

# .bashrc

Adapted from materials by Dr. Carrier

A terminal window with a black background. The text 'ubuntu@ubuntu: ~\$' is displayed in a green monospaced font. A vertical cursor bar, composed of multiple colored lines (yellow, orange, red, blue), is positioned to the right of the prompt.

# Configuration files

# Configuration files

~/ .bashrc on your local machine

Runs when you start a terminal

# Configuration files

~/ .bashrc on your local machine

Runs when you start a terminal

~/ .profile on EOS

Runs when you ssh in

On EOS, this also runs your .bashrc

# Configuration files

`~/ .bashrc` on your local machine

Runs when you start a terminal

`~/ .profile` on EOS

Runs when you ssh in

On EOS, this also runs your `.bashrc`

To run updates after editing a file:

- Either re-login
- Or use source (e.g., `source ~/ .bashrc`)

# Configuration files

~/ .bashrc on your local machine

Runs when you start a terminal

~/ .profile on EOS

Runs when you ssh in

On EOS, this also runs your .bashrc

To run updates after editing a file:

- Either re-login
- Or use source (e.g., `source ~/.bashrc`)

These contain bash commands

We'll talk more about similar files soon!

# Aliases

# Aliases

Common example: la

```
alias la='ls -A'
```



# Aliases

Common example: `la`

```
alias la='ls -A'
```

Now, running “`la`” is the same as running “`ls -A`”

# Aliases

Common example: la

```
alias la='ls -A'
```

Now, running “la” is the same as running “ls -A”

Why does this need to be in ~/.bashrc / ~/.profile?

# Aliases

Common example: `la`

```
alias la='ls -A'
```

Now, running “`la`” is the same as running “`ls -A`”

Why does this need to be in `~/.bashrc` / `~/.profile`?

Ideas for other useful aliases?

# Environment variables

Variables that can be used in this shell and its child processes

# Environment variables

Variables that can be used in this shell and its child processes

How to view our environment variables?

# Environment variables

Variables that can be used in this shell and its child processes

How to view our environment variables?

```
printenv
```

# Environment variables

Variables that can be used in this shell and its child processes

How to view our environment variables?

```
printenv
```

What are they used for?

# Environment variables

Variables that can be used in this shell and its child processes

How to view our environment variables?

```
printenv
```

What are they used for?

Example: PATH - Where bash looks for programs to run



# Editing environment variables (e.g., PATH)

Best to do this in `~/.bashrc`

# Editing environment variables (e.g., PATH)

Best to do this in ~/.bashrc

Add a new line:

```
export PATH="newpath:$PATH"
```

# Editing environment variables (e.g., PATH)

Best to do this in ~/.bashrc

Add a new line:

```
export PATH="newpath:$PATH"
```

“:” allows us to have multiple values

Note that bash searches PATH in order

# Installing software from source

# Installing software from source

Why might we want to do this?

# Installing software from source

Why might we want to do this?

Working through an example: sl

# Installing software from source

Why might we want to do this?

Working through an example: sl

First, let's clone the repo:

```
git clone git@github.com:mtoyoda/sl.git
```

Build the application, if necessary

# Installing software from source

Why might we want to do this?

Working through an example: sl

First, let's clone the repo:

```
git clone git@github.com:mtoyoda/sl.git
```

Build the application, if necessary

Modify PATH to include this directory