

Nama : Fadlila Agustina
NPM : 21083010050
Kelas : Sistem Operasi (B)

Laporan Tipe Data di Pemrograman Shell

1. String

Di dalam string kita dapat mengurutkan karakter, mencari karakter, memilih karakter yang mau ditampilkan dan lain sebagainya.

a. Non-nano version (terminal)

```
fadlila@fadlila-VirtualBox:~$ a='ini single-quoted\n'  
fadlila@fadlila-VirtualBox:~$ b="ini double-quoted\n"  
fadlila@fadlila-VirtualBox:~$ echo -e ,a=$a,b=$b  
,a=ini single-quoted  
,b=ini double-quoted
```

- Di atas merupakan cara untuk menampilkan output melalui terminal.
- Script pada terminal tidak menggunakan \n berbeda dengan di nano.

b. Nano version (menggunakan bash)

```
fadlila@fadlila-VirtualBox:~$ nano string.sh  
fadlila@fadlila-VirtualBox:~$ bash string.sh  
,a=ini single-quoted  
,b=ini double-quoted
```

- Membuat file bernama string.sh menggunakan perintah nano.
- Untuk menampilkan output dari skrip string.sh yang dibuat di dalam nano dapat menggunakan perintah bash.

```
a='ini single-quoted\n'  
b="ini double-quoted\n"  
echo -e ,a=$a,b=$b
```

- Di atas adalah skrip string.sh yang dibuat menggunakan perintah nano.
- \n berfungsi untuk enter bagi kalimat selanjutnya.
- echo adalah perintah yang berfungsi untuk mencetak output.
- Tekan ctrl+s untuk save dan ctrl+x untuk kembali ke terminal.

2. Array

Array adalah kumpulan variable dengan tipe sejenis.

```
fadlila@fadlila-VirtualBox:~$ nano array.sh  
fadlila@fadlila-VirtualBox:~$ bash array.sh  
Saya Memilih Distro 0, Mint !  
fadlila@fadlila-VirtualBox:~$ bash array.sh  
Saya Memilih Distro 1, Ubuntu !  
fadlila@fadlila-VirtualBox:~$ bash array.sh  
Saya Memilih Distro 3, Arch !  
fadlila@fadlila-VirtualBox:~$ bash array.sh  
Saya Memilih Distro 4, Debian !  
fadlila@fadlila-VirtualBox:~$ bash array.sh  
Saya Memilih Distro 4, Debian !  
fadlila@fadlila-VirtualBox:~$ bash array.sh  
Saya Memilih Distro 4, Debian !  
fadlila@fadlila-VirtualBox:~$ bash array.sh  
Saya Memilih Distro 0, Mint !
```

- Membuat file array.sh menggunakan perintah nano.
- Untuk menampilkan output dari skrip array.sh yang dibuat di dalam nano dapat menggunakan perintah bash.
- Output yang ditampilkan berhasil atau tidak error, karena hasil yang keluar berbeda atau acak.

```
#!/bin/bash

# deklarasi array
distroLinux=("Mint" "Ubuntu" "Kali" "Arch" "Debian")

# random distro
let pilih=$((RANDOM%5))

#eksekusi
echo "Saya Memilih Distro $pilih, ${distroLinux[$pilih]} !"
```

- Di atas adalah skrip array.sh yang dibuat menggunakan perintah nano.
- #!/bin/bash merupakan perintah yang diterjemahkan ke kernel linux untuk mengeksekusi path yang disertakan pada direktori /bin. Walaupun tidak menggunakan perintah tersebut, script di atas tetap dapat dijalankan.
- distroLinux untuk mendeklarasikan 5 elemen, yaitu Mint, Ubuntu, Kali, Arch, dan Debian.
- Baris selanjutnya digunakan agar output yang ditampilkan dapat terpilih array secara *random*.
- Lalu echo perintah yang digunakan untuk mencetak script di atas.

3. Integer

Integer adalah salah satu jenis tipe data primitif yang secara standar sudah terdefinisi pada suatu bahasa pemrograman. Data integer adalah tipe data yang digunakan secara khusus untuk menampung bilangan-bilangan bulat positif dan negative, serta tidak mengandung bilangan pecahan desimal.

```
fadlila@fadlila-VirtualBox:~$ nano integer.sh
fadlila@fadlila-VirtualBox:~$ bash integer.sh
579
```

- Membuat file bernama integer.sh dengan menggunakan perintah nano.
- Untuk menampilkan output dari skrip integer.sh yang dibuat di dalam nano dapat menggunakan perintah bash.

```
a=123
b=456
let c=a+b
echo $c
```

- Di atas merupakan skrip integer.sh yang dibuat menggunakan perintah nano.
- a mendeklarasikan angka 123, b mendeklarasikan angka 456, dan c mendeklarasikan penjumlahan dari a dan b.
- karakter \$ digunakan untuk menampilkan nilai dari variable yang dipilih.

4. Operasi Aritmatika

Operasi aritmatika adalah operasi dasar matematika yang terdiri dari penjumlahan (+), pengurangan (-), perkalian (*), pembagian (/), modulus atau melihat sisa dari hasil bagi

(%), menempatkan nilai di sisi kanan ke variable di sisi kiri (=), membandingkan 2 nilai yang sama (==), dan membandingkan 2 nilai yang tidak sama (!=)

```
fadlila@fadlila-VirtualBox:~$ nano op_mtk.sh
fadlila@fadlila-VirtualBox:~$ bash op_mtk.sh
a + b = 22
a - b = 8
a * b = 105
a / b = 2
a % b = 1
a = 15
b = 15
```

- Membuat file bernama op_mtk.sh menggunakan perintah nano.
- Untuk menampilkan output dari skrip op_mtk.sh yang dibuat di dalam nano dapat menggunakan perintah bash.

```
#!/bin/bash

a=15
b=7

#memakai let
let jumlah=$a+$b
let kurang=$a-$b
let kali=$a*$b

#memakai expr
bagi=`expr $a / $b`

#memakai perintah substitusi $((ekspresi))
mod=$(( $a % $b ))

echo "a + b = $jumlah"
echo "a - b = $kurang"
echo "a * b = $kali"
echo "a / b = $bagi"
echo "a % b = $mod"

b=$a

echo "a = $a"
echo "b = $b"
```

- Di atas merupakan skrip op_mtk.sh yang dibuat menggunakan nano.
- a mendeklarasikan angka 15 dan b mendeklarasikan angka 7.
- Pada operasi penjumlahan, pengurangan, dan perkalian menggunakan *let* (*build in*), pembagian menggunakan *expr* atau *awk* (perintah eksternal), dan modulus menggunakan syntax perintah substitusi \$((ekspresi)).
- b=\$a berarti angka yang dideklarasikan b yang sebelumnya 7 berubah menjadi angka yang dideklarasikan a yaitu 15.

5. Input

Input artinya suatu data yang dimasukkan oleh user setelah atau saat proses output terjadi.

```
fadlila@fadlila-VirtualBox:~$ nano input.sh
fadlila@fadlila-VirtualBox:~$ bash input.sh
Hallo, masukkan nama anda : Fadlila Agustina
Selamat datang Fadlila Agustina
```

- Membuat file bernama input.sh menggunakan perintah nano.
- Untuk menampilkan output dari skrip input.sh yang dibuat di dalam nano dapat menggunakan perintah bash.

```
#!/bin/bash

echo -n "Hallo, masukkan nama anda : ";
read nama;
echo "Selamat datang $nama";
```

- Di atas merupakan skrip input.sh yang dibuat menggunakan nano.
- Read merupakan fungsi yang digunakan agar user dapat menginput nama yang diinginkan dan disimpan ke dalam variable, yang mana variable tersebut akan dipanggil di baris selanjutnya.

6. Output

Output merupakan hasil dari suatu proses.

a) Output1

```
fadlila@fadlila-VirtualBox:~$ nano output1.sh
fadlila@fadlila-VirtualBox:~$ bash output1.sh
Siapa namamu?
Fadlila Agustina

Hai Fadlila Agustina!
Selamat datang di praktikum Sistem Operasi:))))
```

- Membuat file bernama output1.sh menggunakan perintah nano.
- Untuk menampilkan output dari skrip output1.sh yang dibuat di dalam nano dapat menggunakan perintah bash.

```
#!/bin/bash
matakuliah="Sistem Operasi"

echo "Siapa namamu?"
read nama
echo -e "\nHai $nama!\nSelamat datang di praktikum $matakuliah:))))"
```

- Di atas merupakan skrip output1.sh yang dibuat menggunakan perintah nano.
- Matakuliah mendeklarasikan Sistem Operasi.
- Perintah read di atas digunakan agar user dapat memasukkan nama yang diinginkan dan disimpan ke dalam variable, yang mana variable tersebut akan dipanggil di baris selanjutnya.

b) Output2

```
fadlila@fadlila-VirtualBox:~$ nano output2.sh
fadlila@fadlila-VirtualBox:~$ bash output2.sh
OS : Ubuntu 19.04 LTS
3
55,00 float
55,0 float
```

- Membuat file bernama output2.sh menggunakan perintah nano.
- Untuk menampilkan output dari skrip output2.sh yang dibuat di dalam nano dapat menggunakan perintah bash.

```
#!/bin/bash

# Inisialisasi Var
a=55;
b=4;
distroLinux="Ubuntu 19.04 LTS";
let c=a%b;

# Output Printf
printf "OS : $distroLinux \n";
printf "$c \n";
printf "%.2f float \n" $a;
printf "%.1f float \n" $a;
```

- Di atas merupakan skrip output2.sh yang dibuat menggunakan perintah nano.
- Langkah pertama yang dilakukan adalah inisialisasi variable.
- Script tersebut menggunakan operasi aritmatika yaitu modulus.
- Printf berbeda dengan echo. Printf menyediakan pembuatan string pemformatan dan menawarkan status berhenti bukan nol ketika gagal, sedangkan echo biasanya keluar dengan status nol dan biasanya mengeluarkan input yang dipimpin oleh karakter akhir baris.
- %.2f artinya hasil dari pengoperasian hanya akan terdapat 2 angka di belakang koma.

7. Percabangan

Percabangan merupakan kontrol seleksi yang memungkinkan program untuk menjalankan suatu perintah berdasarkan kondisi tertentu. Percabangan memiliki berbagai operator, yaitu sebagai berikut:

- -eq digunakan untuk memeriksa apakah nilai kedua operan sama (==)
- -ne untuk memeriksa apakah nilai kedua operan tidak sama (!=)
- -gt untuk memeriksa apakah nilai operan kiri lebih besar daripada operan kanan (>)
- -lt untuk memeriksa apakah nilai operan kiri lebih kecil daripada operan kanan (<)
- -ge untuk memeriksa apakah nilai operan kiri lebih besar atau sama dengan operan kanan (>=)
- -le untuk memeriksa apakah nilai operan kiri lebih kecil atau sama dengan operan kanan (<=)

a. Percabangan1

```
fadlila@fadlila-VirtualBox:~$ nano percabangan1.sh
fadlila@fadlila-VirtualBox:~$ bash percabangan1.sh
a lebih besar dari b
```

- Membuat file bernama percabangan1.sh menggunakan perintah nano.
- Untuk menampilkan output dari skrip percabangan1.sh yang dibuat di dalam nano dapat menggunakan perintah bash.

```
#!/bin/bash

a=15
b=7

if [ $a == $b ]
then
    echo "a sama dengan b"
elif [ $a -gt $b ]
then
    echo "a lebih besar dari b"
elif [ $a -lt $b ]
then
    echo "a lebih kecil dari b"
else
    echo "Tidak ada kondisi yang memenuhi"
fi
```

- Di atas merupakan skrip percabangan1.sh yang dibuat menggunakan perintah nano.
- Script di atas dapat dilihat bahwa perbandingan angka yang terjadi pada a dan b dapat berubah statusnya ketika angka yang dideklarasikan a dan b berubah.

b. Percabangan2

```
fadlila@fadlila-VirtualBox:~$ nano percabangan2.sh
fadlila@fadlila-VirtualBox:~$ bash percabangan2.sh
Jajan apa yang kamu suka ?
-pentol ?
batagor ?
cireng ?
pentol
Pentol buk mah wenak slur!
fadlila@fadlila-VirtualBox:~$ bash percabangan2.sh
Jajan apa yang kamu suka ?
-pentol ?
batagor ?
cireng ?
gorengan
Makanan yang kamu suka gaenak hehe
```

- Membuat file bernama percabangan2.sh menggunakan perintah nano.
- Untuk menampilkan output dari skrip percabangan2.sh yang dibuat di dalam nano dapat menggunakan perintah bash.
- Ketika jawaban yang dipilih user tidak ada dari beberapa penawaran yang diberikan, output yang akan muncul adalah seperti baris yang terakhir.

```
#!/bin/bash

printf "Jajan apa yang kamu suka ?\n"-
printf "pentol ?\n"
printf "batagor ?\n"
printf "cireng ?\n"

read jajan

case "$jajan" in
    "pentol")
        echo "Pentol buk mah wenak slur!"
        ;;
    "batagor")
        echo "Batagore mas budi mantap bat"
        ;;
    "cireng")
        echo "Cireng kantin rasane unch-unch"
        ;;
    *)
        echo "Makanan yang kamu suka gaenak hehe"
        ;;
esac
```

- Di atas merupakan skrip percabangan2.sh yang dibuat menggunakan perintah nano.
- Script di atas disediakan beberapa penawaran yang dapat dipilih oleh user.
- Dalam case jajan, terdapat beberapa peluang user untuk memilih jawaban yang sudah dibuat di dalam script.

Latihan Soal

```
Thank You
fadlila@fadlila-VirtualBox:~$ nano Tugas_2.sh
fadlila@fadlila-VirtualBox:~$ bash Tugas_2.sh
```

- Membuat file bernama Tugas_2.sh menggunakan perintah nano.

```
#!/bin/bash

#penjumlahan
echo "LINUX JUMLAHIN ANGKA YG KAMU MASUKIN YAA"
echo "Masukkan nilai p: "
read p
echo "Masukkan nilai q: "
read q
let hasil=$p+$q
echo "Jadi, hasil dari penjumlahan $p + $q = $hasil"

sleep 1

#perkalian
echo "SELANJUTNYA LINUX BAKAL KALIIN "
echo "Masukkan nilai a: "
read a
echo "Masukkan nilai b: "
read b
let kali=$a*$b
echo "Jawaban dari perkalian tersebut adalah $kali"

sleep 1

#perbandingan
echo "YANG TERAKHIR BANDINGIN NILAI YAGESYAK"
sleep 1
echo "IYADONG KALO AKU DIBANDINGIN SM MANTAN KAMU YO NDAK MAMPU HEHE"
echo "Masukin nilai x: "
read x
echo "Masukin nilai y: "
read y
echo "SATU DUA TIGA CLING"
sleep 1

if [ $x -gt $y ]
then
    echo "Nilai x lebih besar dari nilai y"
elif [ $x -lt $y ]
then
    echo "Nilai x lebih kecil dari nilai y"
elif [ $x -eq $y ]
then
    echo "Nilainya sama gaiss"
else
    echo "Maaf coba masukin nilai yg beda yaa"
fi

sleep 1

echo "THANK YOUU"
```

- Di atas merupakan script yang saya buat sesuai dengan soal menggunakan percabangan sederhana aritmatika yang menerapkan beberapa konsep pemrograman bash.
- Perintah sleep berfungsi untuk memberikan jeda dalam satuan detik.

```

fadlila@fadlila-VirtualBox:~$ bash Tugas_2.sh
LINUX JUMLAHIN ANGKA YG KAMU MASUKIN YAA
Masukkan nilai p:
50
Masukkan nilai q:
75
Jadi, hasil dari penjumlahan  $50 + 75 = 125$ 
SELANJUTNYA LINUX BAKAL KALIIN
Masukkan nilai a:
85
Masukkan nilai b:
15
Jawaban dari perkalian tersebut adalah 1275
YANG TERAKHIR BANDINGIN NILAI YAGESYAK
IYADONG KALO AKU DIBANDINGIN SM MANTAN KAMU YO NDAK MAMPU HEHE
Masukin nilai x:
117
Masukin nilai y:
78
SATU DUA TIGA CLING
Nilai x lebih besar dari nilai y
THANK YOUU

LINUX JUMLAHIN ANGKA YG KAMU MASUKIN YAA
Masukkan nilai p:
213
Masukkan nilai q:
4215
Jadi, hasil dari penjumlahan  $213 + 4215 = 4428$ 
SELANJUTNYA LINUX BAKAL KALIIN
Masukkan nilai a:
213
Masukkan nilai b:
4321
Jawaban dari perkalian tersebut adalah 920373
YANG TERAKHIR BANDINGIN NILAI YAGESYAK
IYADONG KALO AKU DIBANDINGIN SM MANTAN KAMU YO NDAK MAMPU HEHE
Masukin nilai x:
1235342
Masukin nilai y:
1235342
SATU DUA TIGA CLING
Nilainya sama gaiss
THANK YOUU

```

- Di atas adalah output yang muncul hasil dari script yang telah dibuat.
- Sesuai dengan gambar di atas, sudah melakukan 2 percobaan dan berhasil.