Kathmandu University - Radiant College

JavaScript and ReactJS Training Syllabus

JavaScript

Objectives:

- Understand the fundamentals of JavaScript.
- Learn to manipulate the DOM.
- Master asynchronous programming with Promises and async/await.
- Gain proficiency in ES6+ features.

Modules:

1. **Introduction to JavaScript**
   - History and evolution of JavaScript
   - Role of JavaScript in web development
   - Setting up the development environment (Node.js, npm)
2. **Basic Syntax and Operators**
   - Basic syntax and structure
   - Arithmetic, comparison, logical, and assignment operators
3. **Data Types and Variables**
   - Primitive data types (String, Number, Boolean, Null, Undefined, Symbol)
   - Reference data types (Object, Array, Function)
   - Variable declarations (var, let, const)
   - Type conversion and coercion
4. **Control Structures**
   - Conditional statements (if, else, switch)
   - Loops (for, while, do-while)
   - Break and continue statements
5. **Functions**
   - Function declarations and expressions
   - Arrow functions
   - Higher-order functions(advanced)
   - Function scope and closures(advanced)
   - Immediately Invoked Function Expressions (IIFE)(advanced)
6. **Objects and Arrays**
   - Creating and manipulating objects(advanced)
   - Object methods and properties(advanced)
   - Array methods (map, filter, reduce, forEach)
   - Destructuring objects and arrays

7. **DOM Manipulation**
   - Selecting and modifying DOM elements
   - Event handling (click, submit, etc.)
   - Creating and removing elements dynamically
8. **Asynchronous JavaScript**
   - Callbacks(advanced)
   - Promises
   - async/await
   - Fetch API for making HTTP requests
9. **Error Handling**
   - try...catch...finally
   - Throwing custom errors
   - Debugging techniques
10. **ES6+ Features**
    - Template literals
    - Destructuring assignment
    - Spread and rest operators
    - Modules (import/export)
    - Classes and inheritance(advanced)
    - Generators and iterators
11. **Advanced Topics**
    - Prototypes and inheritance(advanced)
    - Closures and scope(advanced)
    - Event delegation(advanced)
    - Memory management and garbage collection(advanced)
12. **JavaScript in the Browser**
    - Browser Object Model (BOM)
    - Document Object Model (DOM)
    - Event propagation (bubbling and capturing)
    - Local storage and session storage
13. **JavaScript Frameworks and Libraries**
    - Introduction to popular frameworks (React, Angular, Vue)
    - Using libraries like jQuery, Lodash(advanced)
14. **Testing and Debugging**
    - Writing unit tests with frameworks like Jest
    - Debugging with browser developer tools

Project Work:

To-Do List Application:

- Create a simple to-do list app that allows users to add, edit, and delete tasks.

Quiz App

- Create a quiz application with multiple-choice questions.

- Track and display the user's score at the end.

Tip Calculator

- Calculate the tip amount based on the bill total and selected tip percentage.
- Split the bill among multiple people.

BMI Calculator

- Calculate Body Mass Index (BMI) based on user input (height and weight).
- Display the BMI category (underweight, normal, overweight, obese).

Digital Clock

- Display the current time in a digital format.
- Include options for 12-hour and 24-hour formats.

Image Slider

- Create an image carousel that automatically transitions between images.
- Include navigation controls for manual transitions.

Memory Game

- Build a card matching game where users flip cards to find pairs.
- Track the number of moves and time taken to complete the game.

Weather App

- Fetch and display weather data from an API.
- Show current weather, forecast, and weather conditions for different locations.

Currency Converter

- Convert amounts between different currencies using real-time exchange rates.
- Fetch exchange rates from an API.

Recipe Finder

- Search for recipes based on ingredients.
- Display recipe details, including ingredients, steps, and nutritional information.

Task Management Tool

- Create, edit, and delete tasks.
- Organize tasks by categories and due dates.

Chat Application

- Real-time messaging using WebSockets.
- User authentication and chat rooms.

Movie Database

- Search and display information about movies.
- User ratings and reviews.

Expense Tracker

- Track income and expenses.
- Visualize data with charts and graphs.

Portfolio Website

- Showcase your projects and skills.
- Include sections like About, Projects, and Contact.

Social Media Dashboard

- Display social media metrics and analytics.
- Integrate with APIs from platforms like Twitter, Facebook, and Instagram

React Training Syllabus

Objectives:

- Understand the core concepts of React.
- Build reusable components.
- Manage state and props effectively.
- Implement routing and state management.

Modules:

1. **Introduction to React**
   - Setting up the environment (Node.js, npm, Create React App)
   - JSX syntax
2. **Components and Props**
   - Functional vs. class components
   - Passing and using props
3. **State and Lifecycle**
   - Managing state in components
   - Lifecycle methods
4. **Event Handling**
   - Handling events in React

- - Synthetic events
5. **Forms and Validation**
   - Controlled components
   - Form submission and validation
6. **Conditional Rendering**
   - Using if statements, ternary operators, and logical &&
7. **Lists and Keys**
   - Rendering lists
   - Importance of keys
8. **Styling Components**
   - CSS, CSS Modules, CSS-in-JS
9. **React Router**
   - Setting up routes
   - Navigation and parameters
10. **State Management with Redux**
    - Setting up Redux
    - Actions and reducers
11. **Hooks**
    - useState, useEffect, useContext
    - Custom hooks
12. **Asynchronous Operations**
    - Fetch API
    - Handling async/await in React
13. **Error Handling and Optimization**
    - Error boundaries
    - Performance optimization
14. **Deployment**
    - Deploying to Netlify, Vercel, AWS

Sample Projects:

1. **Weather App**
   - Fetch weather data from an API.
   - Display current weather, forecast, and weather conditions for different locations.
2. **Recipe Finder**
   - Search for recipes based on ingredients.
   - Display recipe details, including ingredients, steps, and nutritional information.
3. **Expense Tracker**
   - Track income and expenses.
   - Visualize data with charts and graphs.
4. **Chat Application**
   - Real-time messaging using WebSockets.
   - User authentication and chat rooms.
5. **Blog Platform**
   - Create, edit, and delete blog posts.
   - User authentication and comments section.

6. **Portfolio Website**
   - Showcase your projects and skills.
   - Include sections like About, Projects, and Contact.
7. **Movie Database**
   - Search and display information about movies.
   - User ratings and reviews.
8. **E-commerce Store**
   - Product listings, shopping cart, and checkout process.
   - User authentication and order history.
9. **Task Management Tool**
   - Create, edit, and delete tasks.
   - Organize tasks by categories and due dates.
10. **Social Media Dashboard**
    - Display social media metrics and analytics.
    - Integrate with APIs from platforms like Twitter, Facebook, and Instagram.

These projects will help you practice various aspects of React, such as state management, API integration, and user authentication. Happy coding! If you need more details on any specific project, feel free to ask your trainer.