### 1) What is Electron.js or Electron?

Electron.js is commonly known as Electron. It is an open-source library developed and maintained by GitHub and used to develop cross-platform desktop applications with HTML, CSS, and JavaScript.

### 2) What is the usage of the Electron? Or what is Electron.js used for?

Electron was formerly known as Atom Shell. It is an open-source software framework developed and maintained by GitHub. It facilitates developers to develop desktop GUI applications using web technologies.

It combines the Chromium rendering engine and the Node.js runtime.

### 3) What should you know before learning Electron?

Before learning Electron, you should have a basic understanding of JavaScript, HTML, and CSS. It will also be beneficial if you have knowledge of a few native Node.js APIs such as file handling, processes, etc.

### 4) Which technology is used in the Electron?

Electron uses Chromium and Node.js with HTML, CSS and JavaScript.

### 5) Why should we learn Electron?

The main advantage of learning Electron is that you can make cross-platform desktop applications and Native Apps easily. You can also use Electron to convert your website into a desktop application.

### 6) Is Electron.js easy to learn?

Electron.js is a framework used to create a native application with web technologies like HTML, CSS and JavaScript. You can easily create a desktop application with the help of Electron as it takes care of the hard parts of the application to focus on the essential part of the application.

### 7) How does Electron work?

Electron is based on JavaScript. You can use it with normal HTML, CSS, and JavaScript skills to build and distribute scalable desktop applications.

It generally contains few important files such as:

* ****js:**** It is used to specify the windows property, splash screen, and other main processes.
* ****html:**** It has the frontend, which is the HTML syntax like any other web apps.
* ****css:**** We can additionally add style.css and other pages as per your requirements.

The Electron is also used to convert your website into a desktop application.

### 8) Is Google Chrome an app made from Electron?

Electron uses Chromium and Node.js. So, every Electron app has a portion of chrome in it.

### 9) How Chromium and Node.js are used in Electron?

Chromium and Node.js both are used in Electron in the following ways:

****Chromium:**** Chromium is a Web kit developed and maintained by Google. It is an internet browser with a V8 JavaScript engine which aids all of the browser and DOM APIs. It is good for making web pages.

****Node.js:**** Node.js keeps backend code JavaScript state break away frontend utility windows kingdom.

### 10) What are the different types of processes in the Electron?

There are two types of processes in the Electron: Main and Renderer.

****Main process:**** Some modules work on each or in both of the processes. The main process is more behind-the-scenes. It is chargeable for interacting with the native graphical consumer interface (GUI) of the working machine.

****Renderer process:**** The Renderer process is answerable for jogging the personal interface of your application via the usage of HTML files, CSS files, JavaScript files and so on. The renderer process runs in each of the windows in your app.

### 11) What is the Inter-Process Communication module in Electron?

Inter-Process Communication module is a mechanism which lets in exchange of synchronous and asynchronous messages between main and renderer process in Electron.

### 12) How can we build a User Interface in Electron?

A User Interface in Electron can be built by using HTML, CSS, and JavaScript.

### 13) What Are Environment Variables In Electron?

Environment variables are used to control application configuration and behavior without changing code. Environment variables control some Electron behaviors because they are initialized earlier than the command-line flags and the app's code.

There are two types of environment variables in Electron.

* Production variable
* Development variable

### 14) What do you understand by Web View in Electron?

In Electron, WebView is a tag that is used to embed web pages. It runs in a separate system and interacts asynchronously between an app and embedded content material.

### 15) What do you know about Packaging Apps?

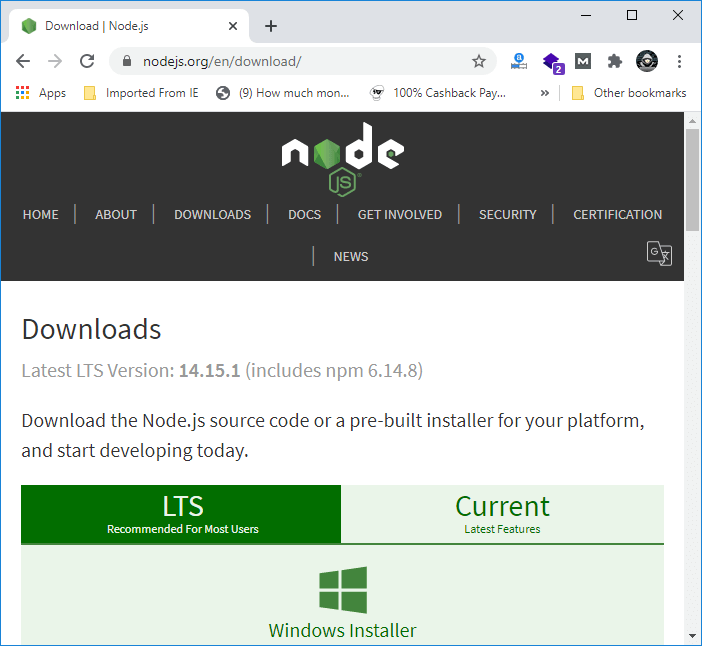
Packaging is an Electron app, refers to creating a desktop installer. Packaging and distributing apps is a crucial part of the improvement system.

These can be accomplished by way of the usage of:

* Electron - packager.
* Electron - builder.

### 16) How could we install Electron on our system?

To install and get started developing using the Electron, we must have Node and npm (node package manager) installed on our system. If you do not have already installed this software, you can download and install these from its official website: [https://nodejs.org/en/download/](https://nodejs.org/en/download/" \t "https://www.javatpoint.com/_blank) on your local system.



You can run the following command in your terminal to confirm that node and npm are successfully installed on your system:

1. node --version
2. npm --version

You can learn more about how to install Node from here: <https://www.javatpoint.com/install-nodejs>

### 17) How can we do debugging in Electron?

Debugging is a process of identifying the problem. It is a set of steps used to identify the problem, find the source of the problem, and then resolve the problem to make the application work successfully.

In Electron, apps are run over the main process and rendered process, so debugging is applied to both processes.

In an Electron browser window, the DevTools is the only debug JavaScript executed in those web pages. It uses an external debugger and launches Electron with commands -debug or -debug-brk, -inspect or inspect-brk.

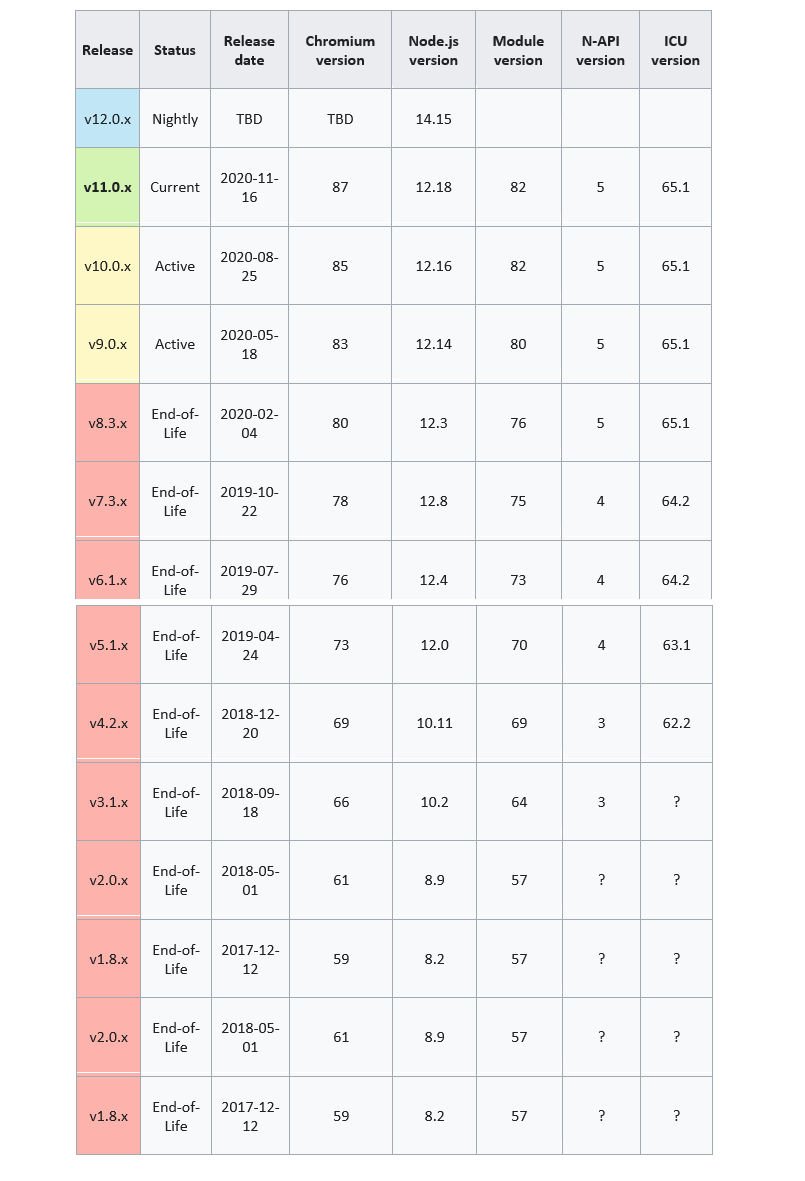
Chrome Devtools is used for the debugging renderer process as it is executed in the browser window.

### 18) Is the Electron secure to use?

Electron applications are web applications run in the Chromium engine so, they may be vulnerable to web-related attacks such as cross-site scripting attacks, through the same attack vectors as a browser (e.g., Chromium) or other internal components (Node.js) if you use the older versions of Electron. These vulnerabilities have been fixed in the latest Electron releases.

### 19) What are the different Electron versions to date?

The following table shows all Electron versions to date:



### 20) What are the different types of menu in Electron?

There are two types of menus are created in Electron desktop apps.

* application menu
* Context menu.

Here, application menus are used to display the top bar and context menu after the right-click.

### 21) What do you know about System Tray in Electron?

In the Electron framework, the system tray represents an icon in the operating system's notification area. In other words, we can say that the system tray is a menu outside of the application window.

### 22) How does notification is created in Electron?

Notification is created by using an npm module called node-notifier in Electron. It is used to notify users in Windows, Linux and MacOS operating systems. Electron also provides native notifications API only for MacOS. The node-notifier module can be installed by using the following command

1. $ npm install -save node-notifier

### 23) Explain the structure of an Electron application?

The basic structure of the Electron application contains the following three files:

* json
* js
* html

Here, package.json file is used for metadata, main.js file is used for code and index.html file is used for the graphical user interface (GUI).

### 24) Which are the biggest companies and software using Electron?

Electron is developed and maintained by Github for creating the text editor Atom. It has been open-sourced since 2014. It is being used by many companies such as Microsoft, Github, Slack, etc.

Electron is now used by a lot of companies to create several apps. Following is a list of some notable apps:

* Slack desktop
* WordPress desktop app
* Visual Studio Code
* Caret Markdown Editor
* Nylas Email App
* GitKraken git client etc.

### 25) Why is Electron criticized as compared to native applications of similar functionality?

Electron applications are generally criticized for containing very significant overhead due to their Chromium dependency. If you compare Electron to other native applications of similar functionality, Electron applications take up more storage and RAM while being slower and providing a non-native user interface.

A few alternatives have been written to use an existing browser instead of bundling Chromium.

Question 1. What Is Electron?

Answer :

Electron is an open source library developed by GitHub. It is used to develop move platform computer utility.

Question 2. Which Technology Electron Uses?

Answer :

Electron uses Chromium and Node.Js with HTML, CSS and JavaScript.

Python Interview Questions  
Question three. How Chromium And Node.Js Are Used In Electron?

Answer :

Chromium is a Web kit, based totally net browser with V8 JavaScript engine which aid all of the browser and DOM APIs. It is good for making web page.

Node.Js keeps backend code JavaScript state break away frontend utility windows kingdom.

Question 4. What Are The Different Types Of Process In Electron?

Answer :

There are  types of system in Electron. These are essential process and renderer method.

Main process is chargeable for interacting with native graphical consumer interface (GUI) of working machine. I additionally reply for vehicle-update of activities and coping with Browser Window times.  
Renderer process is answerable for jogging the person interface of your application via the usage of HTML files, CSS files, JavaScript files and so on.  
Python Tutorial  
Question 5. What Is Inter Process Communication (ipc)?

Answer :

Inter Process Communication module is mechanism which lets in exchange of synchronous and asynchronous message between major and renderer process. IpcMain module communication takes from predominant process to renderer system and ipcRenderer module communique takes from renderer system to most important method.

HTML 5 Interview Questions  
Question 6. How Could We Build User Interface In Electron?

Answer :

User Interface of Electron is built by the usage of HTML, CSS, and JavaScript.

Question 7. What Are Environment Variables In Electron?

Answer :

Environment variables control configuration (conduct) of Electron software without changing code. There are two forms of environment variables in Electron.

Production variable  
Development variable  
HTML five Tutorial Java Script Interview Questions  
Question 8. What Is Web View In Electron?

Answer :

WebView is a tag that is used to embed web web page in electron app. It runs in a separate system and interacts asynchronously between app and embedded content material.

Question nine. Tell Me About Packaging Apps?

Answer :

Packaging is an electron app, honestly refers to creating a desktop installer. Packaging and distributing apps is an imperative a part of the improvement system. These can be accomplished by way of the usage of:

Electron - packager.  
Electron - builder.  
Linux Interview Questions  
Question 10. How Debugging Done In Electron?

Answer :

Debugging is a set of steps that involves identifying the trouble, supply of trouble then resolving the trouble to work software successfully. In electron, app are run over predominant procedure and rendered manner debugging carried out on each approaches.

The Dev Tools in an Electron browser window can most effective debug JavaScript this is execute in that internet pages. It makes use of external debugger and release Electron with instructions –debug or --debug-brk, --look into or investigate-brk.

Chrome Dev tools are used to debug renderer manner as it accomplished in browser window.

Java Script Tutorial  
Question 11. What Are Different Types Of Menu?

Answer :

There are  form of menus are created in electron computer apps.

Application menu  
Context menu  
Application menu are those which display on top bar and context menu display on proper-click.

Mac OS X Deployment Interview Questions  
Question 12. Tell Me Something About System Tray?

Answer :

System tray represents an icon in an running system's notification region in other phrase we are able to say that machine tray is a menu outside of your application window.

Python Interview Questions  
Question 13. How Dose Notification Is Created In Electron?

Answer :

Notification is created by using the use of npm module referred to as node-notified used to inform customers in Windows, Linux and MacOS. Electron additionally affords native notifications API simplest for MacOS.

Node-notified module is set up by using the usage of command $ npm install –store node-notified.

Linux Tutorial  
Question 14. How Electron Works?

Answer :

Electron paintings through taking predominant document created for your package.Json document and executes it. The fundamental report creates application windows carries rendered net pages and interact with local GUI. The most important process is created when you begin your application. Main technique control Browser Window module (offer ability to create new software home windows for your app) and every browser module runs its personal renderer procedure.

Question 15. What Is The Structure Of An Electron Application?

Answer :

The simple structure of electron software includes three documents: bundle.Json, predominant.Js and index.Html. Package.Json used for metadata, primary.Js for code and index.Html for graphical person interface.

Links: https://www.electronjs.org/docs/latest