

Shell Aliases: Making Complicated Incantations Simple

What is an Alias?

Maybe you keep your EECS338 code in `~/338/HW/`, and after logging in, you're tired of constantly typing `cd ~/338/HW`. Here is where an alias would be helpful. An alias is similar to a macro: it's a shortcut, an abbreviation.

Creating a Basic Alias

To solve your annoyance of constantly typing `cd ~/338/HW`, you can create an alias. Maybe you'd rather just type `hw`, because that's shorter and easier than `cd ~/338/HW`. To set that up, you'd run the following command:

```
alias hw='cd ~/338/HW'
```

to verify this works, move somewhere that's not `~/338/HW`, and run `hw`. You can verify you're in the proper directory with `pwd`, or Print Working Directory.

Saving your Alias

Unfortunately, executing the `alias` command isn't permanent: once you exit the shell and re-enter it, the `hw` command will no longer work. To save this, you need some way of re-executing the program every time you log into the shell. The common method of doing this is a dotfile.

A dotfile is a digital backpack: your personal set of tools and preferences for a program. Bash, the default shell for most servers, keeps the dotfile `~/.bashrc`. Whenever the bash shell starts, the contents of this file are executed before you're given the terminal. By putting the alias command in this file, your `hw` alias will be reinitialized everytime you log in to the server.

To append your alias to your `.bashrc` file, run the following command:

```
echo "alias hw='cd ~/338/HW'" >> ~/.bashrc
```

This command places the text `alias hw='cd ~/338/HW'` on a new line at the end of the `~/.bashrc` file. You can verify that everything worked by exiting and re-entering the shell, and running your `hw` alias.