

Nama : Alvin Ryan Dana
NPM : 21083010035
Kelas : Sistem Operasi B

Bash Function

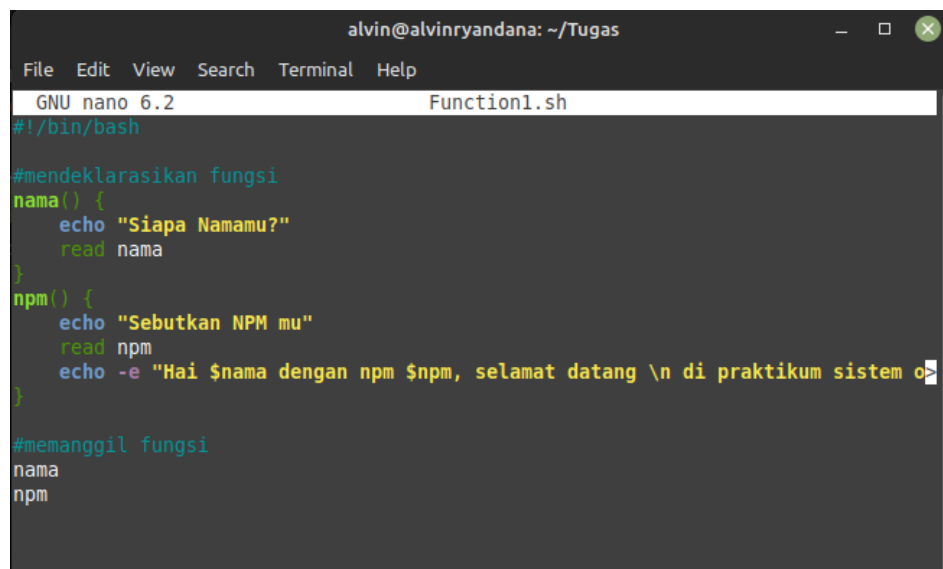
Fungsi merupakan bagian script atau program yang berisi kumpulan beberapa statement yang melaksanakan tugas tertentu. Dengan subrutin kode script kita tentunya lebih sederhana dan terstruktur, karena sekali fungsi telah dibuat dan berhasil maka akan dapat digunakan kapan saja kita inginkan

1. Deklarasi Function Syntax - 1 :

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

Cara untuk mendeklarasikan function dan memanggil function

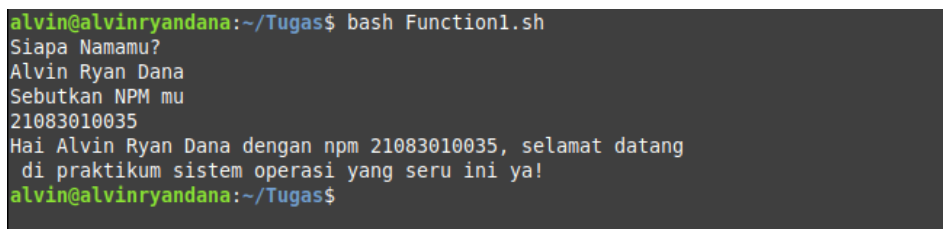
Contoh :



The screenshot shows a terminal window titled 'alvin@alvinryandana: ~/Tugas'. Inside, the GNU nano 6.2 editor is open with a file named 'Function1.sh'. The script content is as follows:

```
#!/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 sistem o  
}  
  
#memanggil fungsi  
nama  
npm
```

Output :



The screenshot shows the terminal output after running 'bash Function1.sh'. The output is:

```
alvin@alvinryandana:~/Tugas$ bash Function1.sh  
Siapa Namamu?  
Alvin Ryan Dana  
Sebutkan NPM mu  
21083010035  
Hai Alvin Ryan Dana dengan npm 21083010035, selamat datang  
di praktikum sistem operasi yang seru ini ya!  
alvin@alvinryandana:~/Tugas$
```

2. Deklarasi Function Syntax - 2 :

```
function namaFungsi {  
    Statement-1 command...
```

Statement-2 command...

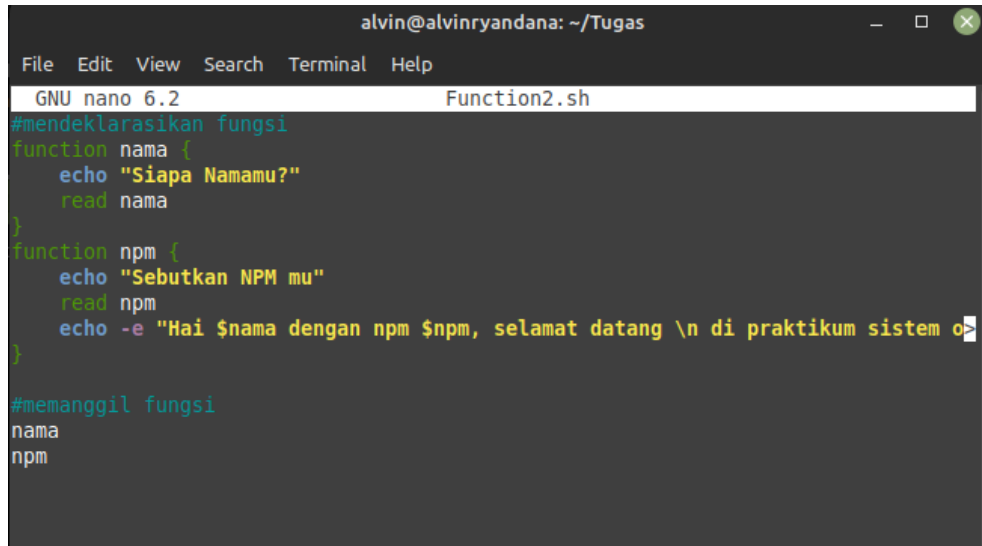
Statement-3 command...

Etc

}

Cara lain untuk mendeklarasikan function, dapat dilakukan dengan menambahkan function sebelum nama fungsi

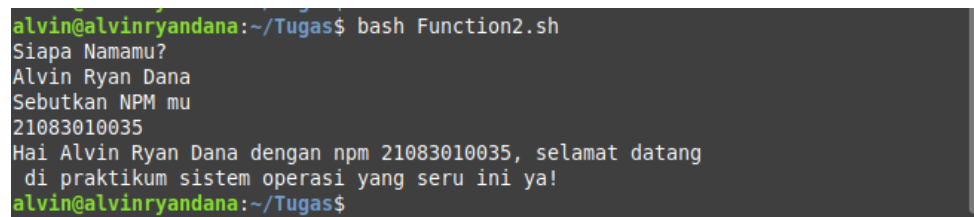
Contoh :



```
alvin@alvinryandana: ~/Tugas
File Edit View Search Terminal Help
GNU nano 6.2 Function2.sh
#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 sistem o
}

#memanggil fungsi
nama
npm
```

Output :

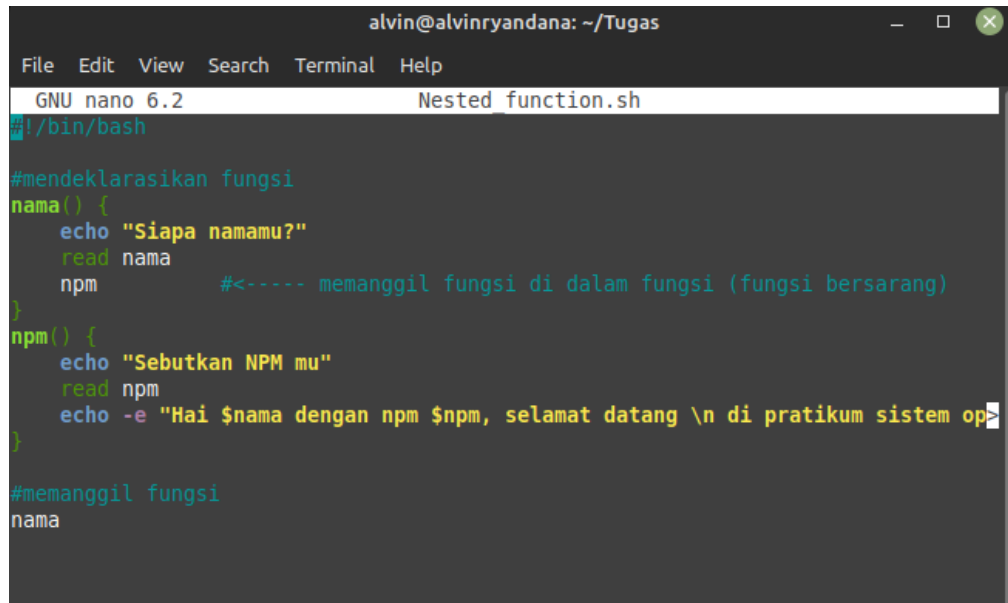


```
alvin@alvinryandana:~/Tugas$ bash Function2.sh
Siapa Namamu?
Alvin Ryan Dana
Sebutkan NPM mu
21083010035
Hai Alvin Ryan Dana dengan npm 21083010035, selamat datang
di praktikum sistem operasi yang seru ini ya!
alvin@alvinryandana:~/Tugas$
```

3. Nested Function (Fungsi bersarang)

Untuk melakukan Nested Function pada fungsi harus memanggil salah satu fungsi lainnya. Dalam contoh fungsi nama memanggil fungsi lainnya yaitu npm, kemudian ketika menampilkan fungsi nama fungsi dari npm akan juga ditampilkan

Contoh :



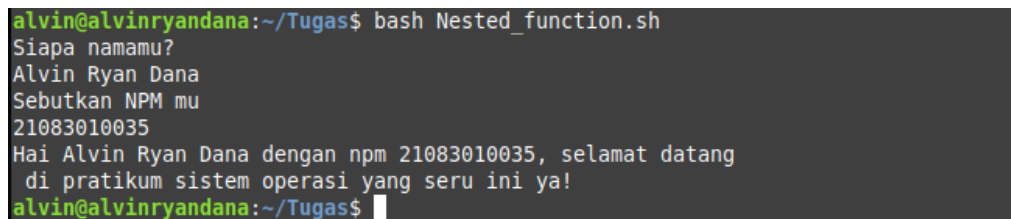
```
alvin@alvinryandana: ~/Tugas
File Edit View Search Terminal Help
GNU nano 6.2 Nested_function.sh
#!/bin/bash

#mendeklarasikan fungsi
nama() {
    echo "Siapa namamu?"
    read nama
    npm      #<----- memanggil fungsi di dalam fungsi (fungsi bersarang)
}

npm() {
    echo "Sebutkan NPM mu"
    read npm
    echo -e "Hai $nama dengan npm $npm, selamat datang \n di pratikum sistem op>
}

#memanggil fungsi
nama
```

Output :

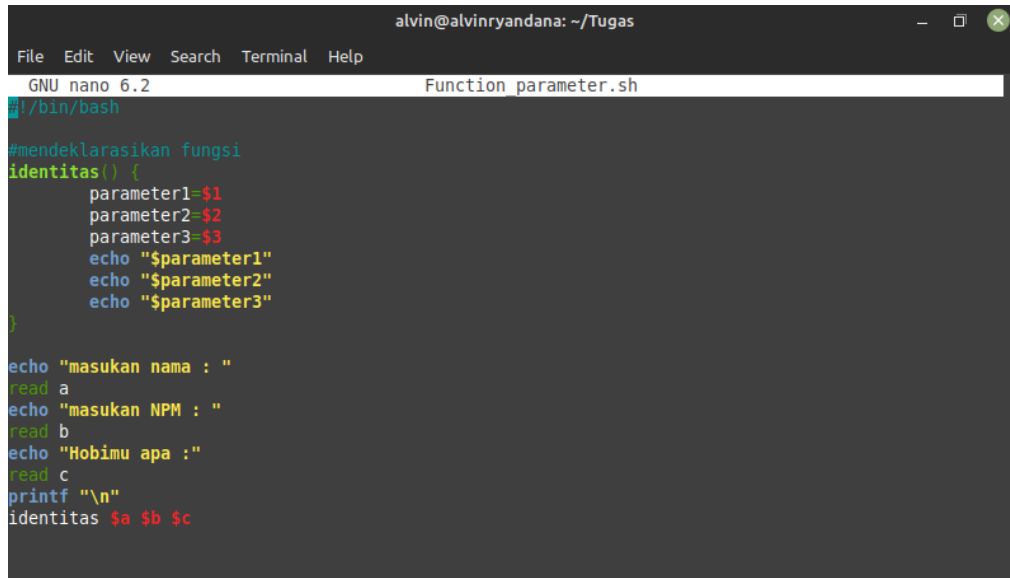


```
alvin@alvinryandana:~/Tugas$ bash Nested_function.sh
Siapa namamu?
Alvin Ryan Dana
Sebutkan NPM mu
21083010035
Hai Alvin Ryan Dana dengan npm 21083010035, selamat datang
di pratikum sistem operasi yang seru ini ya!
alvin@alvinryandana:~/Tugas$
```

4. Bash Function dengan parameter (x,y)
- ```
function namaFungsi atau namafungsi() {
 par1=$1
 par2=$2
 command on $par1
}
```

Pada script bisa menambahkan parameter kemudian memberikan perintah kepada parameter tersebut

Contoh :



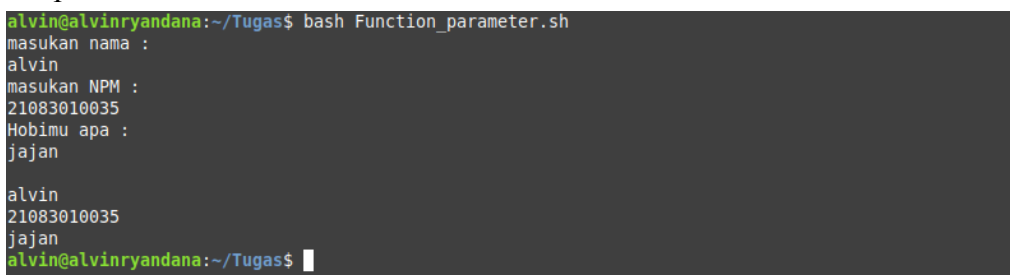
```
alvin@alvinryandana: ~/Tugas
File Edit View Search Terminal Help
GNU nano 6.2 Function_parameter.sh
#!/bin/bash

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

echo "masukan nama : "
read a
echo "masukan NPM : "
read b
echo "Hobimu apa :"
read c
printf "\n"
identitas $a $b $c
```

Pada contoh membuat fungsi dengan tiga parameter yang kemudian akan ditampilkan outputnya menggunakan echo, lalu untuk memasukan input dari user dapat menggunakan read. input dari user kemudian dimasukan ke fungsi yang sudah dibuat identitas

Output :



```
alvin@alvinryandana:~/Tugas$ bash Function_parameter.sh
masukan nama :
alvin
masukan NPM :
21083010035
Hobimu apa :
jajan

alvin
21083010035
jajan
alvin@alvinryandana:~/Tugas$
```

Soal Latihan :

Buatlah program bash function yang dapat menghitung luas bidang persegi!

Contoh Output :

Masukkan Panjang :

10

Masukkan Lebar :

4

Luas Persegi :

40

Pada soal latihan ini saya membuat 3 fungsi yaitu panjang, lebar, dan luas

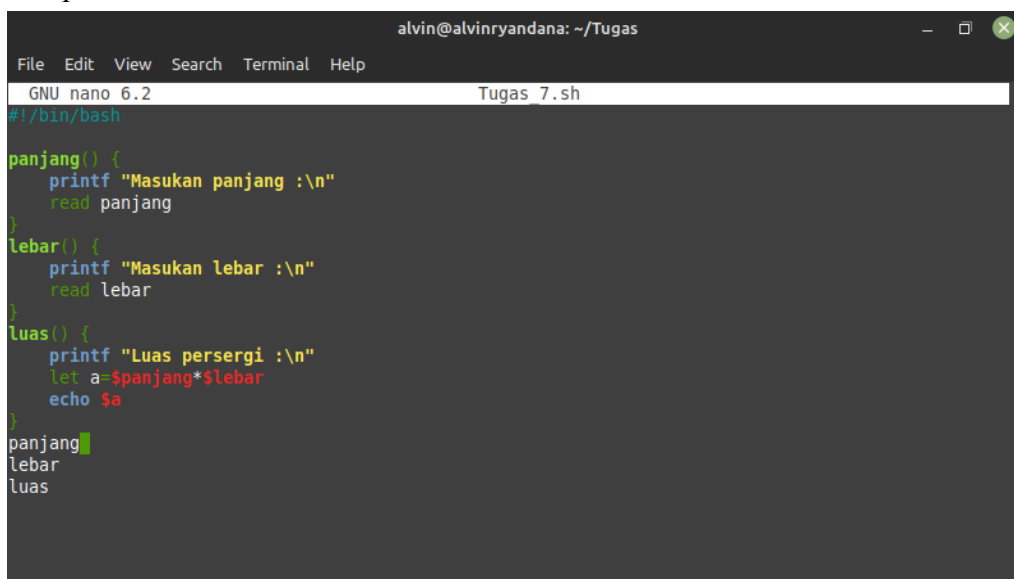
-Fungsi panjang berguna untuk memasukan panjang dari inputan user menggunakan read

-Fungsi lebar berguna untuk memasukan lebar dari inputan user menggunakan read

-Fungsi luas berguna untuk menghitung luas dari persegi menggunakan let untuk melakukan operasi matematika yang dimana  $let\ a=\$panjang*\$lebar$ , artinya variabel a adalah hasil dari operasi panjang \* lebar

Kemudian memanggil ketiga fungsi tersebut

Script :

A screenshot of a terminal window titled 'alvin@alvinryandana: ~/Tugas'. The window shows the GNU nano 6.2 editor with a file named 'Tugas\_7.sh'. The script contains three functions: 'panjang()' which prompts for and reads the length, 'lebar()' which prompts for and reads the width, and 'luas()' which calculates the area using 'let a=\$panjang\*\$lebar' and echoes the result. Below the functions, the three functions are called in sequence: 'panjang', 'lebar', and 'luas'.

```
alvin@alvinryandana: ~/Tugas
File Edit View Search Terminal Help
GNU nano 6.2 Tugas_7.sh
#!/bin/bash

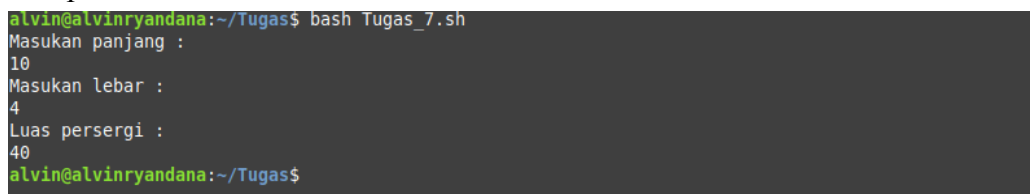
panjang() {
 printf "Masukan panjang :\n"
 read panjang
}

lebar() {
 printf "Masukan lebar :\n"
 read lebar
}

luas() {
 printf "Luas persergi :\n"
 let a=$panjang*$lebar
 echo $a
}

panjang
lebar
luas
```

Output :

A screenshot of a terminal window showing the execution of the script. The prompt is 'alvin@alvinryandana:~/Tugas\$'. The user enters 'bash Tugas\_7.sh'. The script prompts for 'Masukan panjang :' and the user enters '10'. It then prompts for 'Masukan lebar :' and the user enters '4'. Finally, it outputs 'Luas persergi : 40' and returns to the shell prompt.

```
alvin@alvinryandana:~/Tugas$ bash Tugas_7.sh
Masukan panjang :
10
Masukan lebar :
4
Luas persergi :
40
alvin@alvinryandana:~/Tugas$
```