

Paths

 coursera.org/learn/linux-for-developers/supplement/YH1tQ/paths

The path is a critical aspect of your environment, and is encapsulated in the **PATH** environment variable. On an RHEL 7 system for a user named **student**, we get:

```
3
1
2
/usr/local/sbin:/usr/sbin:/home/student/.local/bin:/home/student/bin
$ echo $PATH
/usr/lib64/qt-3.3/bin:/usr/lib64/ccache:/usr/local/bin:/usr/bin:\
```



(Note we have had to split the path across across two lines in the output.)

When a user tries to run a program, the path is searched (from left to right) until an executable program or script is found with that name. You can see what would be found with the **which** command, as in:

```
1
2
$ which --skip-alias emacs
/usr/bin/emacs
```



Note that if there was a **/usr/local/bin/emacs**, it would be executed instead, since it is earlier in the path.

It is easy to add directories to your path, as in:

```
1
2
3
```

```
$ MY_BIN_DIR=$HOME/my_bin_dir

$ export PATH=$MY_BIN_DIR:$PATH

$ export PATH=$PATH:$MY_BIN_DIR
```



with the first form prepending your new directory and the second appending it to the path.

Note that the current directory is noted by `./` and the directory up one level by `../`.

The current directory is never placed in the path by default. Thus, if you want to run **foobar** in the current directory, you must say:

1

```
$ ./foobar
```



for it to work.

You can save changes to your path by putting them in your shell initialization file, **.bashrc** in your home directory.

```
student@ubuntu: ~
student@ubuntu:~$ echo $PATH
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin
student@ubuntu:~$ OLDPATH=$PATH
student@ubuntu:~$ PATH=$PATH:/opt/some_application
student@ubuntu:~$ echo $PATH
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin:/opt/some_application
student@ubuntu:~$ PATH=$OLDPATH
student@ubuntu:~$ echo $PATH
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin
student@ubuntu:~$
```

Another useful path variable is **CDPATH** which is searched when you change directories. For example:

1

2

3

4

5

```
$ cd bin
```

```
-bash: cd: usr: No such file or directory
```

```
$ export CDPATH=/usr:$CDPATH
```

```
$ cd bin
```

```
/usr/bin
```



Any path which begins with / is considered absolute because it specifies the exact filesystem location. Otherwise, it is considered relative and it is implicitly assumed your current directory is prepended.