Assignment 1: AngularJS

1. Create simple JavaScript application which will authenticate user's Login ID and Password from the list using JavaScript.
2. Create form to add, Update and delete the users details and display the users details in proper format using JavaScript
3. Create an angularJS application which will display user full name using two-way data binding method.
4. Create an angularJS application to display the arithmetic operation with two operands in proper format.
5. Create an angular application to customized the style of the user message according to user inputs like background colour, text colour.
6. Create an angular application to display the total bill from the user selected item and item quantity in proper format.
7. Create an angular application to display the subject list also, search the subject according to user inputted list index. [ng-init, ng-repeat]
8. Build a Currency Converter: Create a currency converter application using AngularJS.
9. Create application module and controller module to display message from the controller using MVC architecture.
10. Create student application using MVC pattern and display the student information in a view using proper format.
    1. Create controller which store model data as studentInfo (sName, sID,course,age)
    2. Display the student list in proper format.
    3. Modify the student list with three subject's marks
    4. Create controller function to calculate result of students.
        - Calculate total
        - Calculate Percentage
        - Calculate grade as (AA, AB, BB, BC, CC, CD, DD, FF)
        - Also check if the student has marks < 50 then do not display total and percentage)
    5. Add the display result button to display the individual students result in proper format.
    6. Search the student result as per the student ID selected from dropdown

menu

11. Create a controller for product information and display the details of product in proper format. Apply the below functionality.

    - Display product details (Product ID, Product Name and Image)
    - Apply View Details button to display other details
    - Add quantity textbox for each product
    - Display total price of the product.
    - Add "Add to Cart" button to add the product in to cart with quantity and total.

    Note: consider the details like productID, productName, productCategory, productDescription, manufactureDate, price and product Image)

12. Use the above details of product to apply the filters and perform below task on the details:

    - Apply search using filter which will filter the data according to the user inputted text in the search box.
    - Apply Sort filter to sort the details according to the Product Name. (Use radio button to sort ascending and descending)
    - Apply Next and Previous button to display details (At a time user can view 3 products on the web page)

Assignment 2: NodeJS & ExpressJS

13. Write a Node.js program that outputs "Hello, Node.js!" to the console.

14. Develop a basic HTTP server that responds with "Welcome to the NodeJS App" to any request.

15. Develop a basic HTTP server that responds with "Welcome to Home/Student/Faculty" according to request sent in the URL with different route - /home, /student, /faculty.

16. Develop a basic HTTP server that responds with index.html according to request sent in the URL using HTTP and FS module.

17. Develop a basic HTTP server that responds with .html file according to request sent in the URL using HTTP, FS and URL module.

18. Develop a basic HTTP server that responds with message data which was sent by the local module of NodeJS.

19. Develop local module (MathUtility) to perform arithmetic operation using functions. Export the module in NodeJS file and perform the arithmetic operations. Also,

perform the same task and display the result of arithmetic operations as response of HTTP server.

20. Create a NodeJS application that reads data from a JSON file, add new data to file, modifies it, delete it and saves it back to the file.

21. Develop a basic ExpressJS server that responds with index.html according to request sent in the URL using GET method.

22. Build a simple RESTful API server that handles CRUD operations for a resource (e.g., todoTask).

Assignment 3: MEAN Stack Application

23. Develop expressJS server that handle the CRUD operations for a resource (tasks / products)

24. Create collection for tasks in MongoDB and display the data of task in angularJS application controller using ExpressJS.

25. Create angularJS application which execute through the ExpressJS server to manipulate student's data using Mongoose module of nodeJS.