

Nama : Alisa Jihan Azizah

NPM : 21083010064

Mata Kuliah : Sistem Operasi – B

Laporan Materi-7 : Bash Function

Function

Fungsi merupakan bagian script atau program yang berisi kumpulan beberapa statement yang melaksanakan tugas tertentu. Dengan subrutin kode script kita tentunya lebih sederhana dan terstruktur, karena sekali fungsi telah dibuat dan berhasil maka akan dapat digunakan kapan saja kita inginkan.

Deklarasi fungsi sendiri yaitu

```
nama_fungsi() {  
    perintah1  
    perintah2  
    ...  
    Perintahn  
}
```

Contohnya dapat dilakukan sebagai berikut.

1. Membuat file baru pada terminal dengan menggunakan perintah nano (contoh : nano function1.sh)
2. Pada tampilan nano, masukkan script bash function seperti pada gambar dibawah ini



```
GNU nano 6.2                                Function1.sh  
#!/bin/bash  
  
# mendeklarasikan fungsi  
nama() {  
    echo "Siapa namamu?"  
    read nama  
}  
npm() {  
    echo "Sebutkan NPM-mu"  
    read npm  
    echo -e "Hai, $nama dengan npm $npm, selamat datang \n di praktikum sis>  
}  
  
# memanggil fungsi  
nama  
npm
```

Pada keterangan dari gambar tersebut, pertama kita mendeklarasikan fungsi nama dan fungsi npm. Pada fungsi nama, echo “Siapa namamu?” dan kemudian ketik

read nama. Sedangkan pada fungsi npm, echo “sebutkan npm-mu” kemudian ketik read npm dan echo dari kalimat diatas. Untuk memanggil fungsinya, kita ketik nama dan npm.

3. Hasil Outputnya sebagai berikut :

```
alisa@alisa-VirtualBox:~$ nano Function1.sh
alisa@alisa-VirtualBox:~$ bash Function1.sh
Siapa namamu?
Alisa
Sebutkan NPM-mu
21083010064
Hai, Alisa dengan npm 21083010064, selamat datang
di praktikum sistem operasi yang seru ini ya!
```

Pada contoh Function 2 juga sama, tetapi yang membedakan adalah kita menggunakan ‘function’ nama_fungsi untuk mendeklarasikannya. Bisa dilihat script dibawah ini :

```
GNU nano 6.2 Function2.sh
#!/bin/bash

# mendeklarasikan fungsi
function nama {
    echo "Siapa namamu?"
    read nama
}
function npm {
    echo "Sebutkan NPM-mu"
    read npm
    echo -e "Hai $nama dengan npm $npm, selamat datang \n di praktikum sist>
}

# memanggil fungsi
nama
npm
```

Maka, hasil outputnya yaitu :

```
alisa@alisa-VirtualBox:~$ bash Function2.sh
Siapa namamu?
Alisa
Sebutkan NPM-mu
21083010064
Hai Alisa dengan npm 21083010064, selamat datang
di praktikum sistem operasi yang seru ini ya!
alisa@alisa-VirtualBox:~$
```

Nested Function (Fungsi Bersarang)

Nested function adalah fungsi yang didefinisikan di dalam fungsi lain. Fungsi bersarang ini dapat mengakses variabel dari scope (lingkup) fungsi pembungkusnya. Contohnya dapat dilakukan sebagai berikut.

1. Membuat file baru pada terminal dengan menggunakan perintah nano. (contoh : nested_function.sh)
2. Setelah muncul tampilan nano, kemudian masukkan script bash seperti gambar dibawah ini.

```

GNU nano 6.2                               Nested_Function.sh *
#!/bin/bash

# mendeklarasikan fungsi
nama() {
    echo "Siapa namamu?"
    read nama
    npm                                # <----- memanggil fungsi di dalam fungsi [fu>
}
npm() {
    echo "Sebutkan NPM-mu?"
    read npm
    echo -e "Hai $nama dengan npm $npm, selamat datang \n di praktikum sist>
}

# memanggil fungsi
nama

```

Pada keterangan tersebut menjelaskan pertama mendeklarasikan fungsi nama yang nanti akan memanggil fungsi lain didalam fungsinya tersebut. Fungsi lainnya adalah fungsi npm. Setelah itu isi perintah apa saja yang akan dieksekusi di tiap masing – masing fungsi. Untuk memanggilnya, kita hanya memasukkan ‘nama’.

3. Hasil outputnya sebagai berikut :

```

alisa@alisa-VirtualBox:~$ bash Nested_Function.sh
Siapa namamu?
Alisa
Sebutkan NPM-mu?
21083010064
Hai Alisa dengan npm 21083010064, selamat datang
di praktikum sistem operasi yang seru ini ya!

```

Bash Function dengan Parameter (x, y)

Untuk memanggil fungsi bash dengan parameter dapat dilakukan sebagai berikut.

1. Membuat file baru dengan perintah nano (contoh : bash_function.sh).
2. Kemudian kita masukkan script bash seperti gambar dibawah ini.

```

GNU nano 6.2
#!/bin/bash

# mendeklarasikan fungsi
identitas() {
    parameter1=$1
    parameter2=$2
    parameter3=$3
    echo "$parameter1"
    echo "$parameter2"
    echo "$parameter3"
}

echo "Masukkan nama : "
read a
echo "Masukkan NPM : "
read b
echo "Hobimu apa?"
read c

printf "\n"
identitas $a $b $c

```

Keterangan dari gambar ini menjelaskan pertama kita mendeklarasikan identitas sebagai fungsi. Selanjutnya, masukan parameter1, parameter2, dan parameter3. Setelah itu masukan a, b, dan c nya. Untuk memanggil fungsi tersebut, dapat menggunakan identitas \$a \$b \$c.

3. Maka hasil outputnya sebagai berikut:

```

alisa@alisa-VirtualBox:~$ bash Function_Parameter.sh
Masukkan nama :
Alisa
Masukkan NPM :
21083010064
Hobimu apa?
ngoding

Alisa
21083010064
ngoding

```

Soal Latihan

Buatlah program bash function yang dapat menghitung luas bidang persegi!

Contoh output:

Masukkan Panjang :

10

Masukkan lebar:

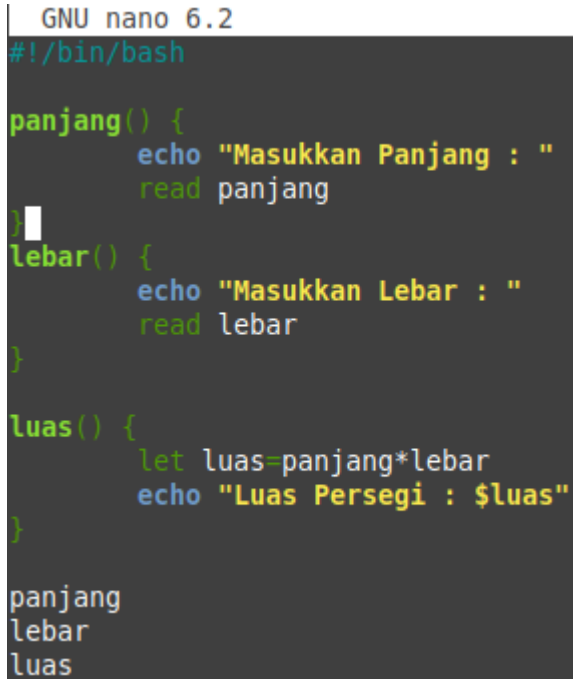
4

Luas persegi :

40

Hal pertama yang dilakukan adalah membuat terminal pada linux.

1. Membuat file baru dengan perintah nano (contoh : tugas_7.sh)
2. Setelah muncul tampilan nano, kemudian masukkan script bash yang akan dieksekusi.



```
GNU nano 6.2
#!/bin/bash

panjang() {
    echo "Masukkan Panjang : "
    read panjang
}

lebar() {
    echo "Masukkan Lebar : "
    read lebar
}

luas() {
    let luas=panjang*lebar
    echo "Luas Persegi : $luas"
}

panjang
lebar
luas
```

Pada keterangan script bash diatas, saya menggunakan function untuk mendeklarasikan masing – masing fungsinya. Fungsinya tersebut terdiri dari Panjang, lebar, dan luas. Pada fungsi Panjang, perintahnya adalah memasukkan nilai panjangnya dan pada fungsi lebar, perintahnya adalah memasukkan nilai lebar. Sedangkan pada fungsi luas, untuk perintahnya saya mendefinisikan luas merupakan hasil perkalian dari Panjang * lebar. Setelah itu saya panggil fungsinya dengan mengetikkan Panjang, lebar, dan luas.

3. Sehingga hasil outputnya adalah sebagai berikut.

```
alisa@alisa-VirtualBox:~$ bash Tugas_7.sh
Masukkan Panjang :
10
Masukkan Lebar :
4
Luas Persegi : 40
```