

SHREE H.N.SHUKLA COLLEGE OF I.T & MGMT

(Affiliated to Saurashtra University and G.T.U)



2 – Vaishalinagar,
Nr.Amrapali Railway Under Bridge
Raiya Road,
Rajkot – 360001.
Ph No. 0281- 2440478

3 – Vaishalinagar,
Nr. Amrapali Railway Under Bridge
Raiya Road,
Rajkot – 360001.
Ph No. 0281- 2471645



Lt. Shree Chimanbhai Shukla

MScIT SEM-3 :: CS-14: Node.js

Shree H.N.Shukla College
Street No. 2, Vaishali Nagar,
Nr. Amrapali Under Bridge,
Raiya Road, Rajkot.
Ph. (0281)2440478, 2472590



Shree H.N.Shukla College
Street No. 3, Vaishali Nagar,
Nr. Amrapali Under Bridge,
Raiya Road, Rajkot.
Ph. (0281)2471645

Shree H.N.Shukla College of I.T & Management "Sky is the Limit"

SHREE H.N.SHUKLA COLLEGE OF I.T & MGMT

(Affiliated to Saurashtra University and G.T.U)



2 – Vaishalinagar,
Nr.Amrapali Railway Under Bridge
Raiya Road,
Rajkot – 360001.
Ph No. 0281- 2440478

3 – Vaishalinagar,
Nr. Amrapali Railway Under Bridge
Raiya Road,
Rajkot – 360001.
Ph No. 0281- 2471645

CS – 14: NODE JS

MSc(IT) SEMESTER : 03 :: CS-14: Node.js		
No	Topics	Details
1.	Introduction, Set up Development Environment, Other JavaScript Based Technologies	• Features and advantages of Node JS • Traditional Web Server Model • Node.js Process Model, • Asynchronous programming with Node.js • Types of applications that can be developed using Node.js • Install Node.js on Windows • working in REPL • Node JS Console • Creating a Node File with JavaScript • Accessing a Node.js File Through the Command Line Interface • Using Node.js in IDE • Node.js vs JavaScript • Node.js vs AJAX • Node.js vs JQuery • Node.js vs Angular JS

SHREE H.N.SHUKLA COLLEGE OF I.T & MGMT

(Affiliated to Saurashtra University and G.T.U)



2 – Vaishalinagar,
Nr.Amrapali Railway Under Bridge
Raiya Road,
Rajkot – 360001.
Ph No. 0281- 2440478

3 – Vaishalinagar,
Nr. Amrapali Railway Under Bridge
Raiya Road,
Rajkot – 360001.
Ph No. 0281- 2471645

What is Node.js?

Node.js is an open-source, cross-platform JavaScript runtime environment suitable for a wide range of projects. It stands out as a game-changer in the development landscape by enabling the use of JavaScript both in the browser and on the server side. This dual capability enhances the flexibility and power of JavaScript, providing a robust toolset for developers.

- Node.js is an open-source, cross-platform JavaScript runtime environment.
- It is suitable for a wide range of projects.
- Node.js stands out as a game-changer in the development landscape.
- It allows the use of JavaScript both in the browser and on the server side.
- This dual capability enhances the flexibility and power of JavaScript.
- Node.js provides a robust toolset for developers.

What are the key advantages of Node.js?

Node.js is an open-source, cross-platform JavaScript runtime environment that executes JavaScript code outside of a web browser. It's a powerful tool used for various types of projects. Let's explore some key aspects:

- 1. JavaScript Runtime:** Node.js runs on the V8 JavaScript engine, which also powers Google Chrome. Unlike the browser context, Node.js executes JavaScript code outside of the browser.
- 2. Single Process Model:** Node.js applications operate within a single process, avoiding the need to create a new thread for every request. This design choice contributes to Node.js's performance.
- 3. Asynchronous I/O:** Node.js provides asynchronous I/O primitives in its standard library, preventing JavaScript code from blocking and making non-blocking behavior the norm. For I/O operations like reading from the network, accessing databases, or the filesystem, Node.js resumes operations only when the response arrives, thus avoiding CPU cycle wastage.
- 4. Concurrency Handling:** Node.js efficiently handles thousands of concurrent connections using a single server, avoiding the complexities and bugs associated with managing thread concurrency.
- 5. JavaScript Everywhere:** Frontend developers familiar with JavaScript can seamlessly transition to writing server-side code using Node.js, without the need to learn a different language.
- 6. ECMAScript Standards:** Node.js supports the latest ECMAScript standards, allowing developers to choose the version they want to use, independent of users' browser updates.

SHREE H.N.SHUKLA COLLEGE OF I.T & MGMT

(Affiliated to Saurashtra University and G.T.U)



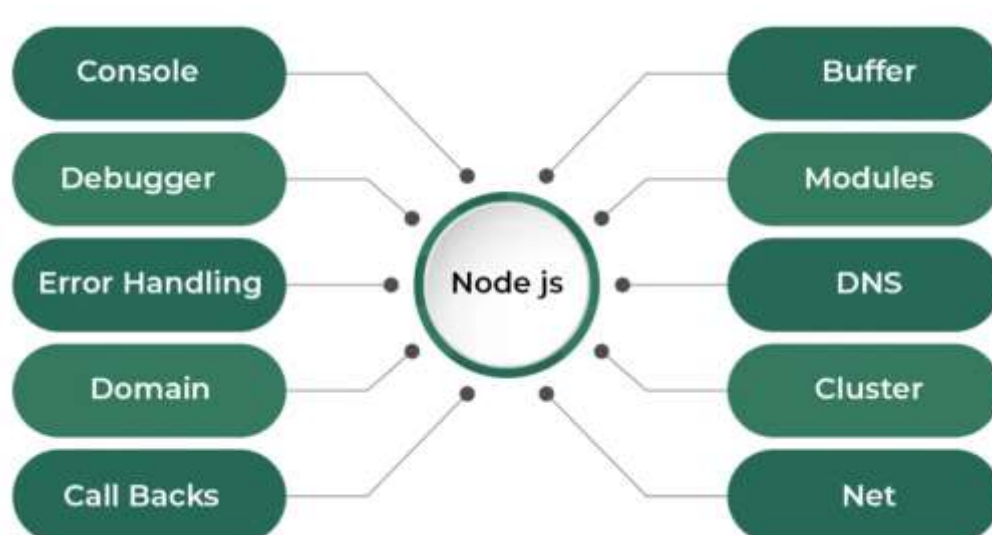
2 – Vaishalinagar,
Nr.Amrapali Railway Under Bridge
Raiya Road,
Rajkot – 360001.
Ph No. 0281- 2440478

3 – Vaishalinagar,
Nr. Amrapali Railway Under Bridge
Raiya Road,
Rajkot – 360001.
Ph No. 0281- 2471645

Why Node.js is popular as a backend?

Node.js is used to build back-end services like APIs for web apps, mobile apps, and web servers. Big companies like PayPal, Uber, Netflix, and Walmart use Node.js in their production systems. Here are the main reasons to choose Node.js:

- 1. Easy to Start:** Node.js is easy to learn and great for quick development.
- 2. Scalable:** Node.js can grow with your app, both in terms of adding more servers and making better use of each server.
- 3. Real-Time Apps:** Node.js is perfect for apps that need real-time updates.
- 4. Fast Performance:** Node.js can quickly handle tasks like accessing databases and making network connections.
- 5. Unified Language:** You can use JavaScript for both the front-end and back-end, simplifying development.
- 6. Large Ecosystem:** Node.js has many open-source libraries and supports non-blocking programming, which makes it very flexible and efficient.



SHREE H.N.SHUKLA COLLEGE OF I.T & MGMT

(Affiliated to Saurashtra University and G.T.U)



2 – Vaishalinagar,
Nr.Amrapali Railway Under Bridge
Raiya Road,
Rajkot – 360001.
Ph No. 0281- 2440478

3 – Vaishalinagar,
Nr. Amrapali Railway Under Bridge
Raiya Road,
Rajkot – 360001.
Ph No. 0281- 2471645

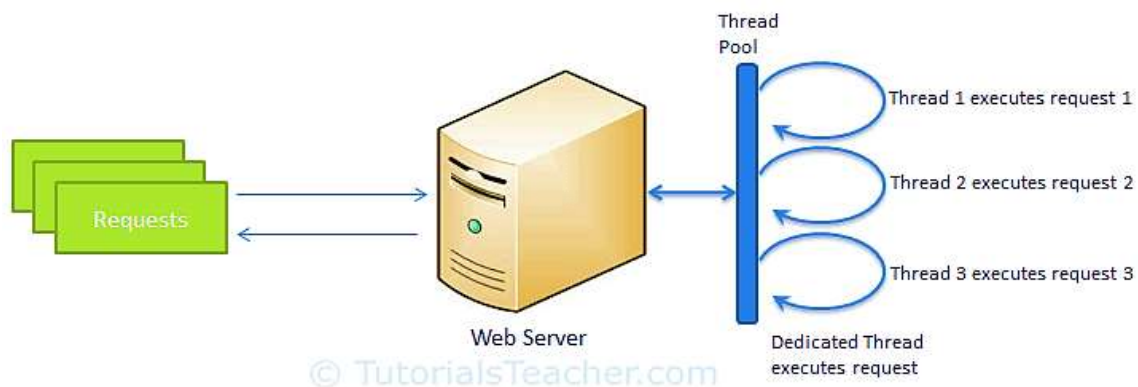
What are the key features and advantages of Node.js?

1. **Event-Driven and Non-Blocking I/O:** Node.js uses an event-driven, non-blocking I/O model, making it efficient and suitable for data-intensive real-time applications.
2. **Single-Threaded Model:** Node.js operates on a single thread using non-blocking I/O calls, allowing it to support tens of thousands of concurrent connections.
3. **Fast Execution:** Built on the V8 JavaScript engine from Google Chrome, Node.js executes code quickly and efficiently.
4. **Scalability:** Node.js applications can handle a large number of simultaneous connections with high throughput, making it highly scalable.
5. **Rich Ecosystem:** The npm (Node Package Manager) provides access to thousands of open-source libraries and tools, facilitating faster development.
6. **Cross-Platform Compatibility:** Node.js applications can run on various operating systems, including Windows, Linux, and macOS.
7. **Community Support:** A large and active community provides extensive support, tutorials, and frameworks for Node.js developers.

How does the traditional web server model differ from Node.js?

Traditional Web Server Model

In the traditional web server model, each request is handled by a dedicated thread from the thread pool. If no thread is available in the thread pool at any point of time then the request waits till the next available thread. Dedicated thread executes a particular request and does not return to thread pool until it completes the execution and returns a response.



SHREE H.N.SHUKLA COLLEGE OF I.T & MGMT

(Affiliated to Saurashtra University and G.T.U)

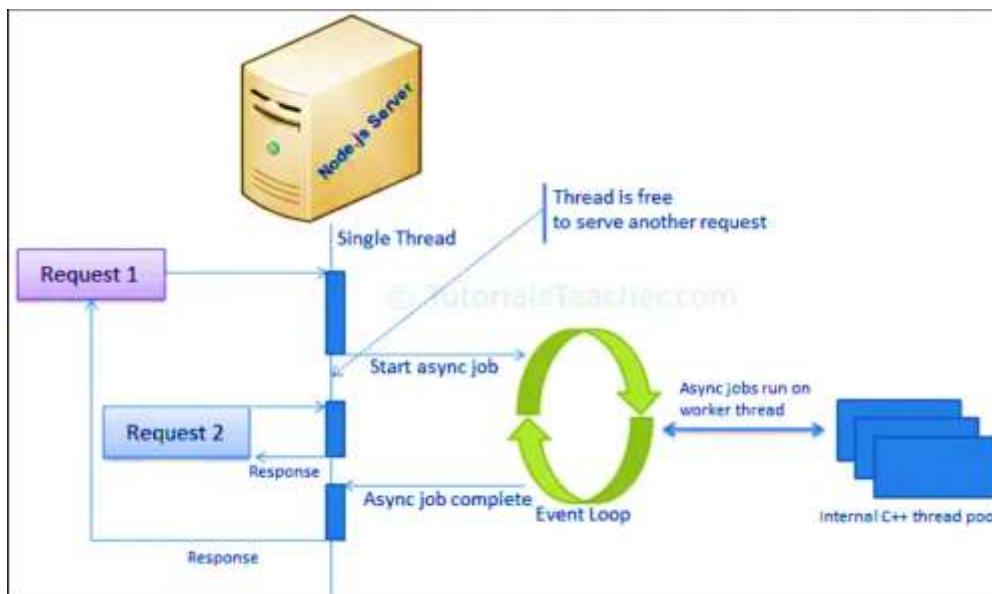


2 – Vaishalinagar,
Nr.Amrapali Railway Under Bridge
Raiya Road,
Rajkot – 360001.
Ph No. 0281- 2440478

3 – Vaishalinagar,
Nr. Amrapali Railway Under Bridge
Raiya Road,
Rajkot – 360001.
Ph No. 0281- 2471645

Node.js Process Model

Node.js processes user requests differently when compared to a traditional web server model. Node.js runs in a single process and the application code runs in a single thread and thereby needs less resources than other platforms. All the user requests to your web application will be handled by a single thread and all the I/O work or long running job is performed asynchronously for a particular request. So, this single thread doesn't have to wait for the request to complete and is free to handle the next request. When asynchronous I/O work completes then it processes the request further and sends the response.



Traditional Web Server Model vs. Node.js Process Model

Feature	Traditional Web Server Model	Node.js Process Model
Concurrency Handling	Multi-threaded, each request handled by a separate thread	Single-threaded, non-blocking I/O with event loop
Resource Usage	High memory and CPU usage per thread	Efficient memory usage due to single-threaded model
Scalability	Limited by the number of threads	Highly scalable, can handle many concurrent connections
Complexity	Easier to manage synchronous code	Requires understanding of asynchronous programming
Latency	Higher latency due to context switching	Lower latency, better performance for I/O-bound tasks

SHREE H.N.SHUKLA COLLEGE OF I.T & MGMT

(Affiliated to Saurashtra University and G.T.U)



2 – Vaishalinagar,
Nr.Amrapali Railway Under Bridge
Raiya Road,
Rajkot – 360001.
Ph No. 0281- 2440478

3 – Vaishalinagar,
Nr. Amrapali Railway Under Bridge
Raiya Road,
Rajkot – 360001.
Ph No. 0281- 2471645

Explain asynchronous Programming with Node.js

What is asynchronous programming and how is it implemented in Node.js?

1. **Asynchronous Programming:** A programming paradigm where operations are executed without blocking the executing thread, allowing other operations to proceed simultaneously.
2. **Callbacks:** Functions passed as arguments to be executed after the completion of asynchronous operations.
3. **Promises:** Objects representing the eventual completion (or failure) of an asynchronous operation, providing methods like `.then()` and `.catch()` for handling results.
4. **Async/Await:** Syntactic sugar over promises, allowing asynchronous code to be written in a synchronous-like manner, improving readability and maintainability.

Types of Applications Developed Using Node.js

What types of applications are best suited for development with Node.js?

1. **Real-Time Applications:** Such as chat applications, online gaming, and live streaming, due to Node.js's efficient handling of multiple simultaneous connections.
2. **Single Page Applications (SPAs):** Where the application needs to update dynamically without reloading the page.
3. **API Services:** For building RESTful APIs and handling multiple concurrent requests efficiently.
4. **Microservices:** Breaking down large applications into smaller, manageable services.
5. **IoT Applications:** Handling a large number of connections with minimal overhead.

SHREE H.N.SHUKLA COLLEGE OF I.T & MGMT (Affiliated to Saurashtra University and G.T.U)



2 – Vaishalinagar,
Nr.Amrapali Railway Under Bridge
Raiya Road,
Rajkot – 360001.
Ph No. 0281- 2440478

3 – Vaishalinagar,
Nr. Amrapali Railway Under Bridge
Raiya Road,
Rajkot – 360001.
Ph No. 0281- 2471645

Installing Node.js on Windows

How do you install Node.js on a Windows machine?

1. Download Node.js: Visit the official Node.js website (<https://nodejs.org/>) and download the Windows installer.
2. Run Installer: Open the downloaded installer and follow the installation prompts.
3. Set Path Variable: During installation, ensure the option to add Node.js to the system PATH is selected.
4. Verify Installation: Open Command Prompt and run `node -v` and `npm -v` to verify Node.js and npm are installed correctly.

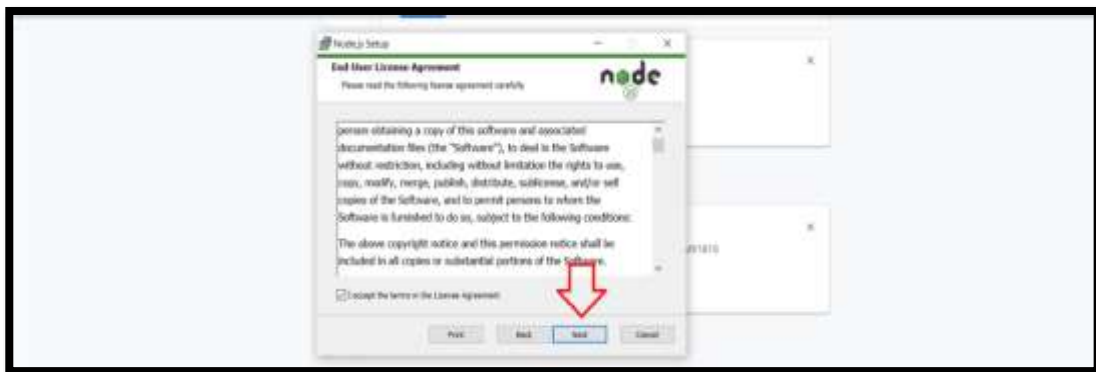


SHREE H.N.SHUKLA COLLEGE OF I.T & MGMT (Affiliated to Saurashtra University and G.T.U)



2 – Vaishalinagar,
Nr.Amrapali Railway Under Bridge
Raiya Road,
Rajkot – 360001.
Ph No. 0281- 2440478

3 – Vaishalinagar,
Nr. Amrapali Railway Under Bridge
Raiya Road,
Rajkot – 360001.
Ph No. 0281- 2471645



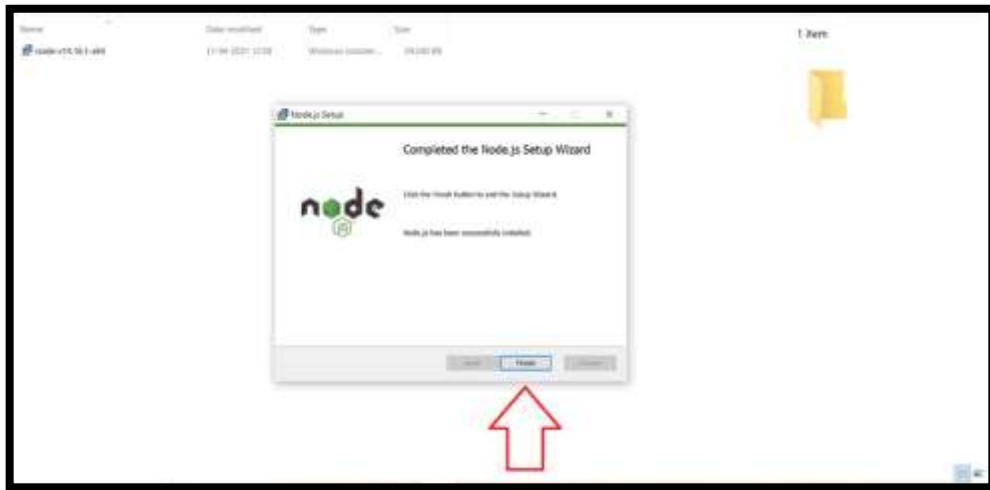
SHREE H.N.SHUKLA COLLEGE OF I.T & MGMT

(Affiliated to Saurashtra University and G.T.U)



2 – Vaishalinagar,
Nr.Amrapali Railway Under Bridge
Raiya Road,
Rajkot – 360001.
Ph No. 0281- 2440478

3 – Vaishalinagar,
Nr. Amrapali Railway Under Bridge
Raiya Road,
Rajkot – 360001.
Ph No. 0281- 2471645



```
Administrator: Command Prompt
Microsoft Windows [Version 10.0.18363.1440]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\WINDOWS\system32>node -v
v14.16.1

C:\WINDOWS\system32>
```

SHREE H.N.SHUKLA COLLEGE OF I.T & MGMT

(Affiliated to Saurashtra University and G.T.U)



2 – Vaishalinagar,
Nr.Amrapali Railway Under Bridge
Raiya Road,
Rajkot – 360001.
Ph No. 0281- 2440478

3 – Vaishalinagar,
Nr. Amrapali Railway Under Bridge
Raiya Road,
Rajkot – 360001.
Ph No. 0281- 2471645

Working in REPL

What is REPL and how do you work in it with Node.js?

1. **REPL Definition:** Read-Eval-Print Loop, an interactive shell for executing JavaScript code.
2. **Using REPL:**
 - **Read:** Input JavaScript code.
 - **Eval:** The code is evaluated.
 - **Print:** The result is output.
 - **Loop:** The process repeats.
3. **Exit REPL:** Type .exit or press Ctrl+C twice.

```
C:\Windows\system32\cmd.exe - node
C:\>node
Welcome to Node.js v14.18.0.
Type ".help" for more information.
>
```

```
C:\Windows\system32\cmd.exe - node
> "Hello" + "World"
'HelloWorld'
> 10 + 20
30
>
```

```
C:\Windows\system32\cmd.exe - node
> var x=10, y=20;
undefined
> x + y
30
>
```

```
C:\Windows\system32\cmd.exe - node
> function multiply(x, y)
... {
... return x*y;
... }
undefined
> multiply(10, 20)
200
>
```

SHREE H.N.SHUKLA COLLEGE OF I.T & MGMT

(Affiliated to Saurashtra University and G.T.U)



2 – Vaishalinagar,
Nr.Amrapali Railway Under Bridge
Raiya Road,
Rajkot – 360001.
Ph No. 0281- 2440478

3 – Vaishalinagar,
Nr. Amrapali Railway Under Bridge
Raiya Road,
Rajkot – 360001.
Ph No. 0281- 2471645

You can execute an external JavaScript file by executing the node fileName command. For example, the following runs mynodejs-app.js on the command prompt/terminal and displays the result.

```
mynodejs-app.js  
console.log("Hello World");
```

C:\Windows\system32\cmd.exe

```
D:\>node mynodejs-app.js  
Hello World  
  
D:\>_
```

To **exit** from the REPL terminal, press Ctrl + C twice or write .exit and press Enter.

Node.js Console

How do you use the console in Node.js?

1. **Console Object:** Provides methods to interact with the console, similar to the browser's console.
2. **Common Methods:**
 - o `console.log()`: Outputs a message to the console.
 - o `console.error()`: Outputs an error message.
 - o `console.warn()`: Outputs a warning message.
 - o `console.table()`: Displays tabular data.
 - o `console.time()` and `console.timeEnd()`: Measures time taken by a block of code.
3. Usage Example:

```
console.log("Hello, World!");  
console.error("This is an error message.");  
console.warn("This is a warning message.");  
console.table([ { name: "John", age: 25 }, { name: "Jane", age: 28 } ]);  
console.time("Timer");  
// Code to be timed  
console.timeEnd("Timer");
```

SHREE H.N.SHUKLA COLLEGE OF I.T & MGMT

(Affiliated to Saurashtra University and G.T.U)



2 – Vaishalinagar,
Nr.Amrapali Railway Under Bridge
Raiya Road,
Rajkot – 360001.
Ph No. 0281- 2440478

3 – Vaishalinagar,
Nr. Amrapali Railway Under Bridge
Raiya Road,
Rajkot – 360001.
Ph No. 0281- 2471645

Creating a Node File with JavaScript

How do you create and run a Node.js file with JavaScript code?

1. **Create a File:** Create a new file with a `.js` extension, e.g., `app.js`.
2. **Write Code:** Write your JavaScript code in the file, e.g.,

```
console.log("Hello, Node.js!");
```

3. **Run the File:** Open Command Prompt or Terminal, navigate to the file's directory, and run `node app.js`.

Accessing a Node.js File Through the Command Line Interface

How do you access and run a Node.js file through the Command Line Interface (CLI)?

1. **Navigate to Directory:** Open Command Prompt or Terminal and navigate to the directory containing your Node.js file using the `cd` command.
2. **Run the File:** Use the `node` command followed by the file name, e.g.,

```
node app.js
```

Using Node.js in an IDE

How do you use Node.js in an Integrated Development Environment (IDE)?

1. **Choose an IDE:** Popular IDEs for Node.js development include Visual Studio Code, WebStorm, and Atom.
2. **Set Up Environment:** Install the Node.js extension or plugin for your chosen IDE.
3. **Create Project:** Create a new project or open an existing one.
4. **Write Code:** Write your Node.js code in the editor.
5. **Run Code:** Use the IDE's built-in terminal or run configuration to execute your Node.js files.

SHREE H.N.SHUKLA COLLEGE OF I.T & MGMT

(Affiliated to Saurashtra University and G.T.U)



2 – Vaishalinagar,
Nr.Amrapali Railway Under Bridge
Raiya Road,
Rajkot – 360001.
Ph No. 0281- 2440478

3 – Vaishalinagar,
Nr. Amrapali Railway Under Bridge
Raiya Road,
Rajkot – 360001.
Ph No. 0281- 2471645

Node.js vs JavaScript

What are the differences between Node.js and JavaScript?

Feature	Node.js	JavaScript
Environment	Server-side runtime environment	Client-side scripting language
Usage	Building server-side applications	Enhancing web pages, client-side logic
APIs	Provides APIs for server-side tasks	DOM manipulation, events, browser APIs
File System	Access to file system, streams, etc.	Limited to browser sandbox
Modules	Uses CommonJS modules	Uses ES6 modules (import/export)
Event Handling	Built-in support for asynchronous I/O	Relies on browser event handling
Execution	Runs in Node.js runtime	Runs in web browsers

Node.js vs AJAX

What are the differences between Node.js and AJAX?

Feature	Node.js	AJAX
Definition	Server-side runtime environment	Technique for asynchronous HTTP requests
Environment	Runs on server-side (Node.js runtime)	Runs on client-side (browser)
Usage	Building server-side applications	Fetching data from server without reloading
Execution	Executes JavaScript on the server	Executes JavaScript in the browser
Purpose	Backend logic, file handling, networking	Updating web pages asynchronously
Libraries/Tools	Uses built-in Node.js modules	Uses XMLHttpRequest or Fetch API
Data Handling	Handles full server-side logic	Handles data exchange between client and server

SHREE H.N.SHUKLA COLLEGE OF I.T & MGMT

(Affiliated to Saurashtra University and G.T.U)



2 – Vaishalinagar,
Nr.Amrapali Railway Under Bridge
Raiya Road,
Rajkot – 360001.
Ph No. 0281- 2440478

3 – Vaishalinagar,
Nr. Amrapali Railway Under Bridge
Raiya Road,
Rajkot – 360001.
Ph No. 0281- 2471645

Node.js vs jQuery

What are the differences between Node.js and jQuery?

Feature	Node.js	jQuery
Definition	Server-side runtime environment	Client-side JavaScript library
Environment	Runs on server-side (Node.js runtime)	Runs on client-side (browser)
Usage	Building server-side applications	DOM manipulation, event handling, animations
Execution	Executes JavaScript on the server	Executes JavaScript in the browser
Purpose	Backend logic, file handling, networking	Simplifying client-side scripting
Libraries/Tools	Built-in Node.js modules	jQuery library (requires inclusion in HTML)
Compatibility	Works on various server environments	Works across all modern browsers

Node.js vs AngularJS

What are the differences between Node.js and AngularJS?

Feature	Node.js	AngularJS
Definition	Server-side runtime environment	Client-side web framework
Environment	Runs on server-side (Node.js runtime)	Runs on client-side (browser)
Usage	Building server-side applications	Building single-page applications (SPAs)
Execution	Executes JavaScript on the server	Executes JavaScript in the browser
Purpose	Backend logic, file handling, networking	Frontend development, dynamic web apps
Libraries/Tools	Built-in Node.js modules	AngularJS framework (requires inclusion in HTML)
MVC Pattern	Not inherently MVC	Follows MVC (Model-View-Controller) pattern