

Functions

```
#!/bin/bash

function function-name()
{
    #function code
}

function-name #calling the function
```

Positional Parameters

Functions can accept parameters and access using positional \$1 \$2 \$3 and so on.

The \$0 is still the name of the script itself

To send data to a function, supply data after the function name.

```
#!/bin/bash
function hello(){
    echo "Hello $1"
}
hello Mark
```

Variable Scope

- By default all variables are global.
- The variable has to be defined before it can be used.

Local Variables

A local variable can only be accessed within the function in which it is declared. Used the `local` keyword before the variable name. The local keyword can only be used inside a function

Return Codes

Functions work like shell scripts within a shell script

The exit status in a function is called return code. If no return statement is used, then the exit status of the function is the exit status of the last command executed in the function.

Return statement only accepts numbers, only integers between 0 and 255 can be used as exit status.

Backup File

The `backup_file` function will create a backup of a file and place it into the `/var/tmp` directory.

Is it recommended to use this function when a script modifies several files to make sure we have a copy to view or restore.

If it is a file and it does exist, a variable called `BACKUP_FILE` is created. It starts off with `/var/tmp` followed by the basename of the passed file, the current date and the PID of the shell script

- The **basename** command removes any leading directory components and returns just the file name. The basename of `/etc/hosts` is just `hosts`.
- The **date** command is using a nice format of the year, followed by the month and finally the day, all separated by dashes.
- special variable `$$` represents the PID of currently running shell script. So if you run the script multiple times on the same day PID will be different each time on the same day

```
function backup_file()
{
    if[ -f "$1" ]
    then
        local BACKUP_FILE="/var/tmp/${basename ${1}}.${date+%F}.$$"
        echo "Backing up $1 to ${BACKUP_FILE}"
        cp $1 $BACKUP_FILE
    fi
}

backup_file /etc/hosts

if[ $? -eq 0 ]
then
    echo "Backup succeeded!"
fi
```