

I ❤️ #!/bin/bash



Shell Scripting - Module 6

Functions

Returning values from function

Nested Functions

Functions

- we will discuss in detail about the shell functions.
- Functions enable you to break down the overall functionality of a script into smaller, logical subsections, which can then be called upon to perform their individual tasks when needed.
- Using functions to perform repetitive tasks is an excellent way to create code reuse.

Syntax :

```
function_name () {  
    list of commands  
}
```

Example :

```
#!/bin/sh

# Define your function here
Hello () {
    echo "Hello World"
}

# Invoke your function
Hello
```

Output :

```
Helo World
```

Returning values from function

If incase you want to return any value from fucntion, you can use return along with it.

Syntax :

```
return code
```

Example :

```
#!/bin/bash

# Define your function here
Hello () {
    echo "Hello World $1 $2 $3"
    return 10
}

# Invoke your function
Hello Arun and Arjun

# Capture value returnd by last command
ret=$?

echo "Your role number is $ret"
```

Output

```
Hello World Arun and Arjun  
Your role number is 10
```

Nested Functions

Let's see how to call one function from another.

Example :

```
#!/bin/bash  
  
# Calling one function from another  
number_one () {  
    echo "This is the first function speaking..."  
    number_two  
}  
  
number_two () {  
    echo "This is now the second function speaking..."  
}  
  
# Calling function one.  
number_one
```

Output :

```
This is the first function speaking...  
This is now the second function speaking...
```