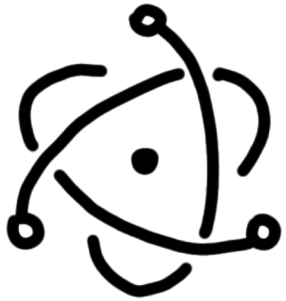


# ELECTRON WORKSHOP

DAVID NEAL @REVERENTGEEK

# Up Ahead

- Intro to Electron
- Create an App
- Live Reload
- Debug
- IPC
- Package
- Tests

# Set up and initialize your project

https://bit.ly/electron-workshop

- or –

https://github.com/reverentgeek/electron-workshop/wiki


After completing Exercise 1, Part 1

# .NET + Mono + Xamarin

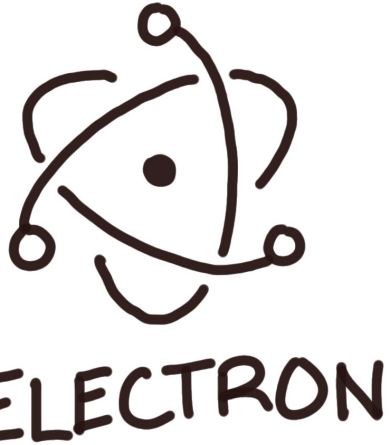

Pros:

- Shared .NET code base

Cons:

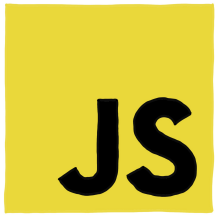
- Xamarin Mac !== Xamarin iOS/Android
- Native UI is hard
- Deployment
- ~~Licensing~~

# Electron

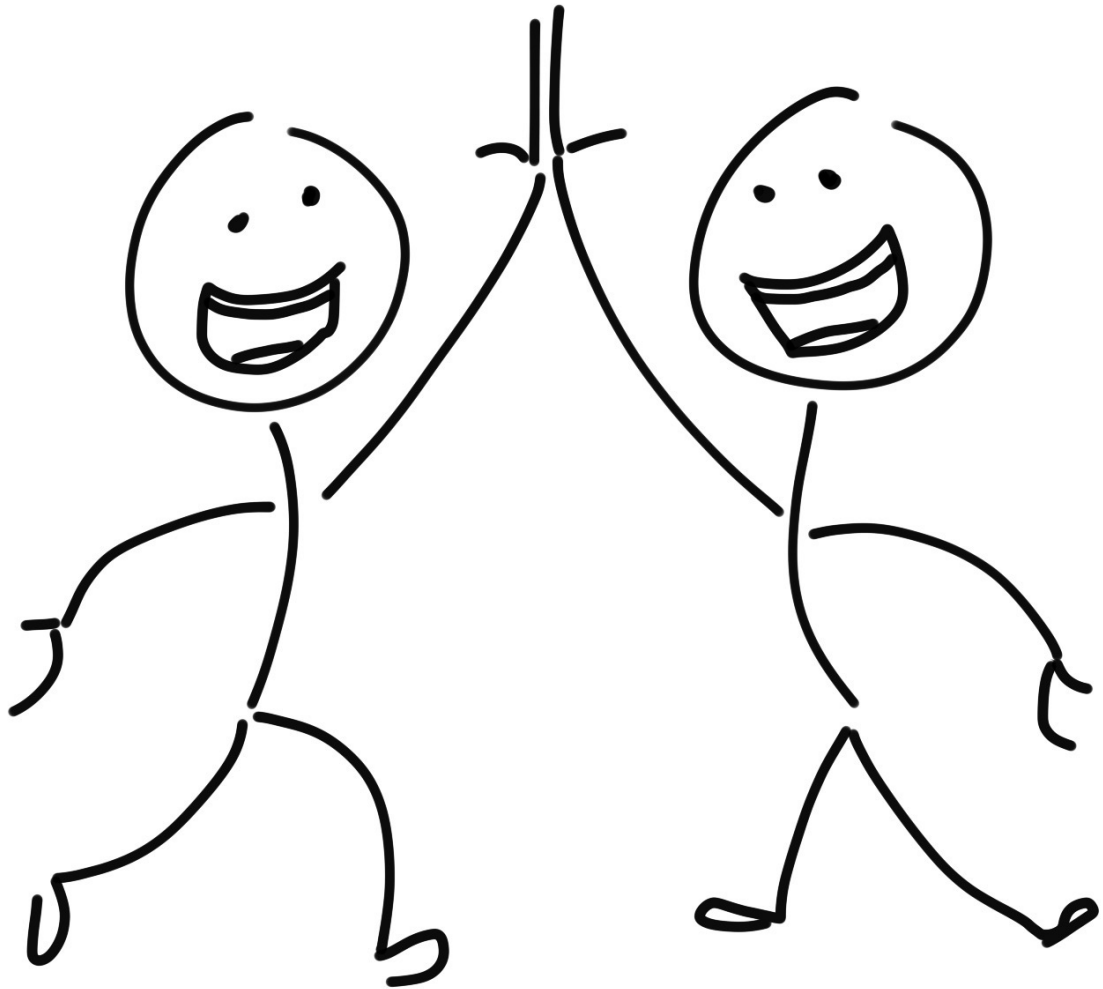



Pros:

- HTML + CSS + JavaScript
- Node.js + Chromium
- No deployment dependencies







DAVID NEAL @REVERENTGEEK

# Not Your Dad's

# JavaScript

David Neal @reverentgeek

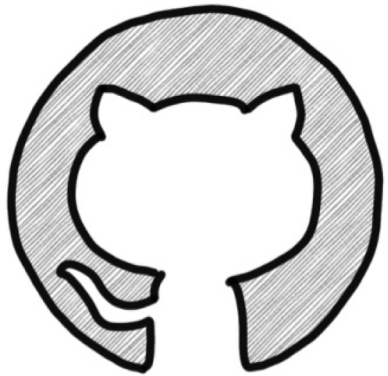
# Electron

Cons:

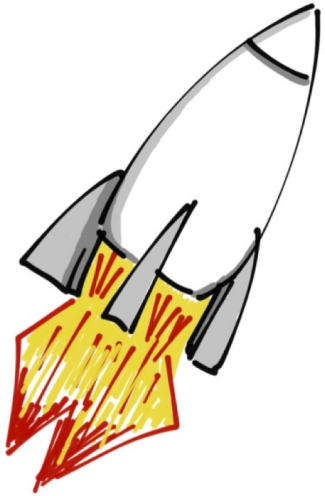
- HTML + CSS + JavaScript

# History

- Created by GitHub for Atom
- Formerly Atom Shell
- Active since January 2013
- 1.0 release May 2016
- 2.0 release May 2018



# Electron Features



- Rapid development
- Themes
- Shared code/UI
- Deployment + “silent” updates
- Native UX

# Why Desktop Apps?

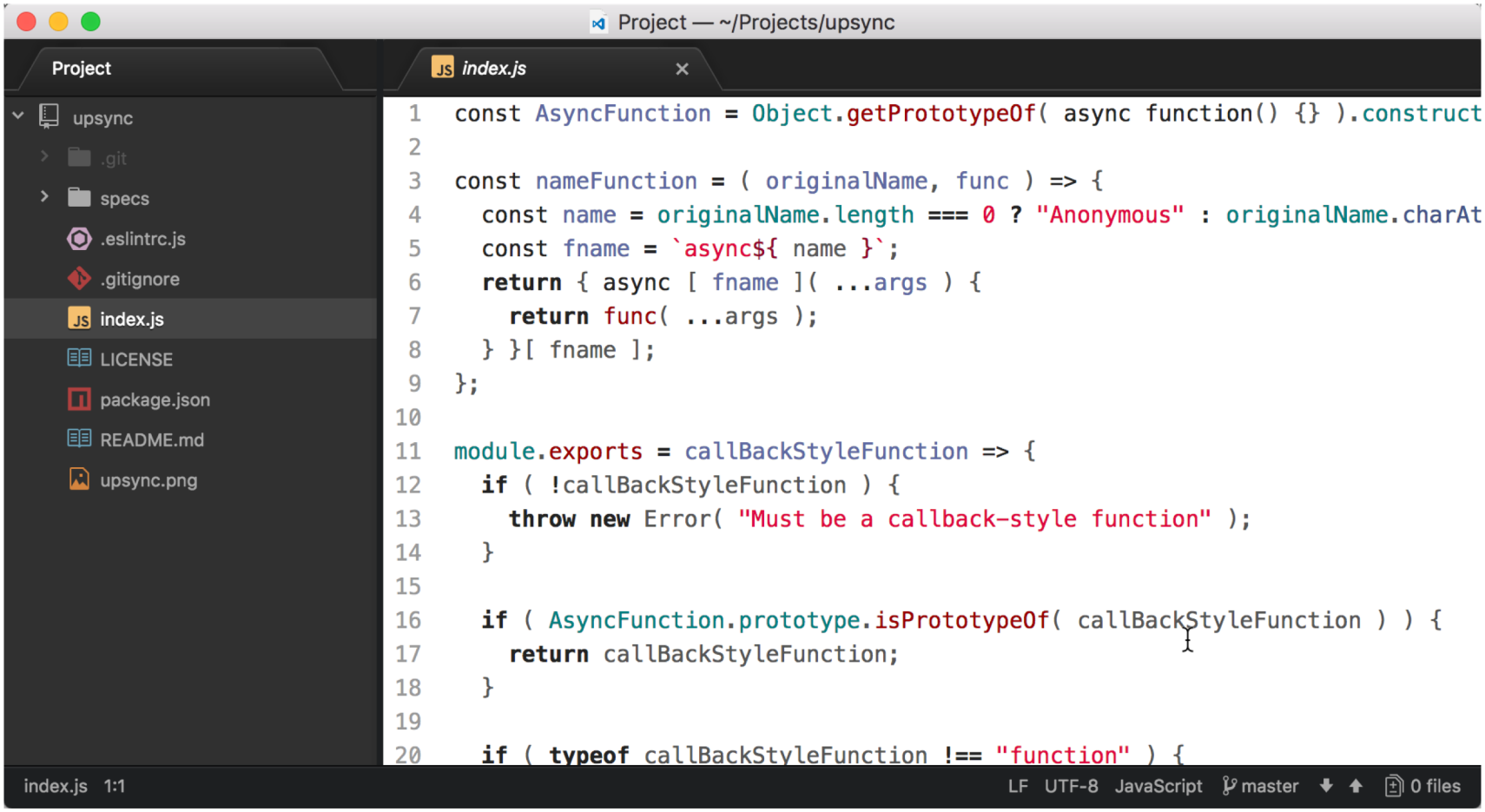

- Offline
- Printers, devices, other local hardware
- On-premises
- Internal, LOB
- Edit local files

- App Store
- Kiosk
- Desktop > Intranet
- Sometimes it “just feels right”

# Desktop App Ideas

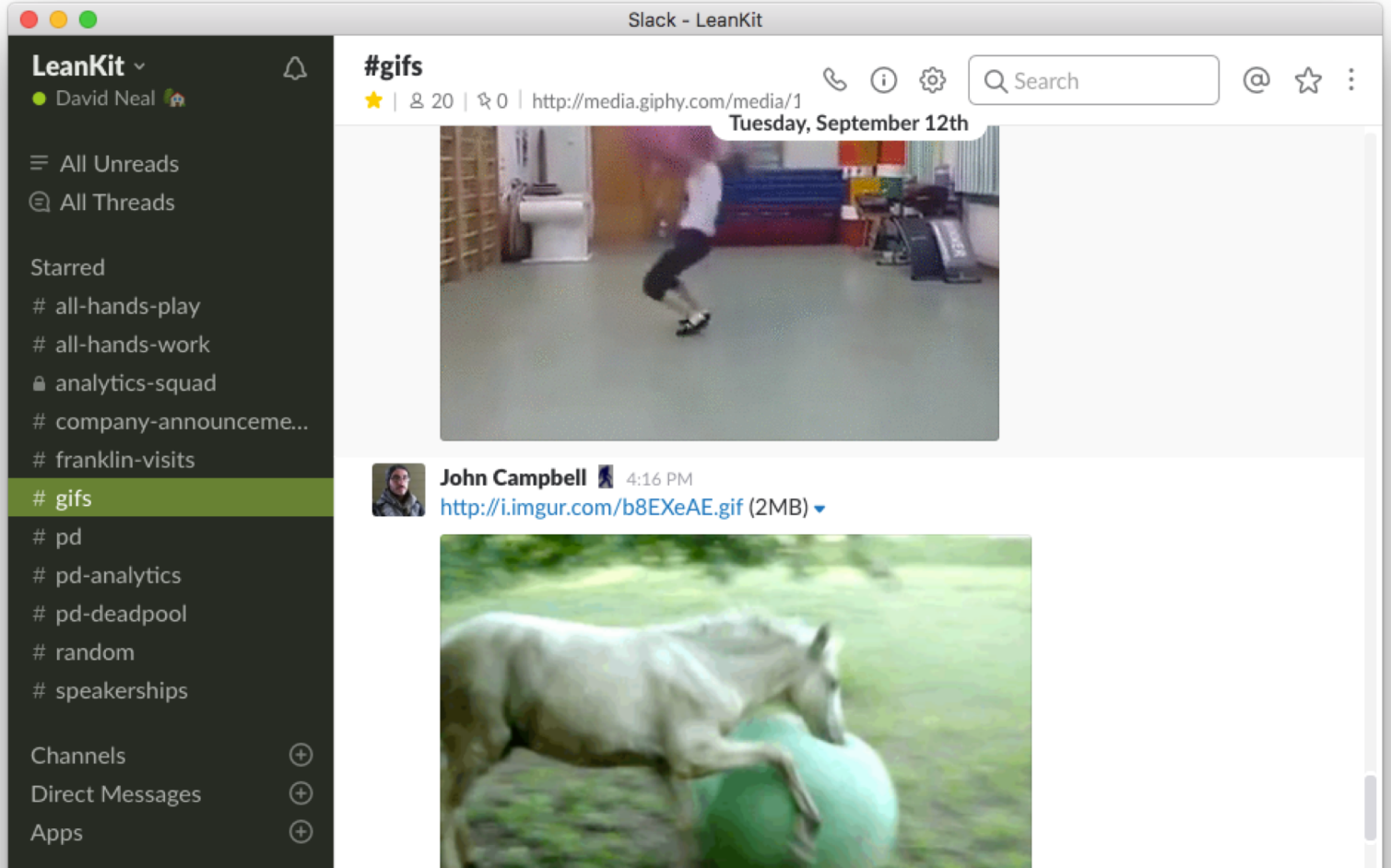
- Disconnected data entry
- Editor
- Time management
- Media player
- Email client
- Messaging, collaboration
- Kiosk
- Mapping, route planner
- Social media client
- Calendar
- Bulk media editor
- File management, backup
- Document generation, reading
- Audio/video conferencing
- Games

# Atom

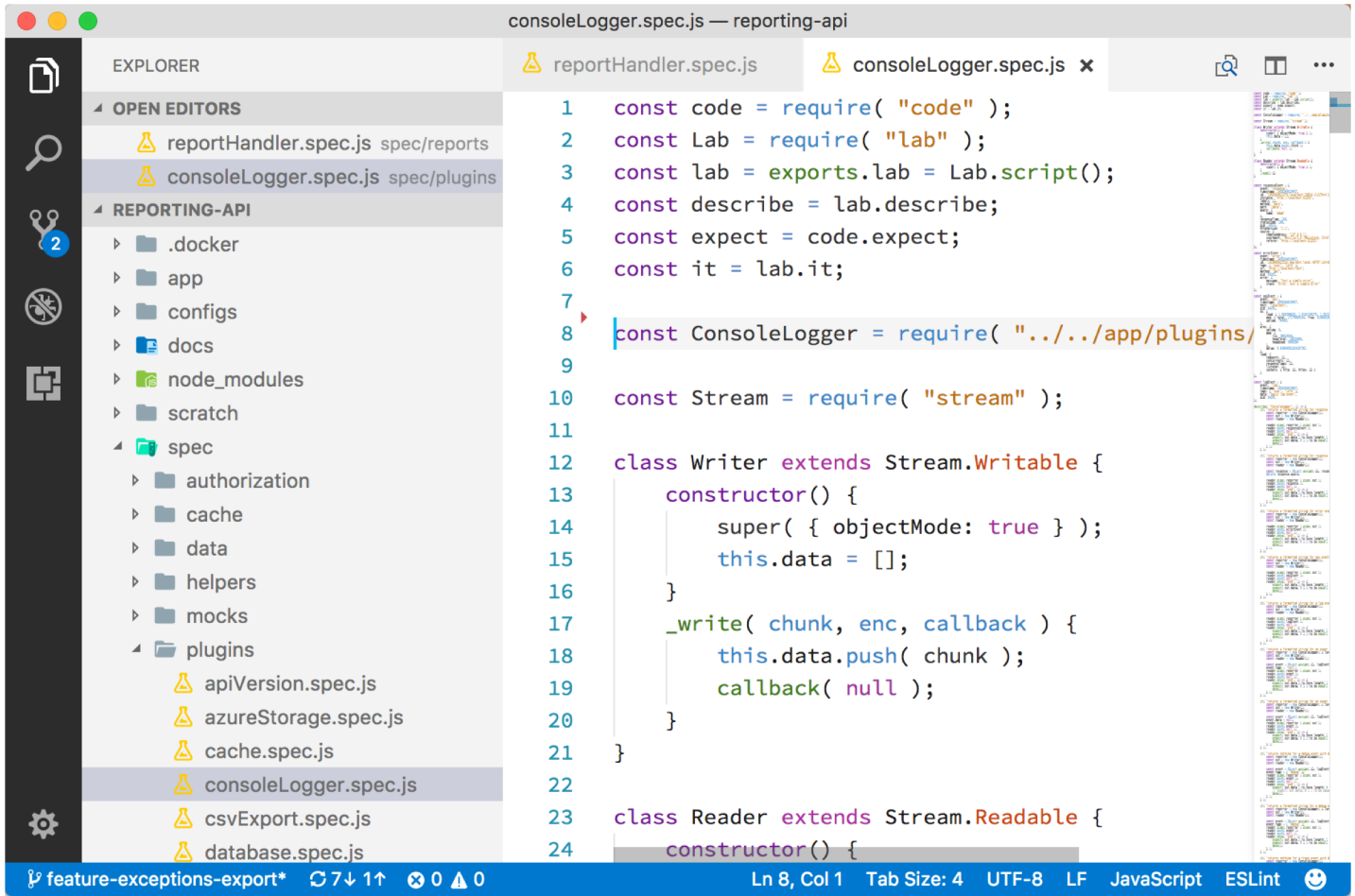



DAVID NEAL @REVERENTGEEK

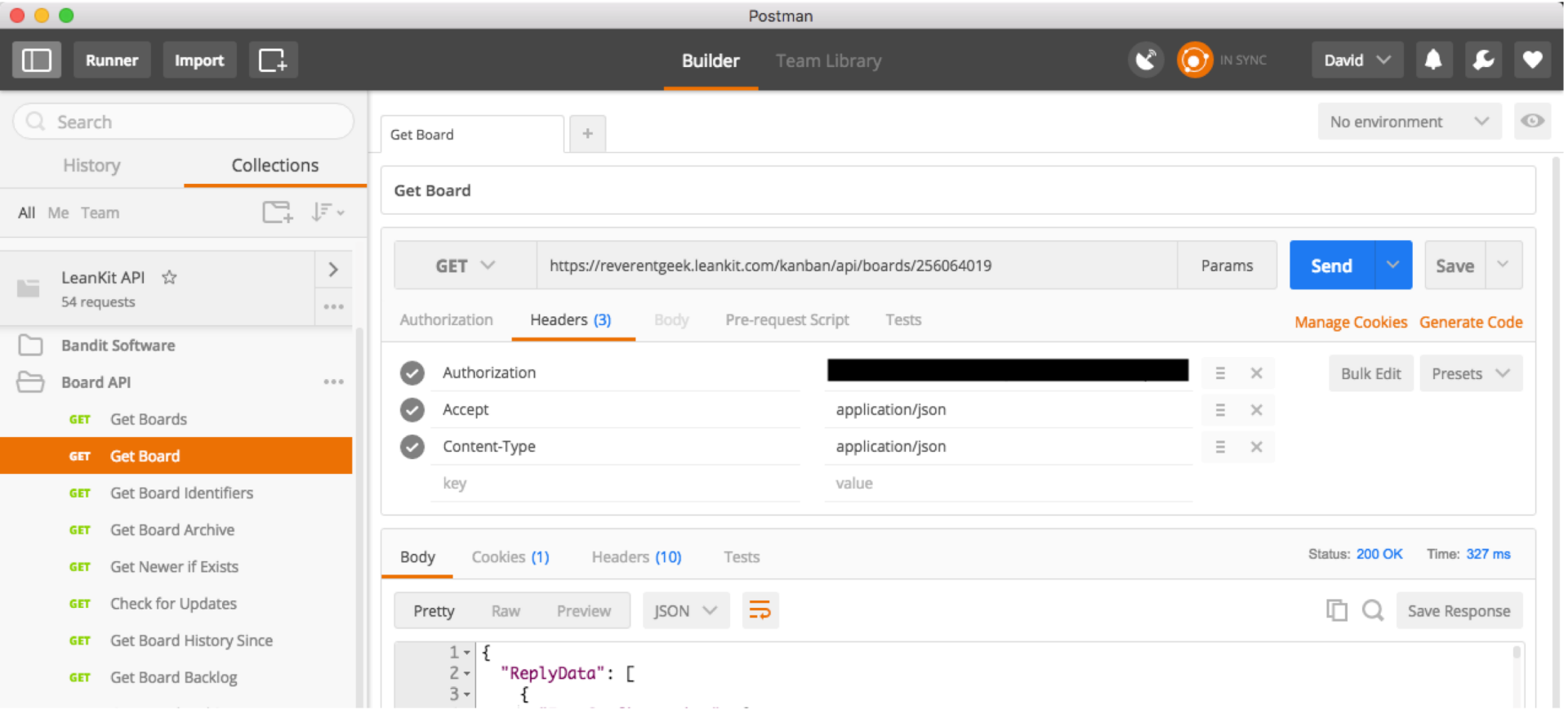
# Slack



# Visual Studio Code

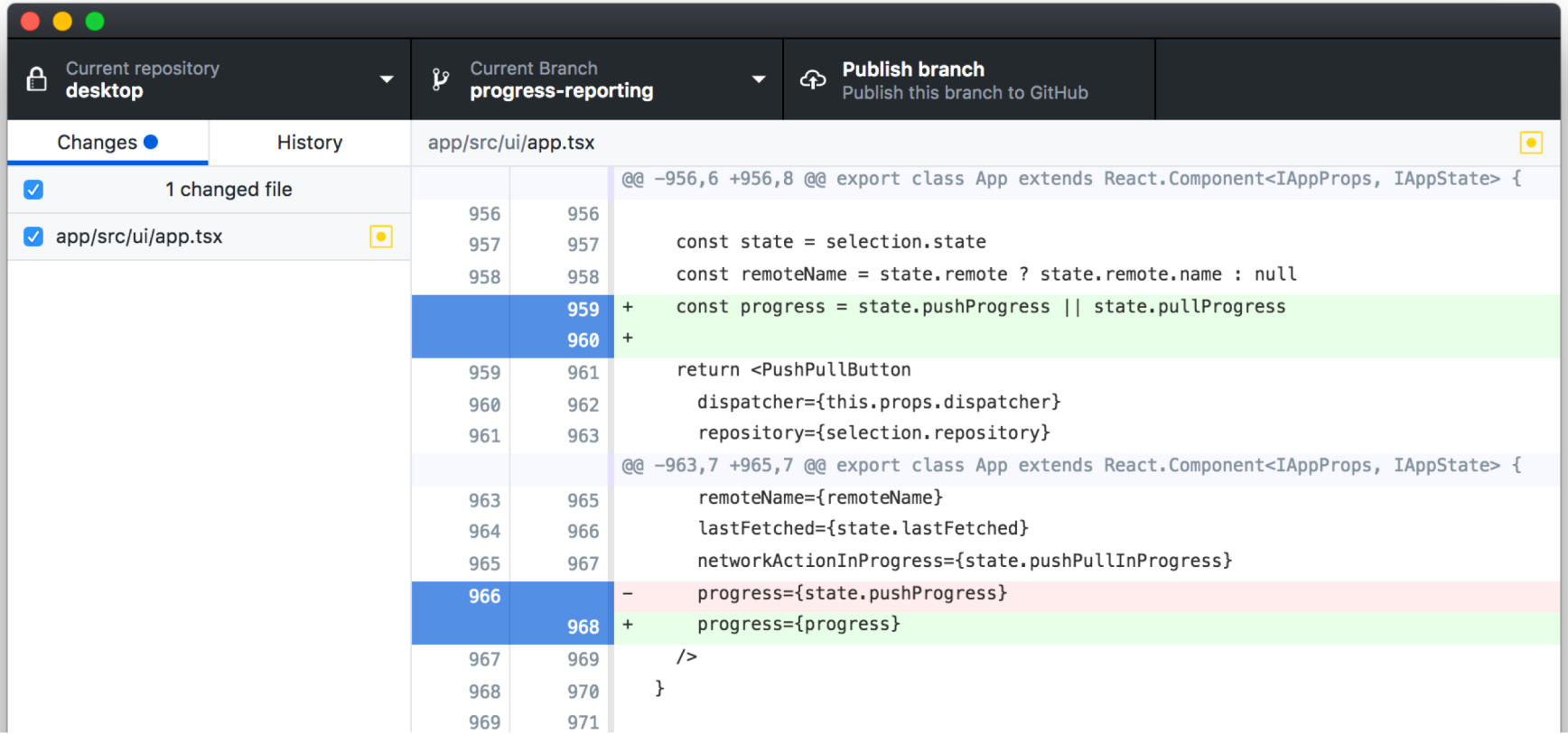


# Postman

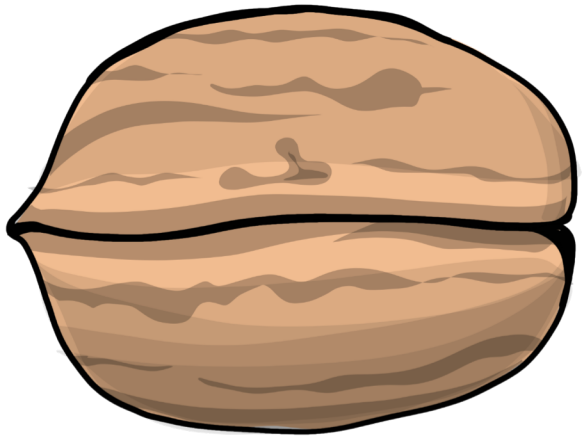



DAVID NEAL @REVERENTGEEK

# GitHub Desktop

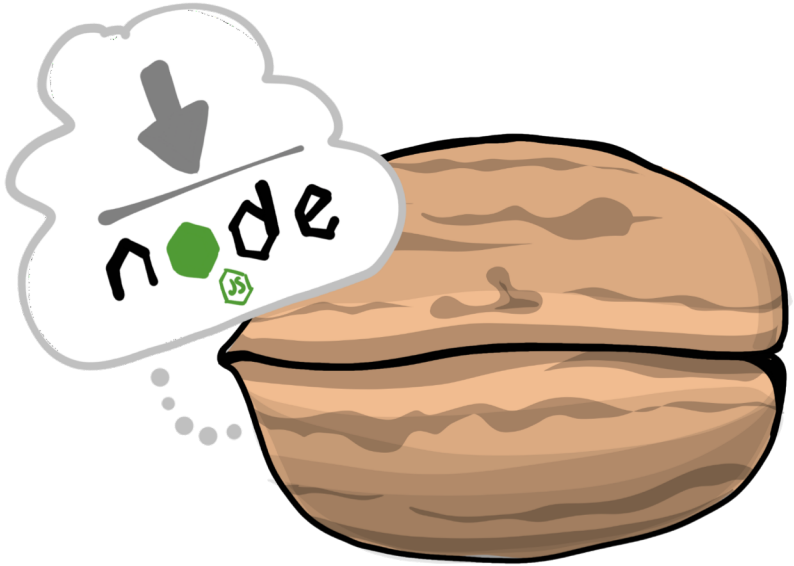


# How to Electron



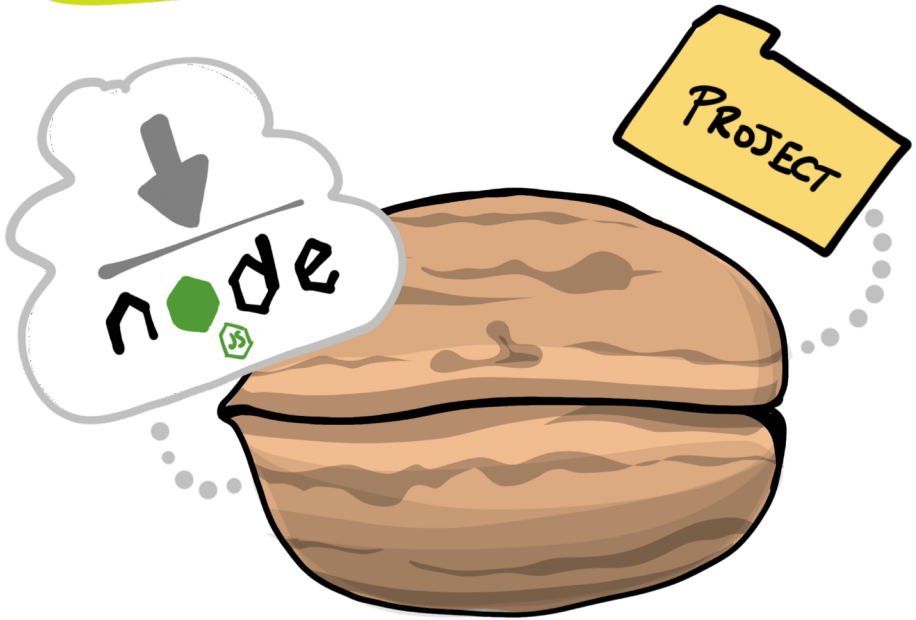
David Neal @reverentgeek

# How to Electron



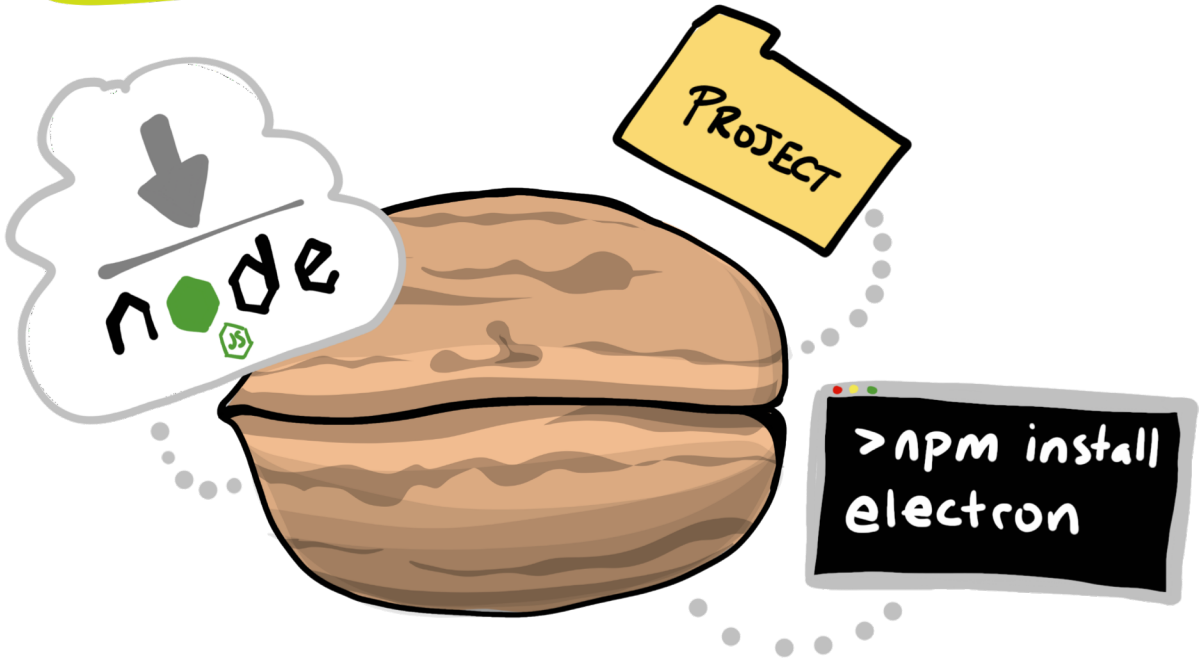

David Neal @reverentgeek

# How to Electron



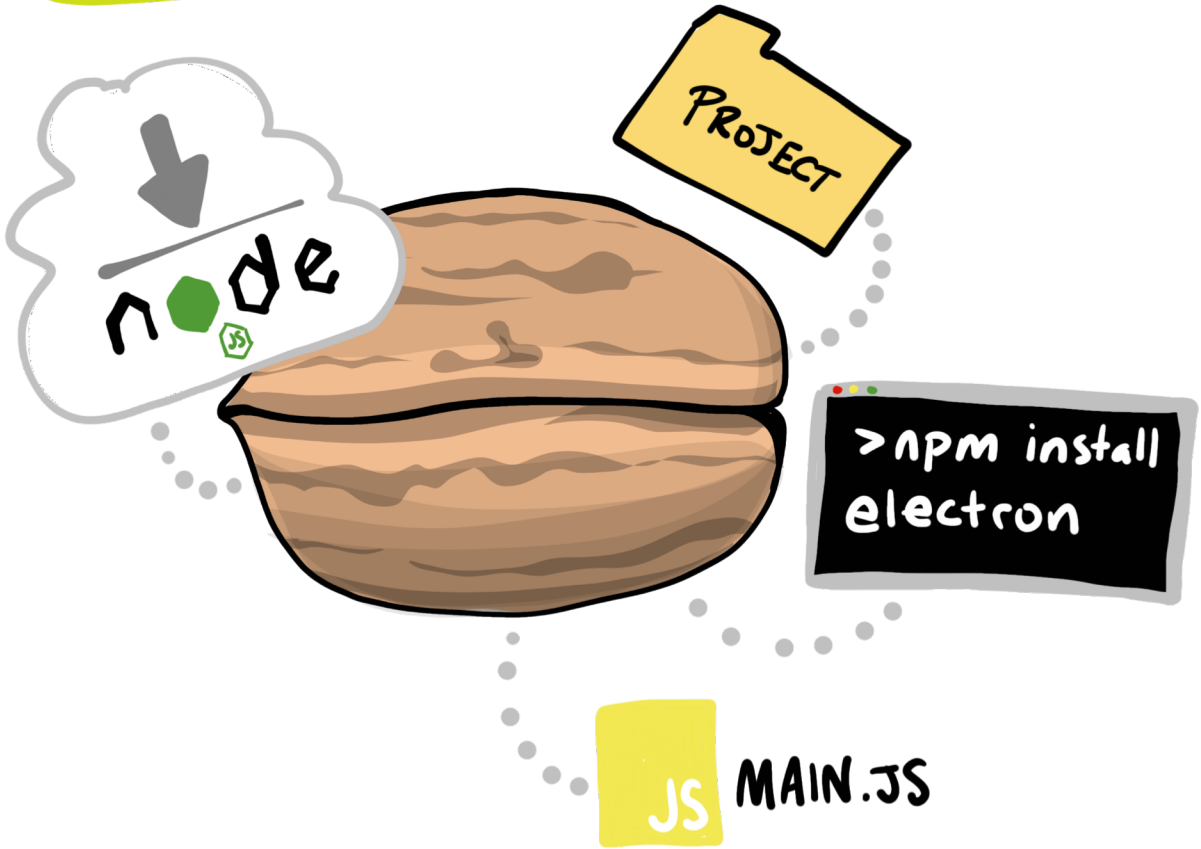

David Neal @reverentgeek

# How to Electron



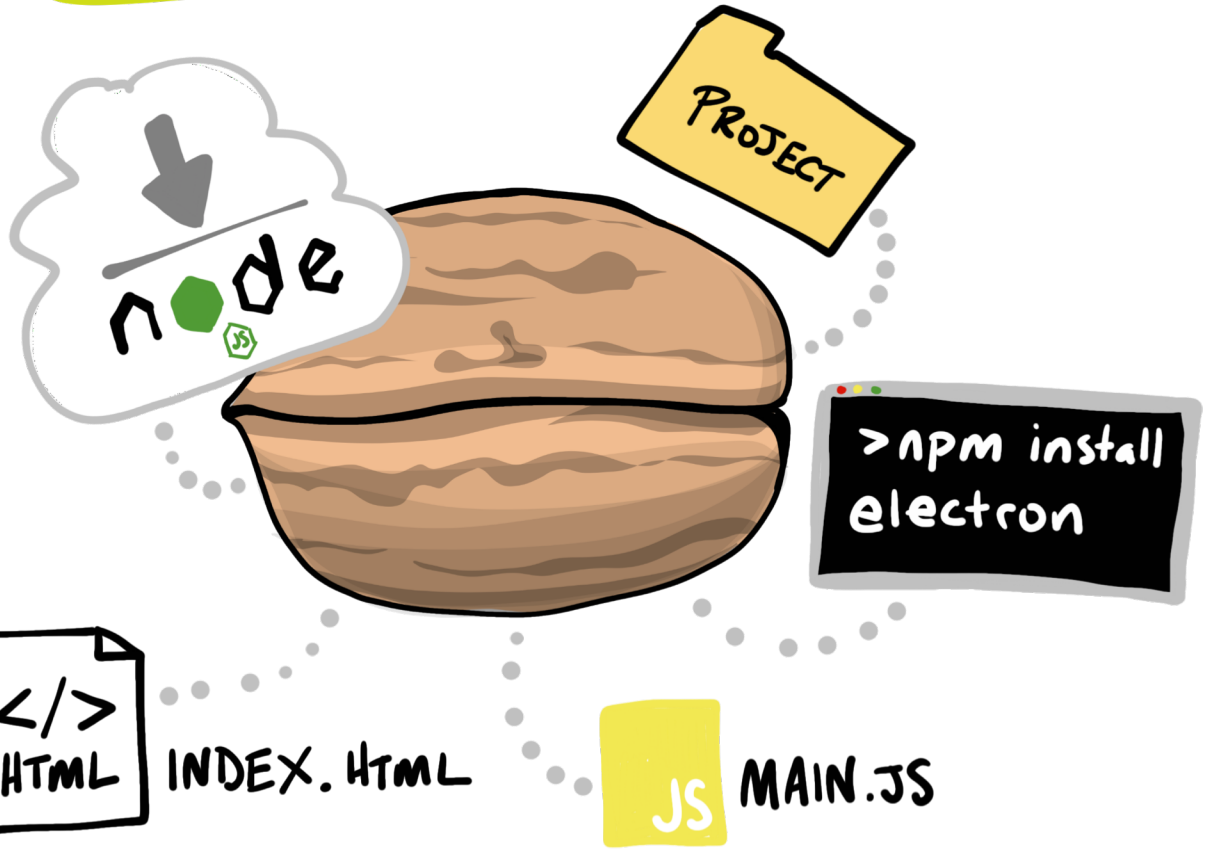

David Neal @reverentgeek

# How to Electron



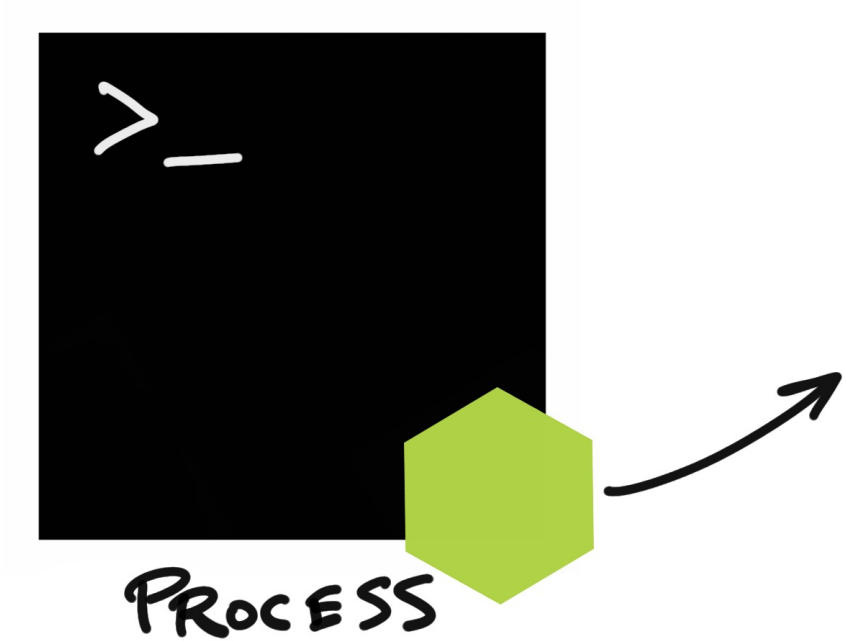

David Neal @reverentgeek

# How to Electron

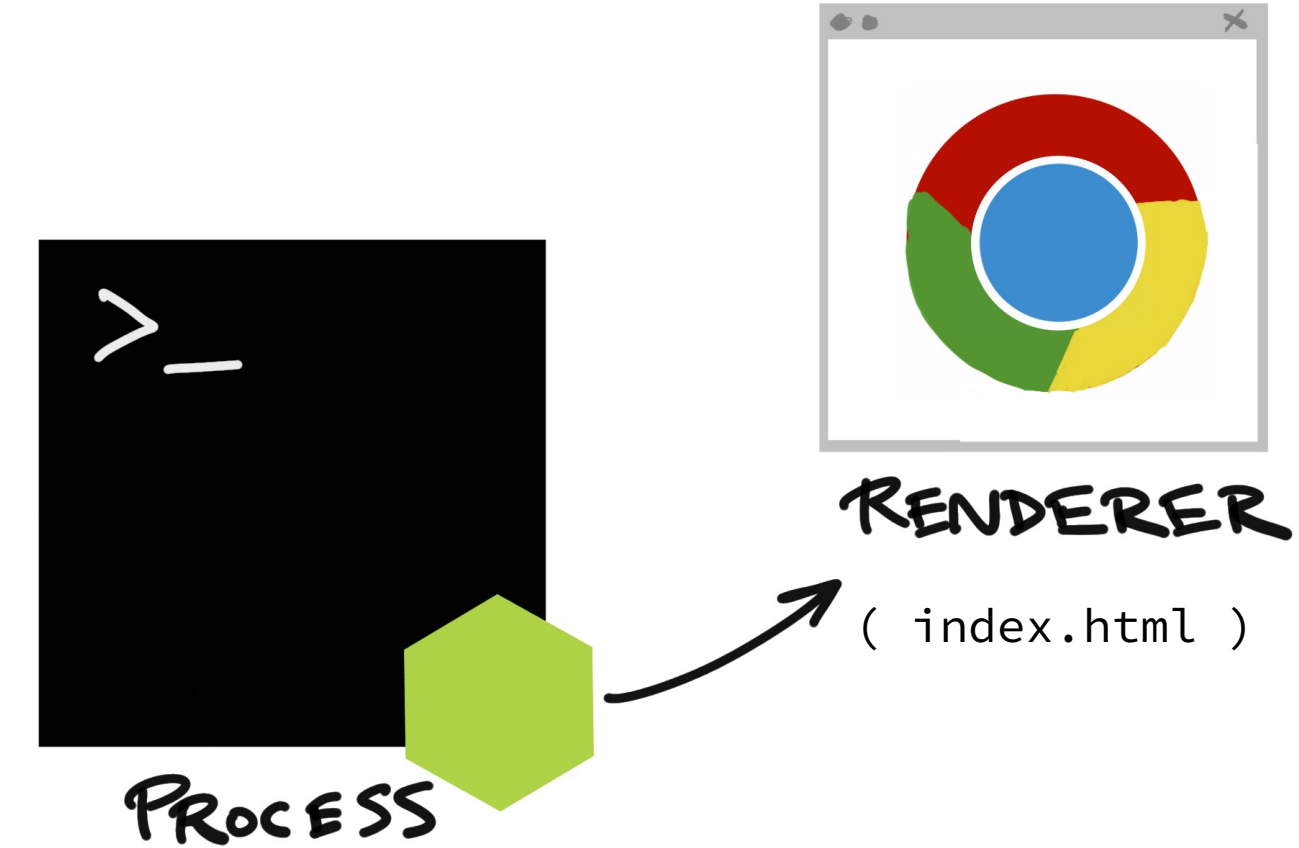


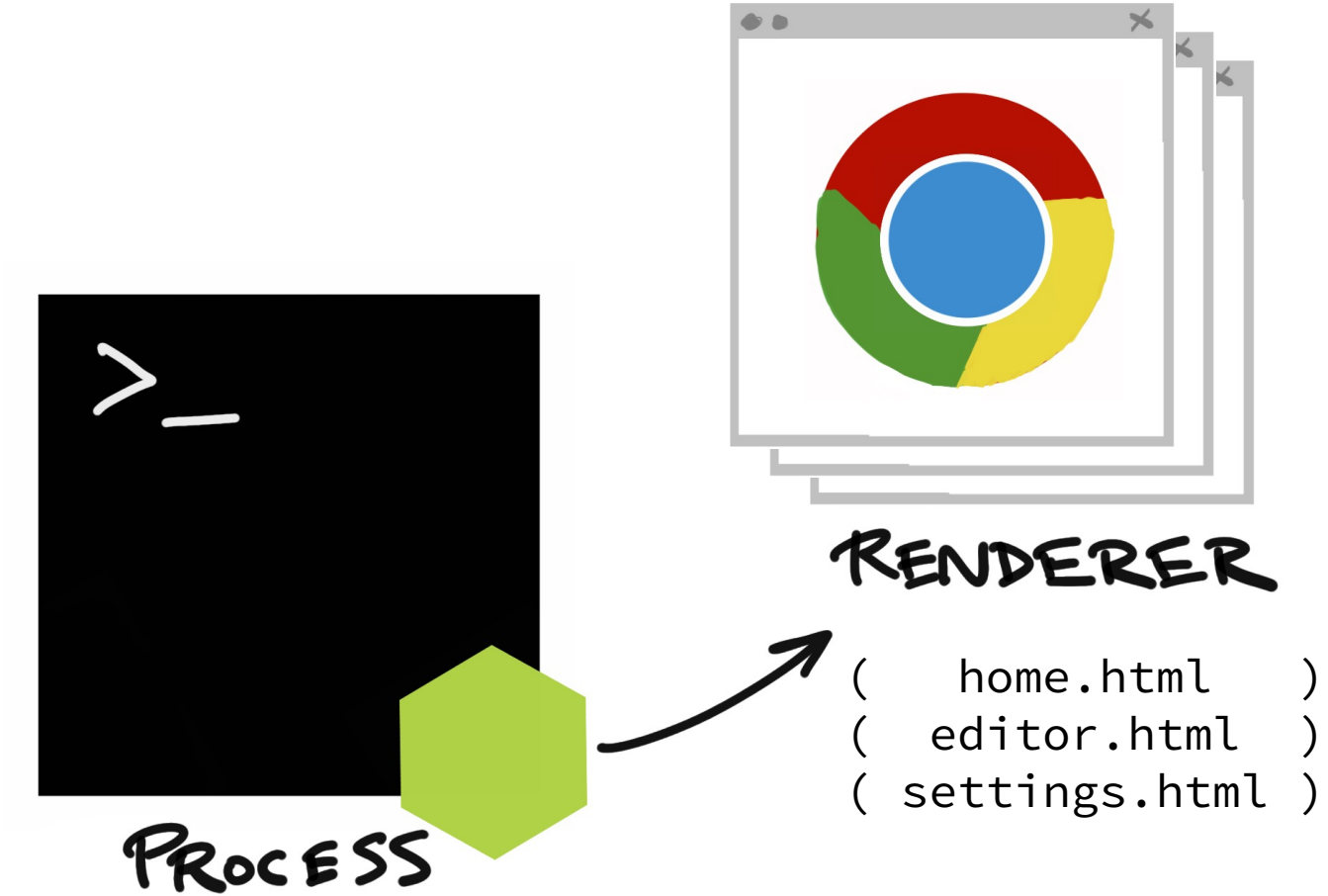

David Neal @reverentgeek


( main.js )




( main.js )


>_
RENDERER
( index.html )
PROCESS
( main.js )


RENDERER
( home.html )
( editor.html )
( settings.html )
PROCESS
( main.js )


DAVID NEAL @REVERENTGEEK

# main.js

```
1   const electron = require( "electron" );
2   const app = electron.app;
3   const BrowserWindow = electron.BrowserWindow;
4   let mainWindow;
5
6   app.on( "ready", () => {
7       mainWindow = new BrowserWindow( {
8           width: 400,
9           height: 300
10      } );
11      mainWindow.loadURL( `file://${ __dirname }/index.html` );
12  } );
```

# main.js

```
1   const electron = require( "electron" );
2   const app = electron.app;
3   const BrowserWindow = electron.BrowserWindow;
4   let mainWindow;
5
6   app.on( "ready", () => {
7       mainWindow = new BrowserWindow( {
8           width: 400,
9           height: 300
10      } );
11      mainWindow.loadURL( `file://${ __dirname }/index.html` );
12  } );
```

# main.js

```
1   const electron = require( "electron" );
2   const app = electron.app;
3   const BrowserWindow = electron.BrowserWindow;
4   let mainWindow;
5
6   app.on( "ready", () => {
7       mainWindow = new BrowserWindow( {
8           width: 400,
9           height: 300
10      } );
11      mainWindow.loadURL( `file://${ __dirname }/index.html` );
12  } );
```

# main.js

```
1   const electron = require( "electron" );
2   const app = electron.app;
3   const BrowserWindow = electron.BrowserWindow;
4   let mainWindow;
5
6   app.on( "ready", () => {
7       mainWindow = new BrowserWindow( {
8           width: 400,
9           height: 300
10      } );
11      mainWindow.loadURL( `file://${ __dirname }/index.html` );
12  } );
```

# main.js

```
1   const electron = require( "electron" );
2   const app = electron.app;
3   const BrowserWindow = electron.BrowserWindow;
4   let mainWindow;
5
6   app.on( "ready", () => {
7       mainWindow = new BrowserWindow( {
8           width: 400,
9           height: 300
10      } );
11      mainWindow.loadURL( `file://${ __dirname }/index.html` );
12  } );
```

# main.js

```
1   const electron = require( "electron" );
2   const app = electron.app;
3   const BrowserWindow = electron.BrowserWindow;
4   let mainWindow;
5
6   app.on( "ready", () => {
7       mainWindow = new BrowserWindow( {
8           width: 400,
9           height: 300
10      } );
11      mainWindow.loadURL( `file://${ __dirname }/index.html` );
12  } );
```

# main.js

```
1   const electron = require( "electron" );
2   const app = electron.app;
3   const BrowserWindow = electron.BrowserWindow;
4   let mainWindow;
5
6   app.on( "ready", () => {
7       mainWindow = new BrowserWindow( {
8           width: 400, // x, y, minWidth, minHeight, movable,
9           height: 300 // resizable, alwaysOnTop, kiosk
10      } );
11      mainWindow.loadURL( `file://${ __dirname }/index.html` );
12  } );
```

# main.js

```
1   const electron = require( "electron" );
2   const app = electron.app;
3   const BrowserWindow = electron.BrowserWindow;
4   let mainWindow;
5
6   app.on( "ready", () => {
7       mainWindow = new BrowserWindow( {
8           width: 400,
9           height: 300
10      } );
11      mainWindow.loadURL( `file://${ __dirname }/index.html` );
12  } );
```

# index.html

```
1   <!DOCTYPE html>
2   <html>
3   <head>
4       <meta charset="UTF-8">
5       <title>Electron Demo</title>
6   </head>
7   <body>
8       <h1>Hello World!</h1>
9   </body>
10  </html>
```

# index.html

```
1   <!DOCTYPE html>
2   <html>
3   <head>
4       <meta charset="UTF-8">
5       <title>Electron Demo</title>
6   </head>
7   <body>
8       <h1>Hello World!</h1>
9   </body>
10  </html>
```

```
> npm start
```

# index.html

```
1   <!DOCTYPE html>
2   <html>
3   <head>
4       <meta charset="UTF-8">
5       <title>Electron Demo</title>
6   </head>
7   <body>
8       <h1>Hello World!</h1>
9   </body>
10  </html>
11
```

# index.html

```
1   <!DOCTYPE html>
2   <html>
3   <head>
4       <meta charset="UTF-8">
5       <title>Electron Demo</title>
6   </head>
7   <body>
8       <h1>Hello World!</h1>
9   </body>
10  </html>
11
```

# index.html

```
1   <!DOCTYPE html>
2   <html>
3   <head>
4       <meta charset="UTF-8">
5       <title>Electron Demo</title>
6   </head>
7   <body>
8       <h1>Hello World!</h1>
9   </body>
10  </html>
11
```

# index.html

```
 7  <body>
 8      <h1>Hello World!</h1>
 9      <div id="info"></div>
10      <script>
11          const msg = `Node.js: ${process.versions.node},
12              Electron: ${ process.versions.electron},
13              Chrome: ${ process.versions.chrome}`;
14          document.getElementById("info").textContent = msg;
15      </script>
16  </body>
```

# index.html

```
 7    <body>
 8        <h1>Hello World!</h1>
 9        <div id="info"></div>
10        <script>
11            const msg = `Node.js: ${process.versions.node},
12                Electron: ${ process.versions.electron},
13                Chrome: ${ process.versions.chrome}`;
14            document.getElementById("info").textContent = msg;
15        </script>
16    </body>
```

# index.html

```
18          const fs = require( "fs" );
19          fs.readdir( process.env.HOME, ( err, entries ) => {
20              entries.forEach( entry => {
21                  if ( !entry.startsWith( "." ) ) {
22                      const li = document.createElement( "li" );
23                      li.textContent = entry;
24                      homeList.appendChild( li );
25                  }
26              } );
27          } );
28      </script>
29  </body>
```

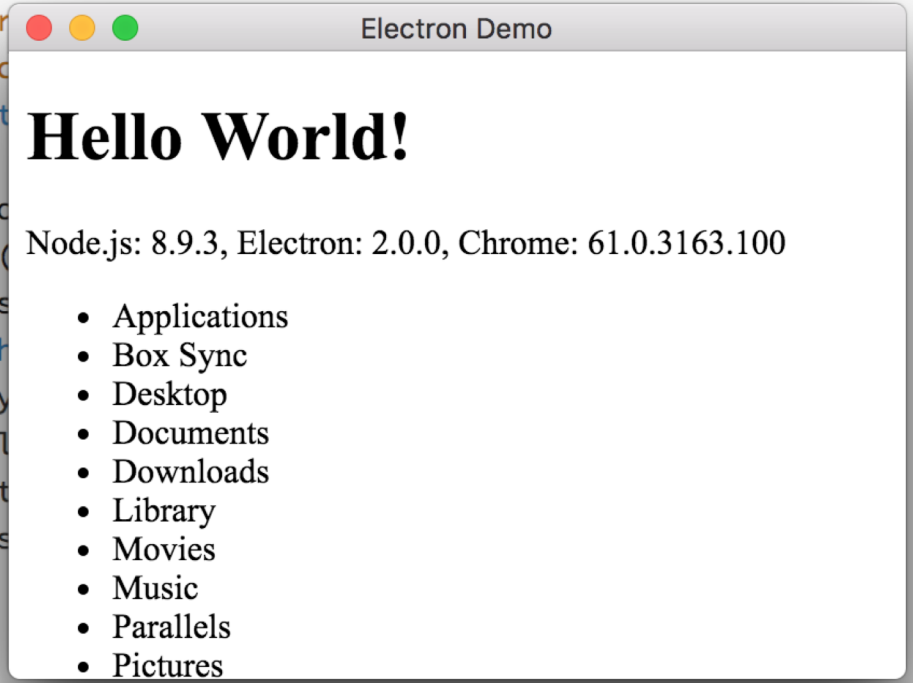
# index.html

```
18          const fs = require( "fs" );
19          fs.readdir( process.env.HOME, ( err, entries ) => {
20              entries.forEach( entry => {
21                  if ( !entry.startsWith( "." ) ) {
22                      const li = document.createElement( "li" );
23                      li.textContent = entry;
24                      homeList.appendChild( li );
25                  }
26              } );
27          } );
28      </script>
29  </body>
```

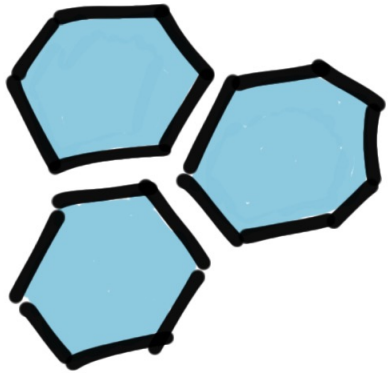
# index.html

```
18            const fs = require( "fs" );
19            fs.readdir( process.env.HOME, ( err, entries ) => {
20                entries.forEach( entry => {
21                    if ( !entry.startsWith( "." ) ) {
22                        const li = document.createElement( "li" );
23                        li.textContent = entry;
24                        homeList.appendChild( li );
25                    }
26                } );
27            } );
28        </script>
29    </body>
```

```
7   <body>
8       <h1>Hello World!</h1>
9       <div id="info"></div>
10      <ul id="homeList"></ul>
11      <script>
12          const msg = `Node.js: ${process.versions.node},
13              Electron: ${ pr
14              Chrome: ${ proc
15          document.getElement
16
17          const homeList = dc
18          const fs = require(
19          fs.readdir( process
20              entries.forEach
21                  if ( !entry
22                      const l
23                      li.text
24                      homeLis
25                  }
26              } );
27          } );
28      </script>
29  </body>
```



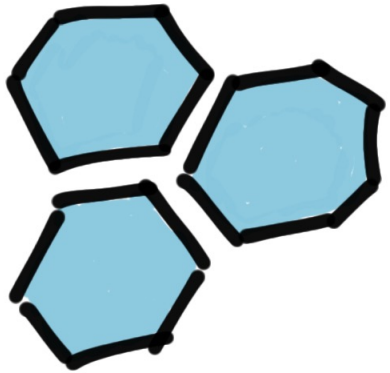
# Process modules



- app
- ipc
- dialog
- menu, menu-item
- power-monitor
- tray

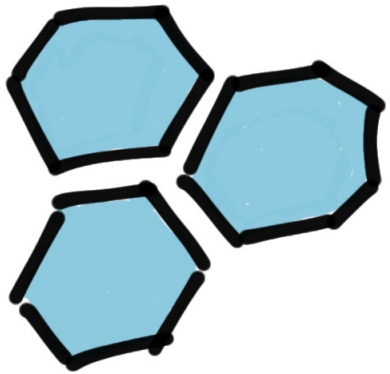
# Render modules

- ipc
- remote
- web-frame

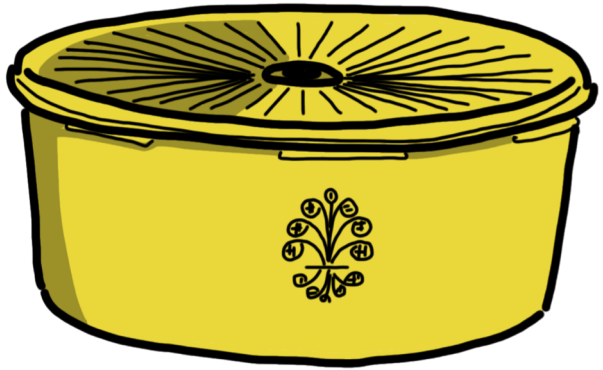


# Modules available to both

- clipboard
- crash-reporter
- native-image
- screen
- shell



# Local Storage



- Read/write .json files
  - Pro tip: use fs-jetpack
- nedb
- pouchdb

# Relational DB Storage

- seriate (SQL Server)
- pg
- mysql
- oracledb

# Let’s Get Started!

https://bit.ly/electron-workshop

- or –

https://github.com/reverentgeek/electron-workshop/wiki


YOU DON'T NEED
PERMISSION
TO BE
AWESOME

# THANK YOU!

DAVID NEAL
@REVERENTGEEK