Node Lectureflow

Module 12) Node - Javascript Fundamental	2

- JavaScript Introduction, Difference between JavaScript and Java, Client Server Model, Objects in JavaScript
- Arrow Function, Bind, Apply, Closure, Classes in JavaScript, JavaScript JSON
- Practicals

Module 13) Node - NodeJS - Introduction	5

- Node JS introduction, Chrome V8 engine, Core Modules, Local Modules, NPM Modules, module.exports, nodemon
- Command line arguments, Process object, Args parsing with Yargs, Storing data with JSON, Filter VS Find, Reading and Writing notes
- File Module example, Local Module example, Validator npm library example
- Create TODO app in node js using command line arguments

Module 14) Node - Web Development with Node	8

- Asynchronous Node JS , Making HTTP request
- JSON Parsing, Geo-coding, Error Handling
- Callback function, Callback chaining, De-structure Object
- Create an app to fetch temperature of given city using geocoding and temperature API,
- Web Server with Express, Install express, Create server
- Serve data like HTML, JSON Object, JSON Array, Static Assets, Path Module
- Template Engine, Dynamic Page, Passing data to .hbs file, Handlebars Particles, Error 404 Pages
- Practical of - Returning html response, Returning json object, Returning json array, Display image from static assets
- Use of CSS and JavaScript file from Assets folder, Create header, footer particles and use in different hbs files like index.hbs, about.hbs, help.hbs
- Accessing API from browser, Query String, Call Weather and Geo Code API from browser
- ES6 : Default Function Parameter, Default Value with De-structuring, Browser HTTP request with fetch()
- Search Form with prevent form submission and fetch weather information for city input by user

Module 15) Node - Node with Mongodb	6

- Mongodb vs MySQL, Installing MongoDB , Robo 3t GUI viewer, Connecting to Mongodb, Object ID, ES6 : Promises
- Inserting Document, Insert Bulk Documents, Read one Document, Read all Documents
- Count all Documents, Update One Document, Update Many Documents, Deleting Documents
-

- Create User Model with name (string) and age (int) field and save it. Create Task model with description (String) and completed (Boolean) fields and save it.
- User Age is more 18 - Remove Space around name, Custom validator : age&gt;0 - Custom validator : email validation
- Add Password field to User with proper validation, Apply validation to Task model

Module 16) Structuring REST API	**4**

- Request, Response, Resource Creation End Points, HTTP status, Resource Reading End Points, Promise chaining
- ES6 - Async/Await, Resource Updating End Points, Deleting End points, Separating Route files for user and task
- Create User &amp; Task creation end point with success and error handling , Create User &amp; Task update end point with success and error handling
- Create User &amp; Task delete end point with success and error handling, Create User &amp; Task reading end point with success and error handling

Module 17) - Node - [API Authentication &amp; Security]	**3**

- Securely storing password, Comparing password when login, Mongoose Middleware, Unique email id in User model
- JSON Web Tokens, Verify Token and Set Expiry, Express Middleware, Logging out
- Login , Logout with token, Hiding private data, Encrypt password - Read Task for login user only
- Api authentication with token
- Image uploading in api
- Payment with node
- socket programming , like chat app
- SMs and email sending , for OTP verification