**Node.js Power Tools: 13 Libraries Every Developer Should Know**

[[](https://pinjarirehan.medium.com/?source=post_page-----94ee989203b9--------------------------------)](https://pinjarirehan.medium.com/?source=post_page-----94ee989203b9--------------------------------)

[[Dev Genius](https://blog.devgenius.io/?source=post_page-----94ee989203b9--------------------------------)](https://blog.devgenius.io/?source=post_page-----94ee989203b9--------------------------------)

[P. Rehan](https://pinjarirehan.medium.com/?source=post_page-----94ee989203b9--------------------------------)

·

Follow

Published in

[Dev Genius](https://blog.devgenius.io/?source=post_page-----94ee989203b9--------------------------------)

·

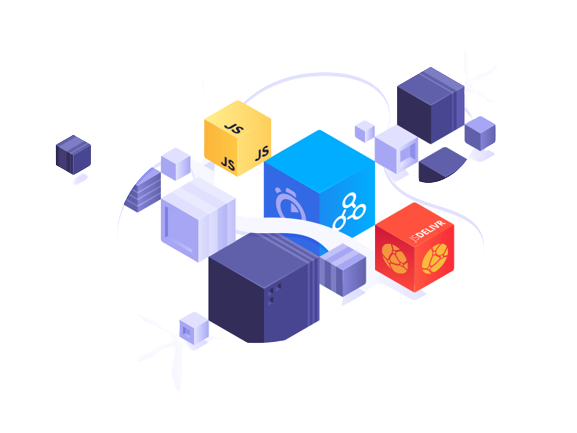
10 min read

·

Oct 14

777

5



Node.js Libraries

Node.js is seen as the ideal runtime environment for many web developers.

Node.js was designed to run code written in JavaScript, one of the most popular programming languages in the world, and it allows a wide community of developers to construct server-side applications.

Node.js offers the ability to reuse code via JavaScript libraries, however selecting the appropriate libraries may be difficult.

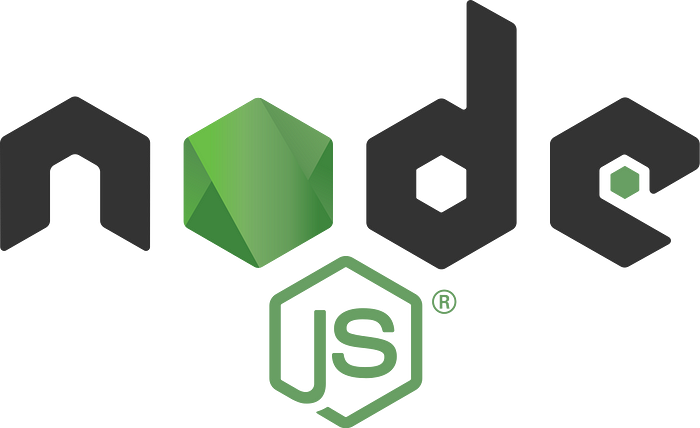
Useful libraries may shorten development time and provide several advantages for your web application, such as quicker load times and decreased application bundle size.

When picking a library, consider the application’s complexity, the community supporting the library, the frequency of updates, and the quality of its documentation.

Node.js libraries are maintained through the Node.js package manager, npm, which may assist in the installation of various open-source libraries.

I have picked 13 important Node.js libraries that make web development simpler.

**What Exactly Is Node.js?**



The NodeJS Logo

Node.js is a server-side runtime environment for JavaScript programming that is open source.

Its asynchronous nature and cross-platform interoperability make it a popular web development base.

Node.js uses event-driven and non-blocking I/O, making it extremely efficient in real-time distributed systems that deal with large amounts of data.

**What Exactly Is a Node.js Library?**



NodeJS Libraries

A library, sometimes known as a module, is pre-written code that encapsulates regularly used operations.

You can utilize libraries to speed up the coding process and encourage code reuse, which will help you keep your work “DRY” (don’t repeat yourself).

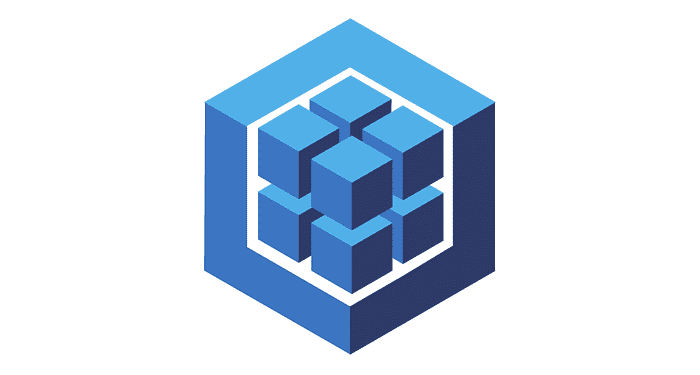
Libraries, as opposed to frameworks, are finished functionality that can be easily used in a project at any stage of development.

A framework, on the other hand, generally serves as a skeleton for a whole program, with substantial influence over how it is created.

**Node.js Libraries That Work So You Don’t Have To**

Let’s have a look at 13 of these Node.js libraries and see what they have to offer.

**1. Sequelize**



The Sequelize Logo

[***Sequelize***](https://sequelize.org/) is a promise-based Node.js object-relational mapper (ORM) use that makes working with relational databases easier for developers.

PostgreSQL, MySQL, MariaDB, SQLite, and more databases are supported.

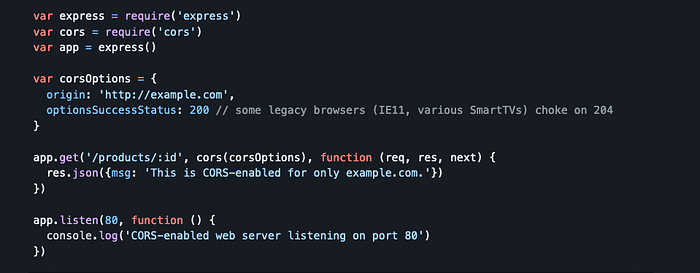
Sequelize models the structure of database tables using JavaScript objects and connects to the favored relational database to query and change data.

It then parses the retrieved data and returns it as a JavaScript object.

**Sequelize Library Features and Benefits**

* Connects to databases and performs operations without writing raw SQL queries
* Reduces SQL injection vulnerabilities and SQL injection attacks
* Compatible with GraphQL

**2. CORS**



An example of CORS code (configuration)

[***CORS***](https://www.npmjs.com/package/cors) is a Node.js package that uses Connect/Express to provide cross-origin resource sharing (CORS) as middleware.

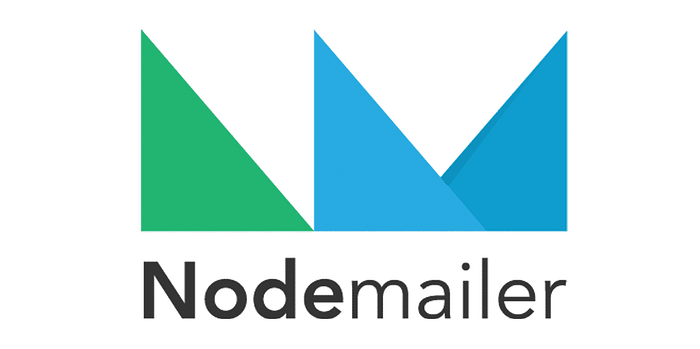
The CORS package wraps the Node.js route middleware, allowing the program to access resources from domains other than its own.

It accepts several arguments to specify cross-origin options, such as origin, headers, and more.

**The CORS Library’s Features and Benefits**

* Cuts down on the amount of code needed to enable CORS in a web application.
* Allows you to specify allow-listed domains and allows the user to enable CORS for some origins while prohibiting others.
* Provides error handling that is smooth and assists developers in analyzing security risks with suspect origins.

**3. Nodemailer**



The Nodemailer Logo

[***Nodemailer***](https://nodemailer.com/about/) streamlines email sending from the Node.js server.

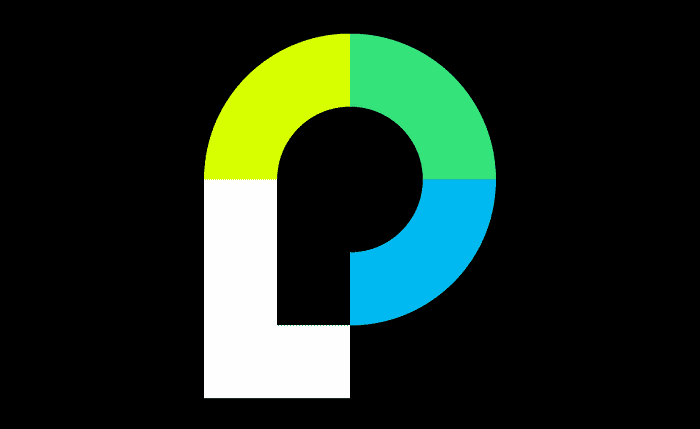
It uses a transport object that, among other supporting transports, is based on the Simple Mail Transfer Protocol (SMTP).

To create a message, this transport object accepts from, to, subject, body, and other parameters as input.

**The Advantages and Features of the Nodemailer Library**

* SMTP, Amazon Simple Email Service (SES), Sendmail, and Stream are all supported by a single module.
* Supports text and HTML material in the email body
* Sets up delivery status notifications and allows for mass email deliveries.

**4. passport**



The Passport Logo

[***Passport***](https://www.passportjs.org/) is a Node.js authentication middleware that is modular.

Over 500 authentication schemes are supported by Passport, including Google, Facebook, Twitter, and other custom and single sign-on (SSO) providers.

Normal username and password login, delegated authentication via OAuth for social networking sites, and OpenID for federated authentication are all options.

**Passport Library Features and Benefits**

* With minimum coding, built-in SSO authentication for social network sites
* Sets up permanent login information for many sessions.
* By using an unobstructed configuration with Express and Connect middleware, it is possible to avoid mounting additional routes in the application.

**5. Async**



The Async Logo

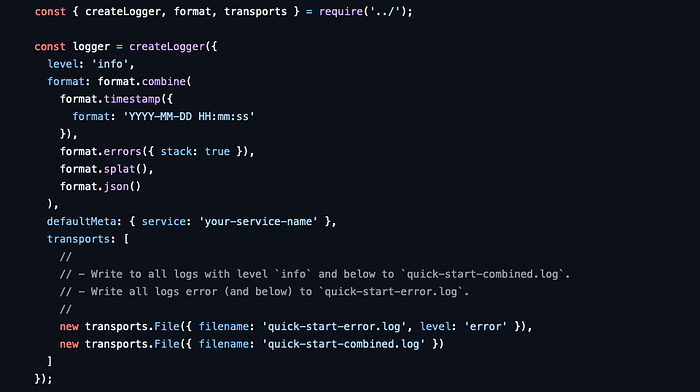
[***Async***](http://caolan.github.io/async/v3/index.html) is a strong Node.js utility module that helps developers in working with asynchronous JavaScript by using JavaScript “async” or callback-accepting methods.

When you offer the Async module an array of callbacks, it runs and wraps them to deliver a promise.

**The Async Library’s Features and Advantages**

* Provides over 70 utility methods for easily developing asynchronous control flow.
* Provides a “parallel” method for dealing with numerous requests to a host (which would otherwise require a significant amount of code to achieve).
* Helps in bringing about the end of nested “callback Hell” in JavaScript.

**6. Winston**



An Example Of Winston Code

[***Winston***](https://www.npmjs.com/package/winston) is a logging package for Node.js that allows universal logging over many transports.

These transporters store and customize logs according to the requirements of your application.

In addition to the default, the ***‘createLogger’*** method allows you to create custom loggers that use available transport choices including consoles, files, and databases.

Custom loggers can also be used in conjunction with custom transports.

**The Winston Library’s Features and Benefits**

* Controls logging from a single configuration file.
* Allows you to customize log formats, such as saving your log in JSON or text format.
* Provides adjustable logging levels that you can customize based on your application’s requirements.

**7. Mongoose**



The Mongoose Logo

[***Mongoose***](https://mongoosejs.com/) is a Node. js-based MongoDB object modeling tool, often known as an object data modeling (ODM) library, that provides features such as hooks, model validation, connecting, and querying.

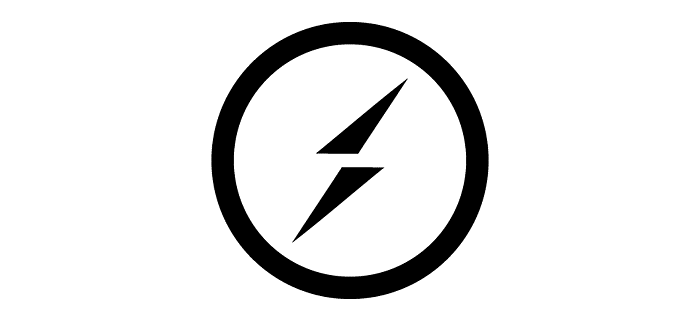
Mongoose provides a schema-based solution for application data by imposing a single schema at the application layer that results in a MongoDB collection.

Each schema is coupled with a Mongoose model, which enables you to perform queries on a MongoDB collection, such as obtaining, updating, and removing data.

**Mongoose Library Features and Benefits**

* Provides simple query abstraction, allowing developers to write less code for MongoDB transactions.
* Built-in data validation allows you to define the types of data that may be added or altered in the database.
* Implements a specified structure for the MongoDB collection, offering developers with a boilerplate MongoDB instance.
* Allows you to work with some queries by chaining them together.

**8. Socket.IO**



The Socket.IO Logo

[***Socket.IO***](https://socket.io/) is a Node.js communication package that allows a client browser and a server to communicate in real time, bi-directionally, and event-based.

It makes a low-level connection between the server and the client by employing a digital handshake through HTTP long-polling.

Once the connection is established, client and server communication takes place in real-time across TCP.

**Sockets.IO Library Benefits & Features**

* Uses WebSocket to provide a low-overhead communication channel, with HTTP long-polling as a backup alternative.
* Scalable, allowing servers to simply broadcast events to numerous clients.
* Supports namespace multiplexing, which decreases the amount of TCP connections and socket ports on the server.

**9. Lodash**



The Lodash Logo

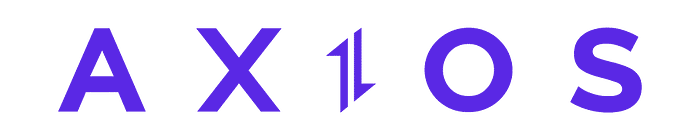
[***Lodash***](https://lodash.com/) is a JavaScript utility package that assists developers in writing simple and maintainable code.

It includes over 200 utility functions to help with common programming chores like as type-checking, simple math operations, and more.

**The Lodash Library’s Features and Advantages**

* Polyfills are used to enable cross-browser compatibility.
* When working with an array of objects, it provides built-in solutions such as filter, search, and flatMap.
* Helps developers avoid redundancy and maintain clean code.

**10. Axios**



The Axios Logo

[***Axios***](https://axios-http.com/) is a Node.js and browser-based promise-based HTTP client.

It also manages the transformation of browser or Node.js request and response data as needed.

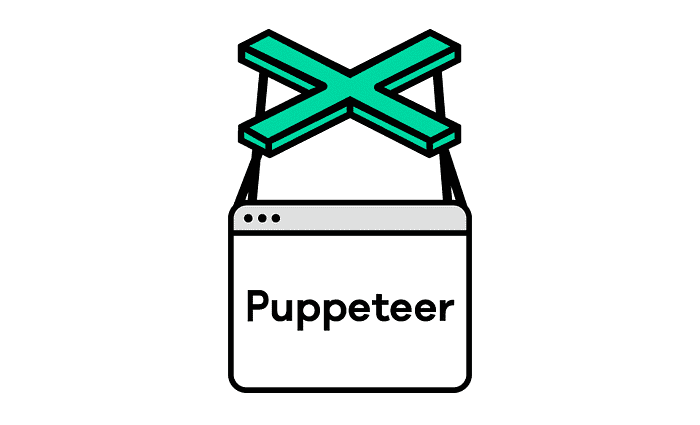
Axios is isomorphic, which means it can operate on both the server and client using the same codebase.

Axios employs a native HTTP module on the server side and XMLHttpRequest on the client side for HTTP communication.

T**he Axios Library’s Features and Advantages**

* Offers API methods for typical HTTP data types such as GET, PUT, POST, and DELETE.
* Increases security by protecting against cross-site request forgery (CSRF) while performing HTTP queries over the Internet.
* Automatic JSON data translation easily converts response data to JSON.

**11. puppeteer**



The Puppeteer Logo

[***Puppeteer***](https://pptr.dev/) is a Node.js framework that allows you to automate Chrome by offering a high-level API for controlling Chrome/Chromium via the DevTools Protocol.

It automates front-end testing, such as request handling tests, identifying and comparing UI components, and performance testing, among other things.

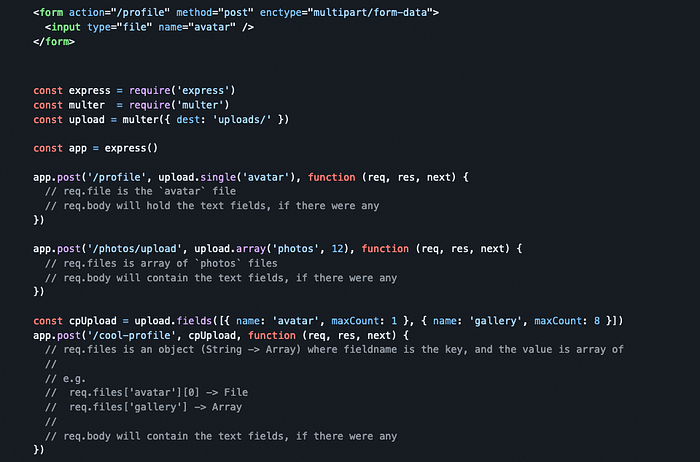
Developers may construct a Chromium instance by importing the Puppeteer package into their code.

The instance may then communicate with the browser engine to automate testing.

**The Puppeteer Library’s Benefits and Features**

* There is no setup, it is simple to configure, and it does not require any additional drivers.
* Websites are crawled to produce pre-rendered content.
* Compatibility with well-known testing frameworks such as Jest and Mocha

**12. Multer**



An Example Of Multer Code

[***Multer***](https://github.com/expressjs/multer) is a Node.js middleware library. It is built on the HTML form parser Busboy and supports multipart and multiform data.

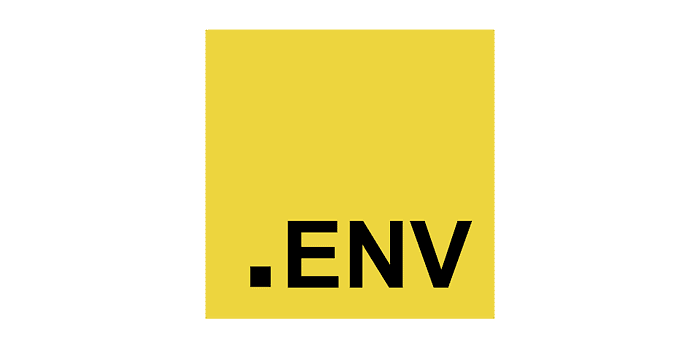
After initializing the Multer instance, one of its parameters is a test object, which specifies where the uploaded file will be saved on the server.

Multer delivers a file object with the upload request, which the Multer API parses and transmits to the target site.

**Multer Library Features and Benefits**

* Through built-in parsing, it makes raw HTTP request data more available for storage.
* Allows you to define the file’s encoding type, which provides an extra degree of protection to the uploaded file.
* Filters and can limit file type and size upload choices

**13. Dotenv**



The Dotenv Logo

[***Dotenv***](https://www.npmjs.com/package/dotenv) is a Node.js utility module that maintains application environment variables and protects critical configuration data.

Dotenv also assists the application in saving environment variables per the twelve-factor app methodology.

When the dotenv library is configured early on, the environment variables from.env are immediately injected into provess.env.

**Dotenv Library Features and Benefits**

* Allows you to segregate secrets from the source code, such as API keys and login credentials, and allows each developer to establish their own.env file.
* Because of its zero-dependency module, it does not add to the size of the program.

**Final Words**

There are many helpful libraries available in Node.js, but choosing the ideal one for your project might be tough.

Some of the Node.js libraries discussed in this post might turn out to be “must-haves” for your future application.

For example, if you do a lot of work with MongoDB, Mongoose can be a lifesaver.

CORS can assist you in keeping up the data across multiple domains at the same time, and Dotenv is especially useful if you need to share code — but not secrets — inside a team.

Don’t go without leaving a $1 coffee tip to support my work.

[[](https://www.buymeacoffee.com/prehan)](https://www.buymeacoffee.com/prehan)

Click 👆 to support!

**👋 Need Web Development Help?**

I’m your go-to web developer, and I’m committed to tailoring solutions to meet your unique needs. Whether you’re in search of a remote team member or a freelance expert, I’m here to provide top-notch services.

Here’s what I offer:

* Front-end development
* Back-end development
* REST API Creation
* Seamless Database Integration
* App UI/UX Design
* Website UI/UX Design
* Smooth Payment Integration
* Function-rich Websites With A/B Testing Capabilities

Feel free to share your specific requirements with me through [***email***](mailto:prehandev@gmail.com), and I’ll bring your vision to life.

**Technologies Stack:**

* React.js
* Node.js
* Next.js
* Express.js
* TypeScript
* MongoDB
* MySQL
* HTML/CSS/JS
* Tailwind CSS
* Bootstrap

**Let’s Collaborate:** Whether you’re a small business owner, entrepreneur, or startup, we can bring your web app ideas to life. Contact me now to discuss your project and let’s make your website or app a reality. Your satisfaction is my top priority.

**🤝 Ready to Get Started?** Reach out to me at [**prehandev@gmail.com**](mailto:prehandev@gmail.com) or connect with me on [**Upwork**](https://www.upwork.com/workwith/prehan) or [**LinkedIn**](https://www.linkedin.com/in/pinjari-rehan/).

Thank you for considering me. I look forward to the opportunity to work together!

Warm regards,  
**P. Rehan.**