

OST Lab Exercise Algorithms

EX.1 (Write a JavaScript Program to Convert Celcius Value to Faranheit Value in JavaScript):

1. Create two input fields for Celsius and Fahrenheit values.
2. Add an event listener to each input field that listens for a keyup event.
3. Define a function that takes a degree parameter (either "C" or "F") and converts the value in the corresponding input field to the other unit of measurement.
4. In the function, use an if-else statement to determine which conversion formula to use based on the degree parameter.
5. Update the value of the other input field with the converted value.

EX.2 (Write a JavaScript program to implement `getElementsByClassName()` method in JavaScript):

- *Add a class name to the element you want to retrieve.
- *Create a button with an onclick event that calls a function.
- *In the function, use the `getElementsByClassName()` method to retrieve all elements with the specified class name.
- *Access the first element in the returned array using its index (0).
- *Use the `innerHTML` property to retrieve the value of the element.

EX.3 (Install Node.js)

Step 1- Visit the official Node.js website i.e) <https://nodejs.org/en/download/> and download the .msi file according to your system environment (32-bit & 64-bit).

*Step 2- Double click on the .msi installer. The Node.js Setup wizard will open. Select "Next".

*Step 3- Check "I accept the terms in the License Agreement" Select "Next"

*Step 4- Set the Destination Folder where you want to install Node.js & Select "Next"

*Step 5- Select Next again and click Install

*Step 6- Run the command 'node -v' to check if node.js is properly installed or not.

EX.4 (Create Web Server)

- * Create a file `httpWebServer.js`.
- * Import the 'http' module and assign it to a variable.
- * Referring that variable call `createServer()` method and define a callback function inside it.
- * Referring response parameter (res) call `write()` method to write some content and `end()` method to end.
- * Finally, listen at port 8080.
- * Run the file using node and Search as 'localhost:8080' in web browser for result.

EX.5 (NPM)

- * Create a empty folder and get into that directory in Terminal.
- * Run the command 'npm init' to initialize the creation of package.json file.
- * Provide the necessary informations to create the file.
- * Run the command 'npm i express' to install Express framework.
- * Check the folder structure to notice if a folder called 'node_modules' is created.

EX.6 (Retrieve form values using express)

- * Create a file 'login.html' which includes a form with two input fields.
- * Create a file 'index.html' where we just have redirect button to login.html.
- * Using 'npm init' create package.json file and run the command 'npm i express body-parser' to install Express.
- * Create a file called server.js and include Express and Body-Parser.
- * Set up routes for Get requests.
- * Set up a route for POST requests.
- * Assign a port and use the command 'node server.js' to run our program.

EX.7 (Highlight menu using Angular.js)

- * Create two files 'index.html' and 'style.css'.
- * Link index file to styles file using a link tag.
- * Add a script tag inside index file and specify angular.js url in the src attribute.
- * Specify body as 'ng-app' and add 3 anchor tags inside a nav tag.
- * Specify nav tag class as active and ng-click as '\$event.preventDefault()'.
- * Specify class names for anchor tags save the state in active variable using ng-click.
- * Add styling using CSS to highlight the selected menu.

EX.8 (Manipulate web page using JQuery)

- * Include the jQuery library in your HTML file using the <script> tag.
- * Select the HTML element you want to manipulate using jQuery selectors.
- * Use jQuery methods to manipulate the selected element's content, attributes, or CSS properties.
- * Add event listeners to HTML elements using jQuery's event handling methods.
- * Test your code and make sure it works as expected.

EX.9 (Attach different events to the elements of a web page using jQuery)

- * Include the jQuery library in the head of the HTML document using the script tag.
- * Define an h1 element with the text "Hello World!".
- * Define a script tag that contains a function that is executed when the document is ready.
- * Attach three different events to the h1 element using jQuery:
 1. Hide the h1 element when it is clicked.
 2. Change the background color of the h1 element when it is hovered over.
 3. Change the background color of the h1 element when it is clicked and held down.
- * Run the program.

EX.10 (Provide different effects using jQuery)

- * Include the jQuery library in your HTML file using the <script> tag.
- * Define a p element with the text "This is a paragraph."
- * Define two button elements with classes of "btn1" and "btn2".
- * Define a script tag that contains a function that is executed when the document is ready.
- * Attach two different events to the button elements using jQuery:
 1. Add a slideUp event to the p element when the "btn1" button is clicked.
 2. Add a slideDown event to the p element when the "btn2" button is clicked.
- * Run the program.