**24-07-2021**

**Day 1**

**Phase 3**

**Node JS**

Node JS Concept.

Basic programming using Node JS

Node JS Modules core

fs module

util module

taking value through keyboards using different external modules.

url module

http module

client and server application

setting the data for node js application using get and post methods.

External module

Express modules

Rest API

Testing rest API get, post, put and delete methods using plugins or cURL.

Database

Mongo DB : No SQL Database

Basic query and adv query

Connecting database through node js using mongo db as well as mongoose modules.

Using Express module creating MVC project with mongoose module to connect the database.

Connecting Angular Application to Express MVC module

MEAN Stack

CRUD Operation

Angular 🡪 Express ->Mongoose 🡪 Mongo DB

Socket programming concept using Node JS

**Node JS :** Node JS is a run time environment for the JavaScript library or JavaScript framework.

Every browser contains JavaScript plugin to check or run the client side JavaScript programs.

In Client Side JavaScript program we can use DOM and BOM.

BOM : Browser object model

DOM :document object model

Client side JavaScript program we have to run on Browser using internal JS file or external JS file.

Front end Team back end team

HTML/CSS/JavaScript JEE-Servlet/JSP/EJB

jQuery and bootstrap Spring boot

Asp.net

Php

CGI

**Using JavaScript creating dynamic web page, receive the request and response from a client application, storing data in file, connecting database, creating rest API etc.**

**Node JS goal is to provide an easy way to build scalable networking programs.**

**Node JS provide non blocking IO Operation through networking environment.**

**Because JavaScript support callback function and asynchronous operation.**

**XMLHttpRequest and ActiveXObject**

**Using API if we send the Data we have wait for the acknowledgement may be success or failure.**

**Node JS is an open source, platform independent runtime environment for JavaScript programs.**

**It provide an event driven architecture and non blocking IO operation.**

**Node JS program are executed using V8 JavaScript engine. Chrome browser use V8 engine.**

**Node JS**

**Node JS is not a framework**

**Node JS is not a good for beginner**

**Web Application or Server or REST API.**

**Not a multi thread**

**Node JS = Runtime environment + JavaScript library /JavaScript framework + core module or external modules.**

**Node JS**

**Running Node JS JavaScript using command prompt**

**To install new version of node JS**

**npm install node**

**then set the path may be temporary or permanent.**

**For temporary set path=URL;.;%PATH%**

**set path=C:\Users\91990\node\_modules\.bin;.;%PATH%**

**Node JS provide its own terminal to run the Node JS Programs.**

**REPL : Read Eval Print Loop**

**In Node JS program we can’t use BOM and DOM.**

**console**

**Node JS provider global object ie console which help to display the output in REPL terminal or normal console.**

**process :**

**Node JS Provide another pre-defined global object ie process. This object is use to find the processor details.**

**Node JS Modules**

**Node JS modules is a simple or complex functionality organized in a single or multiple JavaScript files which can be reused through the Node JS application like a Package in Java technologies.**

**Using Node JS module we can achieve separation of concern.**

**Types of modules**

1. **Core Module**
2. **Local module (user-defined modules)**
3. **Third party modules**

**Core Modules : All node js application by default provide few set of modules which come under the category core modules**

1. **fs : File system**
2. **os : Operating system**
3. **net : networking : socket programming**
4. **url : if we want to extract the data from URL**
5. **http : to create basic web application.**
6. **https : to create secure basic web application.**

**etc**

**Fs Module :**

**Node JS provide pre-defined Core module ie fs module which help to do file handling program may be synchronously as well as asynchronously.**

**Syntax to load the module in ES5 style**

**var/let variableName = require(“moduleName”)**

**var fs = require(“fs”);**

**25-07-2021**

**Day 2**

**Phase 3**

**Reading the value through console in node js**

**readline : readline is external module which help to receive the value asynchronously.**

**Syntax to install the external module**

**Globally**

**npm install –g moduleName**

**or**

**locally**

**npm install moduleName**

**package.json file**

**This file hold the configuration details about node js projects.**

**If we install the module locally the package.json file hold details about that modules.**

**readline-sync : This module help use to read the value synchronously.**

**url module : This module is use to find the URL details like port number, localhost, query params etc.**

let urlInfo = "http://localhost:9090/welcome?name=ravi&age=21";

let url = require("url");

//console.log(urlInfo);

let urlObj = url.parse(urlInfo,true);

console.log(urlObj);

console.log(urlObj.protocol)

console.log(urlObj.hostname)

console.log(urlObj.port)

console.log(urlObj.pathname)

console.log(urlObj.query)   // string consider if parse second parameter is false

let obj = urlObj.query;     // it consider as reference if parser second parameter is true

console.log("Name is "+obj.name)

console.log("Age is "+obj.age)

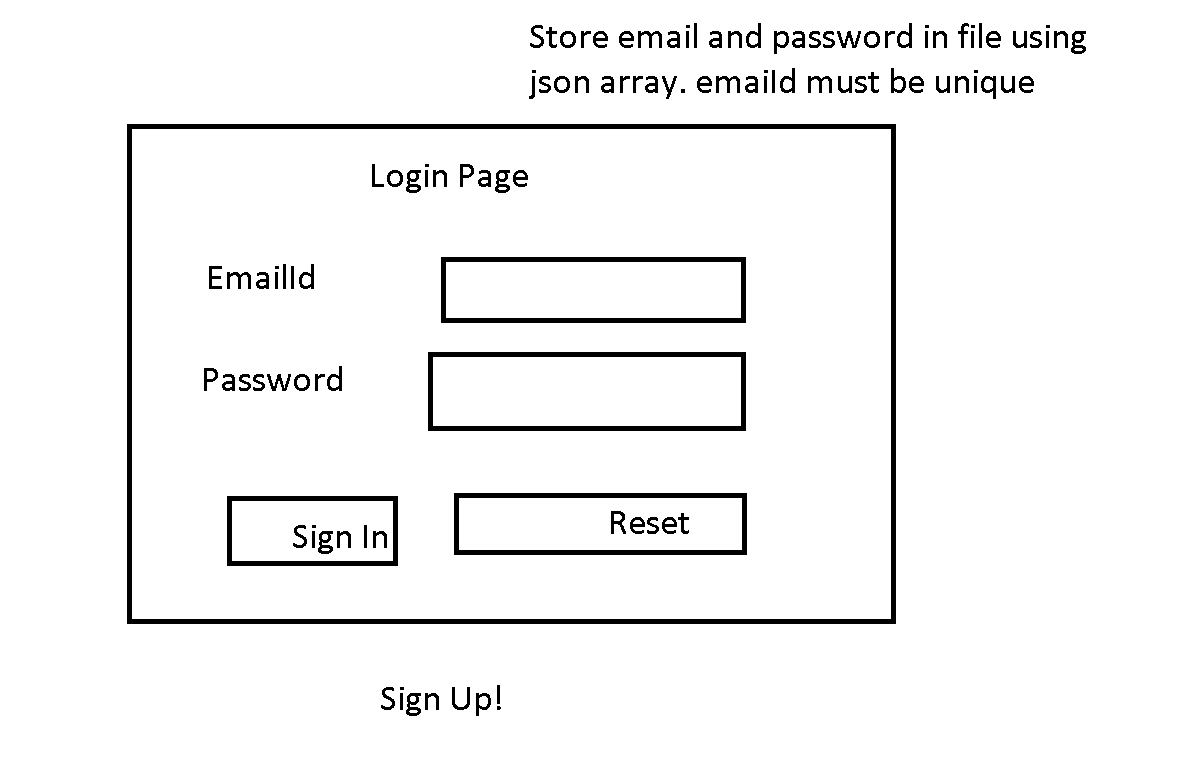
**http module :**

**Node js provide pre-defined module ie http module. Using http module we can create own server using Node JS programs.**

**Jsp, asp, php etc.**

**Tomcat, welogic, jboss, IIS etc**

**http and fs modules**

****

**Difference between Normal Server and Node JS server**

**Non node js server are by default thread base**

**Tomcat, WebLogic, IIS Server etc.**

**Tomcat / IIServer :**

**1,000 : this server can response 1,000 client concurrently.**

**1,001 client if send the request.**

**Online Booking**

**class Booking {**

**int avl=1;**

**}**

**One object created and inside that multiple client equal to multiple thread created.**

**3 client send request.**

**In Thread base server request can be block or lock.**

**Node JS Server provide even driven architecture framework.**

**JavaScript is single thread.**

**We can do or achieve multiple task execute independently using callback and asynchronous communication.**