https://www.electronjs.org/docs/latest/tutorial/quick-start

(1) Install NodeJS

- Windows: https://nodejs.org/en/download/
- Linux (Arch): `sudo pacman -S npm`

(2) Create a directory for your new project and cd into it (i.e. `mkdir firstProject; cd firstProject`)

(3) Use `npm init` to create a package.json

- Make sure that the entry point ("main" in package.json) is set to main.js (or js/main.js)

```
{
  "name": "firstproject",
  "version": "1.0.0",
  "description": "",
  "main": "main.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "author": "",
  "license": "ISC",
  "devDependencies": {
    "electron": "^19.0.3"
  }
}
```

You should get something like this

(4) Install Electron in your project's directory

1. Of course, make sure you're within your project's directory
2. Installing Electron locally to your project's directory
    a. Windows: `npm i electron --save-dev`
    b. Linux: `npm i electron --unsafe-perms=true --save-dev`
    c. Pro: More version control with Electron with multiple projects
    d. Con: Not installed globally, so it's a tad more typing
3. Installing Electron globally on your machine
    a. `npm i electron -g`
    b. Pro: Only have to install it once and you have electron for any project
    c. Con: Static version number, less version control

(4) Add a start script

```
{
  "name": "firstproject",
  "version": "1.0.0",
  "description": "",
  "main": "main.js",
  "scripts": {
    "start": "electron ."
  },
  "author": "",
  "license": "ISC",
  "devDependencies": {
    "electron": "^19.0.3"
  }
}
```

Removed "test" in "scripts" and added "start": "electron ."
We may now call npm start to start our Electron program

(5) Create index.html

```
<!DOCTYPE html>

<html>
  <head>
    <style> body { background: #101010; } </style>
  </head>
  <body>
  </body>
</html>
```

(6) Create main.js

Given that our entry point we set in `npm init` is main.js (or js/main.js), we must create this entry point

```
const { app, BrowserWindow } = require("electron")

// Function to create our window
const createWindow = () => {
  const win = new BrowserWindow({ width: 800, height: 600 })
  win.loadFile("index.html")
  win.setMenu(null) // Electron by default has a menubar, this is how we remove it
}

// Function that fires createWindow() when Electron has initialized
app.whenReady().then(createWindow)
```

`const { app, BrowserWindow } = require("electron")`

- The app module controls the entire application itself (i.e. quitting the application)
- The BrowserWindow module controls and creates windows (also called renders)

`const createWindow = () = { … }`

- We create our main window, win using the BrowserWindow constructor
- In order to load index.html, we use `win.loadFile("index.html")`
- By default, Electron adds a native menubar, we can remove it with `win.setMenu(null)`

`app.whenReady().then(createWindow)`

- Once Electron initializes, we call createWindow

(7) Run with `npm start`