

11/05/24

Ushantika Khamru

PAGE NO.:



## ASSIGNMENT 3.1

### NODE.js

- ① **npm** : Node.js package manager.
  - 'npm init' : Initialize a new project
  - 'npm install <package>' :  
Install a package locally.
- ② **Modules** : Modularize code.
  - 'require(' <module> '):  
↳ Import a module.
  - 'module.exports' = <object> :  
↳ Export a module.
- ③ **File System** : Interact with the file system
  - 'fs.readFile(<file>, <callback>)' :  
↳ Read a file asynchronously.
  - 'fs.writeFileSync(<file>, <data>)' :  
↳ Write to a file synchronously.
- ④ **HTTP** :- Create a server and handle requests
  - 'http.createServer(<requestListener>)' :  
↳ Create an HTTP Server.
  - 'res.writeHead(<statusCode>, <header>)' :  
↳ Write response headers.
  - 'res.end(<data>)' :  
↳ End the response.



⑤ Events :- Utilize event-driven architecture.

- 'eventEmitter.on (<event>, <listener>)' :  
↳ Attach an event listener.
- 'eventEmitter.emit (<event>)' :  
↳ Emit an event.

⑥ Promises :- Handle asynchronous operations.

- 'new Promise ((resolve, reject) => { ... })' :  
↳ Create a new Promise
- '.then (<onFulfilled>)' :-  
↳ Handle successful promise resolution.
- '.catch (<onRejected>)' :-  
↳ Handle promise rejection.

⑦ Express.js :- Web application framework

- 'const express = require ('express')' :  
↳ Import Express.
- 'const app = express ()' :-  
↳ Create an Express app.
- 'app.get (<route>, <callback>)' :-  
↳ Handle GET requests.
- 'app.post (<route>, <callback>)' :-  
↳ Handle POST requests.