Today's Goal

To understand how executable scripts work using the $PATH variable and explore some options for creating your own.

Agenda:

1. Review
2. Understanding the Environment
3. Workshop

Redirect Operators

- Overwrite Operator
- Append Operator
- Input Operator
- Pipe Operator

Redirect Operators

- Overwrite Operator: >
- Append Operator: >>
- Input Operator: <
- Pipe Operator: |

Commands That Use Redirects

- wc
- sort
- uniq
- grep
- sed

Editing Our Settings

- Which text editors have we seen so far?
- What two files configure BASH?
- Which command reloads our settings?
- Which command creates a variable?

Editing Our Settings

- Which text editors have we seen so far? nano, vim, TextEdit
- What two files configure BASH? .bashrc, .bash_profile
- Which command reloads our settings? source
- Which command creates a variable? export

Common Environment Variables

- $USER
- $HOME
- $PS1
- $PATH

Use echo $VARIABLE to print the values for each variable.

So what is the Environment?

The environment is the context that your programs are running in.

```
+----------------------------------------------------------+
|                      Computer                            |
|  +----------------------------------------------------+  |
|  |                                                    |  |
|  |                     BASH                           |  |
|  |                                                    |  |
|  +----------------------------------------------------+  |
|                                                          |
+----------------------------------------------------------+
```

Where did this command come from?

- BASH looks through each directory in the $PATH
- When it finds the program, then it can run it

```
$ which nano
/usr/bin/nano
```

Let's look at the PATH

```
$ echo $PATH
```

Is /usr/bin/ in the list of directories?

What else is in there?

```
$ ls /usr/bin/
```

Recognize anything?

Time to get dangerous!

We're going to break our $PATH

Step 1: Make a new script

```
$ mkdir ~/scripts
$ cd ~/scripts
$ touch rm
$ ls -l
total 0
-rw-r--r--  1 zach  staff  0 Aug 27 12:45 rm
```

Step 2: Make it executable

```
$ chmod +x rm
$ ls -l
total 0
-rwxr-xr-x  1 zach  staff  0 Aug 27 12:45 rm
```

Step 3: Add lines to the script

```
$ nano rm
```

And add these lines to it:

```bash
#!/bin/bash

echo "Hello World"
```

Step 3b: Test the script

```
$ ./rm
Hello World
```

Step 4: Add a directory to the $PATH

```
$ nano ~/.bash_profile
```

At the bottom, add this line:

```
export PATH="$HOME/scripts:$PATH"
```

Step 4b: Update your settings

```
$ source ~/.bash_profile
```

Now try to delete your new script

```
$ rm rm
Hello World
```

Workshop

Your chance to explore something new or practice what we've learned.

Workshop

- [Learn Enough Command Line to Be Dangerous](#)
- [101 Bash Commands and Tips for Beginners to Experts](#)
- [CLI Challenges](#)
- [Codecademy: Learn Bash Scripting](#)
- [TLDP: Bash Programming Intro](#)

The End!