1)Write an asynchronous file reading node Js program
Ans:-

```
const fs = require('fs').promises;

async function readFileAsync(filePath) {
    try {
        const data = await fs.readFile(filePath, 'utf8');
        console.log('File Contents:');
        console.log(data);
    } catch (error) {
        console.error('Error reading file:', error);
    }
}

// Specify the path to your file
const filePath = './example.txt';
readFileAsync(filePath);
```

2) Explain React JSX with suitable react examples such as rendering the greeting message “Hello! Welcome to React”
Ans:-

```
import React from 'react';

function Greeting() {
    return (
        <h1>Hello! Welcome to React</h1>
    );
}

function App() {
    return (
        <div>
            <Greeting />
        </div>
    );
}

export default App;
```

3)Write a Javascript to accept a number from the user and check if it is even or not.
Ans:-

```
// Function to check if a number is even
function checkEvenOrOdd() {
    // Prompt the user for input
    const input = prompt("Please enter a number:");

    // Convert the input to a number
    const number = parseFloat(input);

    // Check if the input is a valid number
    if (isNaN(number)) {
        alert("That's not a valid number. Please try again.");
```

```
        return;
    }

    // Check if the number is even
    if (number % 2 === 0) {
        alert(`${number} is an even number.`);
    } else {
        alert(`${number} is an odd number.`);
    }
}

// Call the function
checkEvenOrOdd();
```

4)Demonstrate its use by writing the code which create a simple text file with the data provided by the user.

Ans:-

```
const fs = require('fs');
const readline = require('readline');

// Create an interface for reading input
const rl = readline.createInterface({
    input: process.stdin,
    output: process.stdout
});

// Function to prompt user for input and write to file
function createFileWithUserInput() {
    rl.question('Please enter the content you want to write to the file: ', (userInput) => {
        // Define the file name
        const fileName = 'output.txt';

        // Write user input to the file
        fs.writeFile(fileName, userInput, (err) => {
            if (err) {
                console.error('Error writing to file:', err);
            } else {
                console.log(`Successfully wrote to ${fileName}`);
            }
            // Close the readline interface
            rl.close();
        });
    });
}

// Call the function
createFileWithUserInput();
```

5)Write a code making use of HOOKS useState function that display the number of times button nammed “CLICK” is clicked

Ans:-

```
import React, { useState } from 'react';

function App() {
   // Declare a state variable `count` with initial value 0
   const [count, setCount] = useState(0);

   // Function to handle button click
   const handleClick = () => {
      setCount(count + 1); // Increment the count
   };

   return (
      <div style={{ textAlign: 'center', marginTop: '50px' }}>
         <h1>Button Click Counter</h1>
         <p>You have clicked the button {count} times.</p>
         <button onClick={handleClick}>CLICK</button>
      </div>
   );
}

export default App;
```

6)Write a class component with constructure to update state in react using state and props.
Ans:-

```
import React, { Component } from 'react';

class Greeting extends Component {
   constructor(props) {
      super(props);
      // Initialize state
      this.state = {
         name: this.props.name || 'Guest'
      };
   }

   // Method to update state based on input
   handleChange = (event) => {
      this.setState({ name: event.target.value });
   };

   render() {
      return (
         <div style={{ textAlign: 'center', marginTop: '50px' }}>
            <h1>Hello, {this.state.name}!</h1>
            <input
               type="text"
               value={this.state.name}
               onChange={this.handleChange}
               placeholder="Enter your name"
            />
         </div>
      );
```

```
    }
}

// Main App component
class App extends Component {
    render() {
        return (
            <div>
                <Greeting name="User" />
            </div>
        );
    }
}

export default App;
```

7)Write a class component without constructure to update state in react using state and props
Ans:-

```
import React, { Component } from 'react';

class Greeting extends Component {
    // Initialize state directly as a class property
    state = {
        name: this.props.name || 'Guest'
    };

    // Method to update state based on input
    handleChange = (event) => {
        this.setState({ name: event.target.value });
    };

    render() {
        return (
            <div style={{ textAlign: 'center', marginTop: '50px' }}>
                <h1>Hello, {this.state.name}!</h1>
                <input
                    type="text"
                    value={this.state.name}
                    onChange={this.handleChange}
                    placeholder="Enter your name"
                />
            </div>
        );
    }
}

// Main App component
class App extends Component {
    render() {
        return (
            <div>
                <Greeting name="User" />
```

```
            </div>
        );
    }
}

export default App;
```

8)Write a program in react to update state and props with functional component.
Ans:-

```
import React, { useState } from 'react';

// Functional component that accepts props
const Greeting = ({ initialName }) => {
    // State to hold the name
    const [name, setName] = useState(initialName || 'Guest');

    // Function to handle input change
    const handleChange = (event) => {
        setName(event.target.value);
    };

    return (
        <div style={{ textAlign: 'center', marginTop: '50px' }}>
            <h1>Hello, {name}!</h1>
            <input
                type="text"
                value={name}
                onChange={handleChange}
                placeholder="Enter your name"
            />
        </div>
    );
};

// Main App component
const App = () => {
    return (
        <div>
            <Greeting initialName="User" />
        </div>
    );
};

export default App;
```

9) Create one class components “Car” in React and invoke it using index.js it
Ans:-

```
// src/Car.js
import React, { Component } from 'react';

class Car extends Component {
    render() {
```

```
        const { make, model, year } = this.props; // Destructure props
        return (
            <div style={{ textAlign: 'center', marginTop: '50px' }}>
                <h1>Car Details</h1>
                <p>Make: {make}</p>
                <p>Model: {model}</p>
                <p>Year: {year}</p>
            </div>
        );
    }
}

export default Car;

// src/index.js
import React from 'react';
import ReactDOM from 'react-dom';
import App from './App';
import Car from './Car'; // Import the Car component

// Create a simple App component to render Car
const App = () => {
    return (
        <div>
            <Car make="Toyota" model="Camry" year="2020" />
        </div>
    );
};

// Render the App component
ReactDOM.render(
    <React.StrictMode>
        <App />
    </React.StrictMode>,
    document.getElementById('root')
);
```

10) Write acode making use of React Hooks that displays four buttons namely ,”Red”,”Blue”,”Green”,”Yellow”. On clicking any of these buttons,the code display the message that you have selected that particular color.

Ans:-

```
import React, { useState } from 'react';

const App = () => {
    // State to hold the selected color message
    const [selectedColor, setSelectedColor] = useState('');

    // Function to handle button click
    const handleColorClick = (color) => {
        setSelectedColor(`You have selected ${color}!`);
    };
```

```
  return (
    <div style={{ textAlign: 'center', marginTop: '50px' }}>
      <h1>Select a Color</h1>
      <div>
        <button onClick={() => handleColorClick('Red')}>Red</button>
        <button onClick={() => handleColorClick('Blue')}>Blue</button>
        <button onClick={() => handleColorClick('Green')}>Green</button>
        <button onClick={() => handleColorClick('Yellow')}>Yellow</button>
      </div>
      {selectedColor && <p>{selectedColor}</p>}
    </div>
  );
};

export default App;
```

11) Write a JavasScript program to display a digital clock on web page.
Ans:-

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Digital Clock</title>
  <style>
    body {
      display: flex;
      justify-content: center;
      align-items: center;
      height: 100vh;
      background-color: #282c34;
      color: white;
      font-family: 'Arial', sans-serif;
      font-size: 48px;
    }
  </style>
</head>
<body>
  <div id="clock"></div>

  <script>
    function updateClock() {
      const now = new Date();
      const hours = String(now.getHours()).padStart(2, '0');
      const minutes = String(now.getMinutes()).padStart(2, '0');
      const seconds = String(now.getSeconds()).padStart(2, '0');

      const timeString = `${hours}:${minutes}:${seconds}`;
      document.getElementById('clock').textContent = timeString;
    }

    // Update the clock every second
```

```
        setInterval(updateClock, 1000);

        // Initialize the clock display
        updateClock();
    </script>
</body>
</html>
```

12)Write a Javascript Program to display three radio buttons on the web page, namely,”Red” “Blue ” and “Green” . Selecting any button changes the background color as per the name of the button.

Ans:-

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Color Selector</title>
    <style>
        body {
            display: flex;
            justify-content: center;
            align-items: center;
            height: 100vh;
            transition: background-color 0.5s; /* Smooth transition */
        }
        label {
            margin: 10px;
            font-size: 24px;
        }
    </style>
</head>
<body>
    <div>
        <label>
            <input type="radio" name="color" value="Red" onclick="changeColor('Red')"> Red
        </label>
        <label>
            <input type="radio" name="color" value="Blue" onclick="changeColor('Blue')"> Blue
        </label>
        <label>
            <input type="radio" name="color" value="Green" onclick="changeColor('Green')"> Green
        </label>
    </div>

    <script>
        function changeColor(color) {
            document.body.style.backgroundColor = color; // Change background color
        }
    </script>
</body>
</html>
```

13) Create a web page using HTML and JavaScript where an image moves across the screen form left to right continuously.
Ans:-

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Moving Image</title>
    <style>
        body {
            margin: 0;
            overflow: hidden; /* Prevent scrollbar from appearing */
        }
        #movingImage {
            position: absolute;
            width: 100px; /* Set the width of the image */
            height: auto; /* Maintain aspect ratio */
        }
    </style>
</head>
<body>
    <img id="movingImage" src="https://via.placeholder.com/100" alt="Moving Image">

    <script>
        const image = document.getElementById('movingImage');
        let position = 0; // Starting position

        function moveImage() {
            position += 2; // Increment position
            image.style.left = position + 'px'; // Set the new position

            // Reset position when the image goes off the screen
            if (position > window.innerWidth) {
                position = -100; // Move back to the left side (off-screen)
            }

            requestAnimationFrame(moveImage); // Call the function again
        }

        moveImage(); // Start moving the image
    </script>
</body>
</html>
```

14) Write a JavasScript code to change the background color of the web page to red color if the button named “RED” is clicked and to green color if the button named “GREEN” is clicked.
Ans:-

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
```

```
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Background Color Changer</title>
    <style>
        body {
            display: flex;
            justify-content: center;
            align-items: center;
            height: 100vh;
            transition: background-color 0.5s; /* Smooth transition */
        }
        button {
            margin: 10px;
            padding: 15px 30px;
            font-size: 16px;
            cursor: pointer;
        }
    </style>
</head>
<body>
    <button onclick="changeColor('red')">RED</button>
    <button onclick="changeColor('green')">GREEN</button>

    <script>
        function changeColor(color) {
            document.body.style.backgroundColor = color; // Change background color
        }
    </script>
</body>
</html>
```

15) Write a JavasScript code to set a cookie in the user’s computer

Ans:-

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Set Cookie Example</title>
    <style>
        body {
            display: flex;
            justify-content: center;
            align-items: center;
            height: 100vh;
            flex-direction: column;
        }
        button {
            padding: 10px 20px;
            font-size: 16px;
            cursor: pointer;
        }
    </style>
```

```
</head>
<body>
   <button onclick="setCookie('username', 'JohnDoe', 7)">Set Cookie</button>
   <button onclick="alert(getCookie('username'))">Get Cookie</button>

   <script>
      function setCookie(name, value, days) {
         let expires = "";
         if (days) {
            const date = new Date();
            date.setTime(date.getTime() + (days * 24 * 60 * 60 * 1000));
            expires = "; expires=" + date.toUTCString();
         }
         document.cookie = name + "=" + (value || "") + expires + "; path=/";
         alert("Cookie set: " + name + "=" + value);
      }

      function getCookie(name) {
         const nameEQ = name + "=";
         const ca = document.cookie.split(';');
         for (let i = 0; i < ca.length; i++) {
            let c = ca[i];
            while (c.charAt(0) === ' ') c = c.substring(1, c.length);
            if (c.indexOf(nameEQ) === 0) return c.substring(nameEQ.length, c.length);
         }
         return null; // Return null if cookie not found
      }
   </script>
</body>
</html>
```

16) Write a javascript to accept two numbers and display their sum using pop up box.
Ans:-

```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>Sum of Two Numbers</title>
   <style>
      body {
         display: flex;
         justify-content: center;
         align-items: center;
         height: 100vh;
         flex-direction: column;
      }
      button {
         padding: 10px 20px;
         font-size: 16px;
         cursor: pointer;
      }
```

```
    </style>
</head>
<body>
    <button onclick="calculateSum()">Calculate Sum</button>

    <script>
        function calculateSum() {
            // Prompt the user for two numbers
            const num1 = parseFloat(prompt("Enter the first number:"));
            const num2 = parseFloat(prompt("Enter the second number:"));

            // Check if the inputs are valid numbers
            if (isNaN(num1) || isNaN(num2)) {
                alert("Please enter valid numbers.");
            } else {
                const sum = num1 + num2; // Calculate the sum
                alert("The sum is: " + sum); // Display the sum in a popup
            }
        }
    </script>
</body>
</html>
```

17) Write a Javascript Program that changes the background color of page by refreshing the page every 2 seconds.

Ans:-

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Background Color Changer</title>
    <style>
        body {
            display: flex;
            justify-content: center;
            align-items: center;
            height: 100vh;
            transition: background-color 1s; /* Smooth transition */
        }
    </style>
</head>
<body>
    <h1 style="color: white;">Background Color Changes Every 2 Seconds</h1>

    <script>
        function getRandomColor() {
            const letters = '0123456789ABCDEF';
            let color = '#';
            for (let i = 0; i < 6; i++) {
                color += letters[Math.floor(Math.random() * 16)];
            }
```

```
            return color;
        }

        function changeBackgroundColor() {
            document.body.style.backgroundColor = getRandomColor(); // Change background color
        }

        setInterval(changeBackgroundColor, 2000); // Change color every 2 seconds
    </script>
</body>
</html>
```

18) i)Write the code to process online Alumni information for your college.Create a form to get a nane,date of birth and email id. Use check boxes for taking hobbies and radio buttons for selecting branch.Write JavaScript code to validate the following. -User has to fill all the fields prior to the form submission -Valid email id (@ and .) -Age validation using DOB(>=22 years) -Password length should not less than 6.

Ans:-

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Alumni Information Form</title>
    <style>
        body {
            display: flex;
            justify-content: center;
            align-items: center;
            height: 100vh;
            flex-direction: column;
            font-family: Arial, sans-serif;
        }
        form {
            display: flex;
            flex-direction: column;
            width: 300px;
        }
        input, select {
            margin: 10px 0;
            padding: 10px;
            font-size: 16px;
        }
        button {
            padding: 10px;
            font-size: 16px;
            cursor: pointer;
        }
    </style>
</head>
<body>
    <h1>Alumni Information Form</h1>
```

```
<form id="alumniForm">
    <input type="text" id="name" placeholder="Name" required>
    <input type="date" id="dob" required>
    <input type="email" id="email" placeholder="Email" required>
    <div>
        <strong>Select Branch:</strong>
        <label><input type="radio" name="branch" value="CSE" required> CSE</label>
        <label><input type="radio" name="branch" value="ECE" required> ECE</label>
        <label><input type="radio" name="branch" value="ME" required> ME</label>
    </div>
    <div>
        <strong>Select Hobbies:</strong><br>
        <label><input type="checkbox" name="hobbies" value="Reading"> Reading</label>
        <label><input type="checkbox" name="hobbies" value="Traveling"> Traveling</label>
        <label><input type="checkbox" name="hobbies" value="Sports"> Sports</label>
    </div>
    <input type="password" id="password" placeholder="Password" required>
    <button type="submit">Submit</button>
</form>

<script>
    document.getElementById('alumniForm').addEventListener('submit', function(event) {
        event.preventDefault(); // Prevent form submission for validation

        const name = document.getElementById('name').value;
        const dob = document.getElementById('dob').value;
        const email = document.getElementById('email').value;
        const password = document.getElementById('password').value;

        // Validate all fields
        if (!name || !dob || !email || !password) {
            alert("Please fill all fields.");
            return;
        }

        // Validate email format
        const emailPattern = /^[^\s@]+@[^\s@]+\.[^\s@]+$/;
        if (!emailPattern.test(email)) {
            alert("Please enter a valid email address.");
            return;
        }

        // Validate age (22 years or older)
        const birthDate = new Date(dob);
        const today = new Date();
        const age = today.getFullYear() - birthDate.getFullYear();
        const monthDifference = today.getMonth() - birthDate.getMonth();
        if (age < 22 || (age === 22 && monthDifference < 0)) {
            alert("You must be at least 22 years old.");
            return;
        }
```

```
            // Validate password length
            if (password.length < 6) {
                alert("Password must be at least 6 characters long.");
                return;
            }

            // If all validations pass, you can process the form data
            alert("Form submitted successfully!");
            // Here you can add code to send the form data to the server
        });
    </script>
</body>
</html>
```

19) Configure VLAN using CISCO packet tracer.
Ans:- Exp 8 of WCNL

20) Configure NAT using CISCO packet tracer
Ans:- Exp 9 of WCNL

21) Configure routing(static/dynamic) using CISCO packet traccer.
Ans:- Exp 7 of WCNL