

Function

```
function functionName () {  
    Commands to be executed  
}
```

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

```
# Define your function here
```

```
Hello () {  
    echo "Hello World"  
}
```

```
# Invoke your function
```

```
Hello
```

Passing Arguments to Bash Functions

```
#!/bin/bash
```

```
new_function() {  
    echo "Hi! How's it going?"  
    echo "I was invoked with $1 $2"  
}
```

```
new_function math code
```

```
#!/bin/bash
```

```
user_function () {  
    div=$(( $1 / $2 ))  
    echo "dividing 60 by 10 is : $div"  
}  
user_function 60 10
```

```
sum() {  
    local value1=$1  
    local value2=$2  
    local result=$((value1 + value2))  
    echo "The sum of ${value1} and ${value2} is ${result}"  
}  
sum 3 6
```

Local Variable

A local variable is a special type of variable which has its scope only within a specific function or block of code. Local variables can override the same variable name in the larger scope.

We will understand all the concepts related to Local variables using different examples.

Example 1: Limited Scope

This example depicts that a local variable written inside a function has its scope limited only within the function.

Shell Script:

```
#!/bin/sh

getNUM(){
    NUM=100 #local variable
    echo "$NUM - inside function"
}

echo "$NUM - outside function"
```

getNUM

Output:

```
- outside function
100 - inside function
```

In this example, NUM is a local variable because it is within the getNUM() function. When we are accessing the value of it from the function itself, then we are getting it, but when we are trying to access the same from outside the function, it is not visible.

Below is the terminal shell pictorial representation after executing the following script:-

Example 2: Overrides Global Variable

This example depicts that a local variable can override the global variable in its scope.

Shell Script:

```
#!/bin/sh
NUM=200 #global variable
getNUM(){
    NUM=100 #local variable
    echo "$NUM - inside function"
}

echo "$NUM - outside function"
```

getNUM

Output:

```
200 - outside function
100 - inside function
```

Here, there is a global variable and a local variable within the getNUM() function having the same name. The local variable within its scope i.e. the getNUM() function, override it but outside its scope, the global variable is accessed.

Below is the terminal shell pictorial representation after executing the following script:-

