# Creating an Electron App from an Ionic Angular Project

## Prerequisites

1. Node.js and npm installed.  
2. Existing Ionic Angular project.

## Steps to Create Electron App

### Step 1: Install Required Dependencies

Run the following commands to install Electron and other required packages:

npm install --save-dev electron  
npm install electron electron-builder concurrently --save-dev

### Step 2: Update index.html

Modify the <base> tag in index.html located in the www folder (after the Angular build):

<base href="./">

### Step 3: Create Required Files

Create the following files in the root directory of your project:

#### main.js

This file initializes the Electron app.

const { app, BrowserWindow } = require('electron');  
const path = require('path');  
  
function createWindow() {  
 const win = new BrowserWindow({  
 width: 800,  
 height: 600,  
 webPreferences: {  
 preload: path.join(\_\_dirname, 'preload.js'),  
 contextIsolation: true,  
 nodeIntegration: true,  
 },  
 });  
  
 win.loadFile(\_\_dirname + '/www/index.html'); // Serve the Angular app  
}  
  
app.on('ready', createWindow);  
app.on('window-all-closed', () => {  
 if (process.platform !== 'darwin') app.quit();  
});  
app.on('activate', () => {  
 if (BrowserWindow.getAllWindows().length === 0) createWindow();  
});

#### preload.js

This file allows safe communication between Electron and the DOM.

window.addEventListener('DOMContentLoaded', () => {  
 const replaceText = (selector, text) => {  
 const element = document.getElementById(selector);  
 if (element) element.innerText = text;  
 };  
  
 for (const dependency of ['chrome', 'node', 'electron']) {  
 replaceText(`${dependency}-version`, process.versions[dependency]);  
 }  
});

### Step 4: Update package.json

Modify package.json to include Electron scripts and define the entry point:

{  
 "main": "main.js",  
 "scripts": {  
 "start": "ng serve",  
 "build": "ng build",  
 "electron:start": "ng build && concurrently \"ng build --watch\" \"electron .\"",  
 "electron:build": "ng build && electron-builder"  
 }  
}

### Step 5: Build and Run the App

Start Electron App for Development:

npm run electron:start

Build Electron App:

npm run electron:build

### Step 6: Package as an Executable (Optional)

To create a .exe file for Windows:

electron-packager ./ demoIonictoAngularApp --platform=win32 --overwrite

### Step 7: Configure Electron Builder (Optional)

For advanced configurations, add a build section in package.json or create a electron-builder.json file:

{  
 "build": {  
 "appId": "com.example.yourapp",  
 "productName": "YourAppName",  
 "directories": {  
 "output": "dist\_electron"  
 },  
 "files": [  
 "www/\*\*/\*",  
 "main.js"  
 ],  
 "win": {  
 "target": "nsis"  
 },  
 "mac": {  
 "target": "dmg"  
 },  
 "linux": {  
 "target": "AppImage"  
 }  
 }  
}

## Notes

1. Ensure www contains your Ionic app's built files (ng build output).  
2. For further Electron customizations, refer to the Electron Documentation.