//newly created mongoDB

//username : ehirim87

//password: qDV4wgdd5ajz8MPW

**Frontend**

Install vite - npm create vite@latest

Install node –

Run dev environ – npm run dev

Install tailwind –

Update node version – nvm install node

**Tailwind Pros**

* You dont have to deal with classes.
* You have pre-defined styles.

**React Must Know**

* Data types, variables, functions, loops etc.
* Promises & asynchronous programming.
* Array methods like forEach() & map().
* Fetch API & making HTTP requests.
* Components have state.
* State is an object that determines how component renders and behaves.
* App or global state is a state available to the entire UI not just a single component.
* We use functional components with hooks.
* Context api used to manage multiple app states that may be difficult to manage.
* Context API is built into react.
* Redux is a third-party state manager like context API.
* React hooks are functions that let us hook into React State and lifecycle features from function components.
  + useState: Returns a stateful value and a function to update it.
  + useState: Performs side effects in function components. (used to make http requests).
  + useContext, useReducer, useRef:
* index.js is the entry point for react.
* App is the root component.
* You can only return a single parent element in components.
* You can also use fragments or angle brackets to return components.

**Backend/Full-Stack**

* Node & Express
* Complex rest API
* MERN App
* More Projects

**Node JS**

* Node fundamentals
* Express.js
* MongoDB, Mongoose
* Applications
* Deployment

**What is Node.js**

* An environment to run JS outside the browser.
* Built on Chrome’s V8 JS engine. (compiles our code to machine code)
* Big community, created in 2019.
* Full-Stack.
* JS Nuggets YouTube channel/coding addict

**Differences between Browser and Node JS -** [**https://johnsmilga.com/**](https://johnsmilga.com/)

|  |  |
| --- | --- |
| Browser JS | Node JS |
| DOM | No DOM |
| Window | No Window |
| Interactive Apps | Server-Side Apps |
| No File System | File System |
| Fragmentation | Versions |
| ES Modules | Common JS |

**Modules in Node**

* We split codes in modules even if they are in one file.
* Module is an encapsulated code that only shares minimum.
* Common JS, every file is a module by default.
* Common JS is a standard used in Node.js for working with modules.
* Suitable for server-side.
* Uses require and module.exports.
* Cannot be analyzed statistically.
* Npm ecosystem is built on CM JS.
* Load synchronously.
* ES modules are preferred to CM JS modules.
* Node uses CM JS under the hood.
* Modules are like variable/class encapsulation in Java.

**Built-in Modules (no need to install any modules)**

* OS: provides many useful properties and methods for OS and server
* PATH: provides methods that helps with directories/file paths
* FS: provides methods and properties to read a file or write on it.
* HTTP: provides methods to create and consume API (endpoints) – CRUD. It helps us to set up a web server. Express helps to manage servers and routes.
* When we install node, we auto install npm.

**External Modules/Modules/Dependencies**

* To find packages visit <https://www.npm.js>
* To check npm version npm –v.
* We can install as a local dependency (only the project) or as global in any project with -g <packageName>.
* To install package – npm init or npm init -y
* All dependencies are stored in the node module.
* In the package.json we provide info about our project.
* You can manually add dependencies on the json file and run npm install update to update the node module with the new dependencies.
* Node modules should be added to .gitignore before committing files to repository
* Anybody using/cloning our repo files can have auto dependencies using npm install.
* Another dependency nodemon that auto restarts node when it detects a slight change in the node file.
* Command for dev dependencies is adding –D to the install or –save –dev.

Event Loop/Async Patterns/Event Emitter & Stream

Main Concept

Pre-Built Code