

# Creating your first shell script

In this section, we will take a look at creating your first shell script.

To Create a shell script name create-and-launch-script.sh

```
$ vi create-and-launch-script.sh
```

## Creating your First Script

```
$ mkdir lunar-mission
```

```
$ rocket-add lunar-mission
```

```
$ rocket-start-power lunar-mission  
$ rocket-internal-power lunar-mission  
$ rocket-start-sequence lunar-mission  
$ rocket-start-engine lunar-mission  
$ rocket-lift-off lunar-mission
```

```
$ rocket-status lunar-mission
```

```
create-and-launch-rocket.sh
```

```
mkdir lunar-mission
```

```
rocket-add lunar-mission
```

```
rocket-start-power lunar-mission  
rocket-internal-power lunar-mission  
rocket-start-sequence lunar-mission  
rocket-start-engine lunar-mission  
rocket-lift-off lunar-mission  
rocket-status lunar-mission
```

## Run script as command

- There are different ways to execute a shell script
  - Execute a shell script with `bash` command

```
$ bash create-and-launch-script.sh
```

- Execute a shell script as an `executable`

```
$ create-and-launch-script.sh
```

**Note** : It is a best practice to not name your script with the `.sh` extension when you would like to create an executable of a script.

```
$ create-and-launch-script
```

## Configure a script to run as command

- Whenever a command is run at a linux system, the O.S looks at the path configured in the `$PATH` environment variable to locate the executable or script for the command.
- If it cannot find the command in the `$PATH` then a `command not found` error will be thrown.
- To add our script as a command, append the path to the directory containing the script to the end of the `$PATH` variable.

```
$ export PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/home/michael
```

- A better way to do this is to append the path to the directory to `$PATH` variable

```
$ export PATH=$PATH:/home/michael
```

- Run the command

```
$ create-and-launch-script
```

- To see the location of the command

```
$ which create-and-launch-script
```

## Run script as Command

```
$ bash create-and-launch-rocket.sh
```

```
$ create-and-launch-rocket ✖  
create-and-launch-rocket : command not found
```

```
$ echo $PATH  
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
```

```
$ export PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/home/michael
```

or

```
$ export PATH=$PATH:/home/michael
```

```
$ create-and-launch-rocket ✔
```

```
$ which create-and-launch-rocket  
/home/michael/create-and-launch-rocket
```

### 🔗 Executing a script

- For a shell script to work, we must set the correct permissions to the file, if the permissions are not set a "Permission Denied" error will be thrown when you run the script for the first time.
- Inspect the file permissions

```
$ ls -l /home/michael/create-and-launch-script
```

- Add executable permissions to a file and then inspect the file to check if it got write permissions.

```
$ chmod +x /home/michael/create-and-launch-script  
$ ls -l /home/michael/create-and-launch-script
```

- Now, run the script

```
$ /home/michael/create-and-launch-script
```

## Executing a Script

```
$ /home/michael/ create-and-launch-rocket ✗  
-bash: ./create-and-launch-rocket: Permission denied  
  
$ ls -l /home/michael/create-and-launch-rocket  
-rw-rw-r-- 1 michael michael 19 Mar 16 09:47 create-and-launch-rocket
```

```
$ chmod +x /home/michael/create-and-launch-rocket  
  
$ ls -l /home/michael/create-and-launch-rocket  
-rwx-rwx-r-x 1 michael michael 19 Mar 16 09:47 create-and-launch-rocket
```

```
$ /home/michael/create-and-launch-rocket ✓
```

### Best Practices

#### Best Practice

*"Give your script a name  
that makes sense"*

*good:*

*create-and-launch-rocket*

*bad:*

*script1.sh*

*myscript.sh*

*test.sh*

*"Leave out .sh extension for  
executables"*

*good:*

*create-and-launch-rocket*

*bad:*

*create-and-launch-rocket.sh*