Program shell
#
```

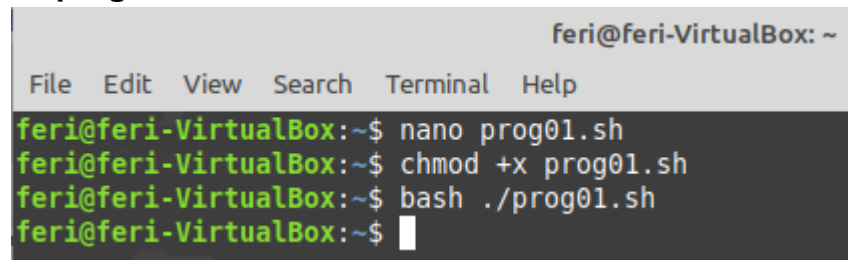
```
var1=x
var2=8
```



GNU nano 2.9.3 prog01.sh

```
#!/bin/sh
# Program Shell
#
var1=x
var2=8
```

- 2) Untuk menjalankan shell, gunakan notasi TITIK di depan nama program
\$. prog01.sh
- 3) Untuk menjalankan shell, dapat juga dengan membuat executable file dan dieksekusi relatif dari current directory
\$ chmod +x prog01.sh
\$./prog01.sh



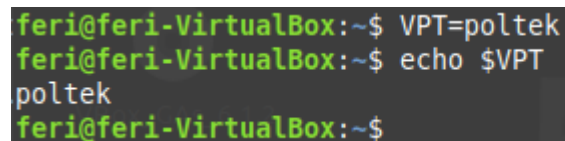
feri@feri-VirtualBox: ~

```
File Edit View Search Terminal Help
feri@feri-VirtualBox:~$ nano prog01.sh
feri@feri-VirtualBox:~$ chmod +x prog01.sh
feri@feri-VirtualBox:~$ bash ./prog01.sh
feri@feri-VirtualBox:~$
```

2. Percobaan 2.

- 1) Contoh menggunakan variable pada shell interaktif

```
$ VPT=poltek
$ echo $VPT
```



```
feri@feri-VirtualBox:~$ VPT=poltek
feri@feri-VirtualBox:~$ echo $VPT
poltek
feri@feri-VirtualBox:~$
```

- 2) Pemisahan 2 kata dengan spasi menandakan eksekusi 2 buah instruksi. Karakter \$ harus ada pada awal nama variable untuk melihat isi variable tersebut, jika tidak, maka echo akan mengambil parameter tersebut sebagai string.

```
$ VPT2=poltek elektronika (Terdapat pesan error)
```

```
$ VPT2="poltek elektronika"
```

```
$ echo VPT2
```

```
$ echo $VPT2
```

```
feri@feri-VirtualBox: ~  
File Edit View Search Terminal Help  
feri@feri-VirtualBox:~$ VPT2=poltek elektronika  
elektronika: command not found  
feri@feri-VirtualBox:~$ VPT2="poltek elektronika"  
feri@feri-VirtualBox:~$ echo VPT2  
VPT2  
feri@feri-VirtualBox:~$ echo $VPT2  
poltek elektronika  
feri@feri-VirtualBox:~$
```

- 3) Menggabungkan dua variable atau lebih

```
$ V1=poltek
```

```
$ V2=':'
```

```
$ V3=elektronika
```

```
$ V4=$V1$V2$V3
```

```
$ echo $V4
```

```
feri@feri-VirtualBox:~$ V1=poltek  
feri@feri-VirtualBox:~$ V2=':'  
feri@feri-VirtualBox:~$ V3=elektronika  
feri@feri-VirtualBox:~$ V4=$V1$V2$V3  
feri@feri-VirtualBox:~$ echo $V4  
poltek:elektronika  
feri@feri-VirtualBox:~$
```

- 4) Menggabungkan isi variable dengan string yang lain. Jika digabungkan dengan nama variable yang belum didefinisikan (kosong) maka instruksi echo menghasilkan string kosong. Untuk menghindari kekeliruan, nama variable perlu diproteksi dengan { } dan kemudian isi variable tersebut digabung dengan string.

```
$ echo $V3
```

```
$ echo $V3ITS
```

```
$ echo ${V3}ITS
```

```
feri@feri-VirtualBox:~$ echo $V3  
elektronika  
feri@feri-VirtualBox:~$ echo $V3ITS  
  
feri@feri-VirtualBox:~$ echo ${V3}ITS  
elektronikaITS  
feri@feri-VirtualBox:~$
```

- 5) Variabel dapat berisi instruksi, yang kemudian bila dijadikan input untuk shell, instruksi tersebut akan dieksekusi

```
$ CMD=who
```

```
$ $CMD
```

```
$ CMD="ls -l"
```

```
$ $CMD
```

```
feri@feri-VirtualBox:~$ CMD=who
feri@feri-VirtualBox:~$ $CMD
feri      tty7          2020-03-14 12:05 (:0)
feri@feri-VirtualBox:~$ CMD="ls -l"
feri@feri-VirtualBox:~$ $CMD
total 18288
-rw-r--r-- 1 feri feri      25 Mar  1 23:07 baru
-rw-r--r-- 1 feri feri    2394 Mar  1 22:57 baru.urut
-rwxr-xr-x 1 feri feri     43 Mar 10 11:50 coba.ph
drwxr-xr-x 2 feri feri   4096 Feb 21 10:43 Desktop
-rw-r--r-- 1 feri feri    257 Mar  8 21:58 directories.txt
drwxr-xr-x 2 feri feri   4096 Feb 21 10:43 Documents
```

- 6) Modifikasi file prog01.sh berikut

```
$ vi prog01.sh
```

```
#!/bin/sh
```

```
V1=poltek
```

```
V2=':'
```

```
V3=elektronika
```

```
echo "Pemrograman shell"
```

```
echo $V1$V2$V3
```

```
V3=ITS
```

```
echo $V1$V2 di $V3
```

```
GNU nano 2.9.3 prog01.sh

#!/bin/sh
# Program Shell
#
V1=poltek
V2=':'
V3=elektronika
echo "Pemrograman shell"
echo $V1$V2$V3
V3=ITS
echo $V1$V2 di $V3
```

- 7) Cara sederhana mengeksekusi shell adalah dengan menggunakan notasi titik di depan nama shell script tersebut. Bila direktori actual tidak terdaftar dalam PATH, maka command tersebut tidak dapat ditemukan. Bila script belum executable, script tidak dapat dieksekusi.

```
$ . prog01.sh
```

```
$ prog01.sh (Terdapat pesan error)
```

```
$ ./prog01.sh (Terdapat pesan error)
```

```
$ chmod +x prog01.sh
```

```
$ ./prog01.sh$ cat /bin/bash | string | grep shell | less$ ^shell^alias^
```

```
feri@feri-VirtualBox:~$ . prog01.sh
Pemrograman shell
poltek:elektronika
poltek: di ITS
feri@feri-VirtualBox:~$ prog01.sh
prog01.sh: command not found
feri@feri-VirtualBox:~$ ./prog01.sh
Pemrograman shell
poltek:elektronika
poltek: di ITS
feri@feri-VirtualBox:~$ chmod +x prog01.sh
feri@feri-VirtualBox:~$ ./prog01.sh
Pemrograman shell
poltek:elektronika
poltek: di ITS
feri@feri-VirtualBox:~$
```

3. Percobaan 3.

1) Menggunakan instruksi read

```
$ read nama
```

```
amir
```

```
$ echo $nama
```

```
feri@feri-VirtualBox:~$ read nama
feri
feri@feri-VirtualBox:~$ echo $nama
feri
feri@feri-VirtualBox:~$
```

2) Membaca nama dan alamat dari keyboard

```
$ vi prog02.sh
```

```
#!/bin/sh
```

```
# prog02.sh
```

```
# membaca nama dan alamat
```

```
echo "Nama Anda : "
```

```
read nama
```

```
echo "Alamat : "
```

```
read alamat
```

```
echo "Kota : "
```

```
read kota
```

```
echo
```

```
echo "Hasil adalah : $nama, $alamat di $kota"
```

```
GNU nano 2.9.3 prog02.sh

#!/bin/sh
# prog01.sh
# membaca nama dan alamat
echo "Nama Anda : "
read nama
echo "Alamat : "
read alamat
echo "Kota : "
read kota
echo
echo "Hasil adalah : $nama,$alamat di $kota"
```

3) Eksekusi program prog02.sh

```
$ . prog02.sh
```

Nama Anda :

Amir

Alamat :

Jl semangka 67

Kota :

Surabaya

Hasil adalah : Amir, Jl semangka di Surabaya

```
feri@feri-VirtualBox:~$ . prog02.sh
Nama Anda :
Feri
Alamat :
Kauman
Kota :
Tulungagung

Hasil adalah : Feri,Kauman di Tulungagung
feri@feri-VirtualBox:~$
```

4) Instruksi echo secara otomatis memberikan baris baru, maka untuk menghindari hal tersebut disediakan opsi -n, yang menyatakan kepada echo untuk menghilangkan baris baru. Modifikasi program prog02.sh

```
GNU nano 2.9.3 prog02.sh

#!/bin/sh
# prog01.sh
# membaca nama dan alamat
echo -n "Nama Anda : "
read nama
echo -n "Alamat : "
read alamat
echo -n "Kota : "
read kota
echo
echo "Hasil adalah : $nama,$alamat di $kota"
```

5) Eksekusi program prog02.sh

```
feri@feri-VirtualBox:~$ . prog02.sh
Nama Anda : Feri
Alamat : Kauman
Kota : Tulungagung

Hasil adalah : Feri,Kauman di Tulungagung
feri@feri-VirtualBox:~$
```

- 6) Variabel kosong adalah variable yang tidak mempunyai nilai. Variabel ini didapat atas assignment atau membaca dari keyboard atau variable yang belum didefinisikan.

```
feri@feri-VirtualBox:~$ read nama
<CR>
feri@feri-VirtualBox:~$ echo $nama
<CR>
feri@feri-VirtualBox:~$ A=
feri@feri-VirtualBox:~$ B=""
feri@feri-VirtualBox:~$ C=$A$B
feri@feri-VirtualBox:~$ echo $c

feri@feri-VirtualBox:~$ echo $C

feri@feri-VirtualBox:~$
```

- 7) Variabel dapat disubstitusikan dengan hasil eksekusi dari sebuah instruksi. Pada contoh dibawah , instruksi pwd dieksekusi lebih dahulu dengan sepasang Back Quate (tanda kutip terbalik). Hasil dari eksekusi tersebut akan masuk sebagai nilai variable DIR.

```
feri@feri-VirtualBox:~$ pwd
/home/feri
feri@feri-VirtualBox:~$ DIR=`pwd`
feri@feri-VirtualBox:~$ echo $DIR
/home/feri
feri@feri-VirtualBox:~$
```

- 8) Buatlah shell script prog03.sh

```
GNU nano 2.9.3 prog03.sh

#!/bin/sh
# prog03.sh
#
NAMA=`whoami`
echo Nama Pengguna Aktif adalah $NAMA
tanggal=`date | cut -c1-10`
echo Hari ini tanggal $tanggal
```

- 9) Eksekusi prog03.sh

```
feri@feri-VirtualBox:~$ . prog03.sh
Nama Pengguna Aktif adalah feri
Hari ini tanggal Sat Mar 14
feri@feri-VirtualBox:~$
```


4. Percobaan 4

1) Membuat shell script prog04.sh

```
GNU nano 2.9.3 prog04.sh
#!/bin/sh
# prog04.sh versi 1
# Parameter Passing
#
echo "Nama Program adalah $0"
echo "Parameter 1 adalah #1"
echo "Parameter 2 adalah $2"
echo "Parameter 3 adalah $3"
```

2) Eksekusi prog04.sh tanpa parameter, dengan 2 parameter, dengan 4 parameter

```
feri@feri-VirtualBox:~$ . prog04.sh
Nama Program adalah bash
Parameter 1 adalah
Parameter 2 adalah
Parameter 3 adalah
feri@feri-VirtualBox:~$ . prog04.sh amir hasan
Nama Program adalah bash
Parameter 1 adalah amir
Parameter 2 adalah hasan
Parameter 3 adalah
feri@feri-VirtualBox:~$ . prog04.sh amir hasan badu ali
Nama Program adalah bash
Parameter 1 adalah amir
Parameter 2 adalah hasan
Parameter 3 adalah badu
feri@feri-VirtualBox:~$
```

3) Membuat shell script prog04.sh versi 2 dengan memberikan jumlah parameter

```
GNU nano 2.9.3 prog04.sh
#!/bin/sh
# prog04.sh versi 2
# Parameter Passing
#
echo "Jumlah parameter yang diberikan adalah $#"
```

4) Eksekusi prog04.sh tanpa parameter dan dengan 4 parameter

```
feri@feri-VirtualBox:~$ . prog04.sh
Jumlah parameter yang diberikan adalah 0
Nama Program adalah bash
Parameter 1 adalah
Parameter 2 adalah
Parameter 3 adalah
feri@feri-VirtualBox:~$ . prog04.sh amir hasan badu ali
Jumlah parameter yang diberikan adalah 4
Nama Program adalah bash
Parameter 1 adalah amir
Parameter 2 adalah hasan
Parameter 3 adalah badu
feri@feri-VirtualBox:~$
```

5) Membuat shell script prog04.sh versi 3 dengan menambahkan total parameter dan nomor proses id (PID)

```
GNU nano 2.9.3 prog04.sh

#!/bin/sh
# prog04.sh versi 2
# Parameter Passing
#
echo "Jumlah parameter yang diberikan adalah $#"
```

```
echo "Nama Program adalah $0"
```

```
echo "Parameter 1 adalah $1"
```

```
echo "Parameter 2 adalah $2"
```

```
echo "Parameter 3 adalah $3"
```

```
echo "Total parameter adalah $*"
```

```
echo "PID proses shell ini adalah $$"
```

6) Eksekusi prog04.sh dengan 4 parameter

```
feri@feri-VirtualBox:~$ . prog04.sh amir hasan badu ali
Jumlah parameter yang diberikan adalah 4
Nama Program adalah bash
Parameter 1 adalah amir
Parameter 2 adalah hasan
Parameter 3 adalah badu
Total parameter adalah amir hasan badu ali
PID proses shell ini adalah 2189
feri@feri-VirtualBox:~$
```

5. Percobaan 5

1) String tidak ditemukan, maka status exit adalah 1

```
feri@feri-VirtualBox:~$ grep xyz /etc/passwd
feri@feri-VirtualBox:~$ echo $?
1
feri@feri-VirtualBox:~$
```

2) String ditemukan, maka status exit adalah 0

```
feri@feri-VirtualBox:~$ grep feri /etc/passwd
feri:x:1000:1000:FERI AFRIANTO,1,085676544563,03557865:/home/feri:/bin/bash
feri@feri-VirtualBox:~$ echo $?
0
```

6. Percobaan 6

1) Instruksi dengan exit status 0

```
feri@feri-VirtualBox:~$ who
feri      tty7      2020-03-14 12:05 (:0)
feri@feri-VirtualBox:~$ who | grep feri
feri      tty7      2020-03-14 12:05 (:0)
feri@feri-VirtualBox:~$ echo $?
0
```

2) If membandingkan exit status dengan 0, bila sama, maka blok program masuk ke dalam blok then-if.

```
feri@feri-VirtualBox:~$ if [ $? = 0 ]
> then
> echo "Pemakai tersebut sedang aktif"
> fi
Pemakai tersebut sedang aktif
feri@feri-VirtualBox:~$
```

3) Nomor (1) dan (2) diatas dapat disederhanakan dengan

```
feri@feri-VirtualBox:~$ if who | grep feri > /dev/null
> then
> echo Okay
> fi
Okay
feri@feri-VirtualBox:~$
```

7. Percobaan 7

1) Membuat shell script prog05.sh

```
#!/bin/sh
# prog05.sh
# Program akan memberikan konfirmasi apakah nama
# user sedang aktif atau tidak
#
echo -n "Berikan nama pemakai : "
read nama
if who | grep $nama > /dev/null
then
    echo "$nama sedang aktif"
else
    echo "$nama tidak aktif"
fi
```

2) Jalankan prog05.sh, masukkan nama pemakai yang aktif yang tampil pada instruksi who dan coba juga untuk nama pemakai yang tidak aktif

```
feri@feri-VirtualBox:~$ . prog05.sh
Berikan nama pemakai : Zaid Abillah
grep: Abillah: No such file or directory
Zaid Abillah tidak aktif
feri@feri-VirtualBox:~$ . prog05.sh
Berikan nama pemakai : feri
feri sedang aktif
```

2. KESIMPULAN

Dalam praktikum ini saya belajar bagaimana cara mendeklarasikan sebuah variabel, bagaimana pemrograman string, Input output dan parameter yang ada dalam linux.