

Simplify linux find command using shell functions

Do you need to find a file and then perform some action on it and get caught up in curly brackets, back slashes and syntax errors when you could swear “this command worked in the past?”. It’s one of the joys of Linux I guess, but quickly becomes tedious when you’re working against a problem and are under stress.

Here is a reference *find* command that works. I hope it helps. It’ll no doubt help me at some point (*the entire purpose of my blog is to actually remind myself how to do half of this stuff from time to time*).

```
sudo find ./ -name *.mkv -exec ls {} \;
```

Something I like to do is create shell functions in the `.bashrc` file in your home directory to simplify commonly used commands that are long to type and quite syntax sensitive.

```
#SHELL FUNCTIONS FOR .bashrc
```

```
f() { find . -name "$1"; }
```

This is a nice useful one that can be used to find any files that have the specified string anywhere in the filename. Just type `f All` to find any files with the word All occurring anywhere in the filename.

You could create other versions such as this one, that will find and remove files with a specified string in the filename – but I’d really not recommend it.

```
fr() { find ./ -name "$1" -exec rm {} \; }
```

Be sure to run `man fr` first to check that your shell function name isn’t the name of an existing binary on the system!