md-pdf-converter

Uses the 'markdown-pdf' 3rd party node module to convert md files to a pdf.

markdown-pdf CLI interface usage:

```
Usage: markdown-pdf [options] <markdown-file-path>
```

I want to extend this module and create a module that converts all md files within a specific directory into pdf files (if one already exists, then overwrite).

If possible I would have a watch method that executes when a md file is added or changed.

When syncing my mobile with a specific folder, I can have all of my recent study notes writen in markdown readily available on my mobile to read over with use of dropbox.

Making your new scripts executable

Rather than having to type...

`node path/to/script.js arguments`

adjusting the script's file path all the time relative to our current location we could just run `script` in our terminal anywhere in our directory tree and know it will work just like unix `cat`, `ls`, and `grep`?

You can do that in node and it only takes a few simple steps!

1. Add this to the top line of the file:

```
#!/usr/bin/env node
```

2. Run `npm init` in the root folder of the repo and create a `package.json`.

3. Adjust your `package.json` as follows:

- remove the main entry: this is only used for modules that will be used through the module system (e.g. var _ = require('underscore');).
- add preferGlobal and set it to true, which means if someone installs this module through npm and doesn't use the --global option, they will be warned that the module is designed to be installed globally.
- add the bin object, which maps commands to files. This means when this module is installed, npm will set up the named executables to execute their assigned javascript files. Don't name them after existing commands like ls, grep, etc, it needs to be unique.

Your `package.json` should now look like this:

```
{
  "name": "node-shell-workshop",
  "version": "1.0.0",
  "description": "learn how to shell script with node",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "author": "your lovely selves",
  "license": "ISC",
  "preferGlobal": true,
  "bin": {
    "your-name-here-cat": "lesson-1/cat.js",
    "your-name-here-ls": "lesson-1/ls.js",
    "your-name-here-grep": "lesson-1/grep.js"
  }
}
```

4. Now in the same repo you can run `npm link` to install the script on your system. This creates a symlink to your project so that you can run the project whilst working on it, with no need to keep reinstalling it over and over again.

5. Now... move into a different directory and try out...

```
your-name-here-cat file.extension
your-name-here-ls
your-name-here-ls -ex extension
your-name-here-grep pattern file.extension
```