Exp1

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Student Portfolio</title>

<style>

body {

font-family: Arial, sans-serif;

line-height: 1.6;

margin: 0;

padding: 0;

background-color: #f4f4f4;

}

header {

background-color: #0073e6;

color: white;

padding: 20px 0;

text-align: center;

}

nav {

background-color: #333;

padding: 10px;

}

nav a {

color: white;

text-decoration: none;

padding: 10px 20px;

display: inline-block;

}

nav a:hover {

background-color: #575757;

}

main {

display: flex;

flex-wrap: wrap;

padding: 20px;

}

section {

flex: 3;

background-color: white;

padding: 20px;

margin-right: 20px;

}

article {

margin-bottom: 20px;

}

aside {

flex: 1;

background-color: #e2e2e2;

padding: 20px;

}

footer {

text-align: center;

background-color: #333;

color: white;

padding: 10px;

margin-top: 20px;

}

img {

border-radius: 50%;

}

</style>

</head>

<body>

<header>

<h1>John Doe</h1>

<p>Student | Web Developer | Tech Enthusiast</p>

</header>

<nav>

<a href="#about">About</a>

<a href="#projects">Projects</a>

<a href="#skills">Skills</a>

<a href="#contact">Contact</a>

</nav>

<main>

<section id="about">

<article>

<h2>About Me</h2>

<img src="https://via.placeholder.com/150" alt="Profile Picture">

<p>Hello! I'm John Doe, a computer science student passionate about web development and problem-solving. I love learning new technologies and building creative projects.</p>

</article>

</section>

<section id="projects">

<article>

<h2>Projects</h2>

<ul>

<li><strong>Portfolio Website:</strong> A responsive personal portfolio built using HTML5, CSS3, and JavaScript.</li>

<li><strong>Weather App:</strong> Displays real-time weather data using an API.</li>

<li><strong>Library Management System:</strong> A mini project using Python and SQLite.</li>

</ul>

</article>

</section>

<section id="skills">

<article>

<h2>Skills</h2>

<p>HTML5, CSS3, JavaScript, Python, C++, SQL, Git, React</p>

</article>

</section>

<aside>

<h3>Contact Info</h3>

<p>Email: johndoe@example.com</p>

<p>Phone: +1 234 567 890</p>

<p>Location: New York, USA</p>

<p><a href="https://github.com/johndoe" target="\_blank">GitHub Profile</a></p>

</aside>

</main>

<footer>

<p>&copy; 2025 John Doe | All Rights Reserved</p>

</footer>

</body>

</html>

Exp2

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Placement Registration Form</title>

<style>

body {

font-family: Arial, sans-serif;

background-color: #eef2f3;

margin: 0;

padding: 0;

}

header {

background-color: #004aad;

color: white;

text-align: center;

padding: 20px 0;

}

form {

max-width: 700px;

margin: 30px auto;

background-color: white;

padding: 25px;

border-radius: 8px;

box-shadow: 0 0 10px rgba(0,0,0,0.1);

}

fieldset {

border: 2px solid #004aad;

border-radius: 5px;

margin-bottom: 20px;

padding: 20px;

}

legend {

font-weight: bold;

color: #004aad;

}

label {

display: block;

margin-top: 10px;

font-weight: bold;

}

input, select, textarea {

width: 100%;

padding: 8px;

margin-top: 5px;

border: 1px solid #ccc;

border-radius: 4px;

box-sizing: border-box;

}

input[type="radio"],

input[type="checkbox"] {

width: auto;

}

input[type="submit"],

input[type="reset"] {

background-color: #004aad;

color: white;

padding: 10px 20px;

border: none;

border-radius: 5px;

cursor: pointer;

margin-top: 10px;

}

input[type="submit"]:hover,

input[type="reset"]:hover {

background-color: #0066ff;

}

footer {

background-color: #004aad;

color: white;

text-align: center;

padding: 10px;

margin-top: 30px;

}

</style>

</head>

<body>

<header>

<h1>College Placement Registration Form</h1>

<p>Register below for the upcoming campus placements</p>

</header>

<main>

<!-- The enctype attribute is required for file uploads -->

<form enctype="multipart/form-data" method="post">

<fieldset>

<legend>Personal Details</legend>

<label for="name">Full Name:</label>

<input type="text" id="name" name="name" placeholder="Enter your full name" required>

<label for="regno">Registration Number:</label>

<input type="text" id="regno" name="regno" placeholder="Enter your college registration number" required>

<label>Gender:</label>

<input type="radio" id="male" name="gender" value="Male" required> Male

<input type="radio" id="female" name="gender" value="Female"> Female

<input type="radio" id="other" name="gender" value="Other"> Other

<label for="dob">Date of Birth:</label>

<input type="date" id="dob" name="dob" required>

</fieldset>

<fieldset>

<legend>Academic Details</legend>

<label for="department">Department:</label>

<select id="department" name="department" required>

<option value="">--Select Department--</option>

<option value="CSE">Computer Science</option>

<option value="ECE">Electronics & Communication</option>

<option value="EEE">Electrical & Electronics</option>

<option value="MECH">Mechanical</option>

<option value="CIVIL">Civil</option>

</select>

<label for="cgpa">Current CGPA:</label>

<input type="number" step="0.01" min="0" max="10" id="cgpa" name="cgpa" placeholder="e.g. 8.5" required>

<label for="year">Year of Graduation:</label>

<input type="number" id="year" name="year" placeholder="e.g. 2025" required>

</fieldset>

<fieldset>

<legend>Contact Details</legend>

<label for="email">Email ID:</label>

<input type="email" id="email" name="email" placeholder="Enter your email address" required>

<label for="phone">Mobile Number:</label>

<input type="tel" id="phone" name="phone" placeholder="Enter your phone number" required pattern="[0-9]{10}">

<label for="address">Address:</label>

<textarea id="address" name="address" rows="3" placeholder="Enter your current address"></textarea>

</fieldset>

<fieldset>

<legend>Skills & Preferences</legend>

<label>Technical Skills:</label>

<input type="checkbox" name="skills" value="C/C++"> C/C++

<input type="checkbox" name="skills" value="Java"> Java

<input type="checkbox" name="skills" value="Python"> Python

<input type="checkbox" name="skills" value="Web Development"> Web Development

<input type="checkbox" name="skills" value="AI/ML"> AI/ML

<label for="company">Preferred Company:</label>

<input type="text" id="company" name="company" placeholder="Enter your preferred company name">

</fieldset>

<fieldset>

<legend>Resume Upload</legend>

<label for="resume">Upload Resume (PDF/DOC/DOCX):</label>

<input type="file" id="resume" name="resume" accept=".pdf, .doc, .docx" required>

</fieldset>

<div style="text-align:center;">

<input type="submit" value="Submit">

<input type="reset" value="Reset">

</div>

</form>

</main>

<footer>

<p>&copy; 2025 College Placement Cell | All Rights Reserved</p>

</footer>

</body>

</html>

Exp 3

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>ShopEase - Online Store</title>

<style>

/\* Element Selector \*/

body {

font-family: 'Segoe UI', sans-serif;

margin: 0;

padding: 0;

background-color: #f5f6f8;

}

/\* ID Selector \*/

#header {

background-color: #2b2d42;

color: white;

text-align: center;

padding: 20px;

}

/\* Class Selector \*/

.nav-item {

display: inline-block;

margin: 0 15px;

}

.nav-item a {

color: white;

text-decoration: none;

font-weight: bold;

}

/\* Pseudo-class Selector \*/

.nav-item a:hover {

color: #ef233c;

text-decoration: underline;

}

/\* Group Selector \*/

h1, h2, h3 {

font-family: 'Trebuchet MS', sans-serif;

}

/\* Descendant Selector \*/

main p {

color: #333;

line-height: 1.6;

}

/\* Attribute Selector \*/

input[type="text"], input[type="email"] {

padding: 10px;

border: 1px solid #999;

border-radius: 5px;

width: 250px;

}

/\* Product Section (using Flexbox) \*/

#products {

display: flex;

justify-content: center; /\* Centers items horizontally \*/

align-items: stretch;

flex-wrap: wrap; /\* Allows wrapping for smaller screens \*/

gap: 20px; /\* Space between products \*/

padding: 20px;

}

.product {

background-color: white;

border: 1px solid #ddd;

border-radius: 5px;

text-align: center;

padding: 15px;

width: 280px;

box-shadow: 0 0 10px rgba(0,0,0,0.1);

transition: transform 0.3s;

}

.product:hover {

transform: translateY(-5px);

}

.product img {

width: 100%;

height: 180px;

border-radius: 5px;

object-fit: cover;

}

.product h3 {

color: #2b2d42;

}

.price {

color: #ef233c;

font-weight: bold;

}

/\* Pseudo-element Selector \*/

footer::before {

content: "★ ";

color: gold;

}

footer {

clear: both;

background-color: #2b2d42;

color: white;

text-align: center;

padding: 15px;

margin-top: 20px;

}

/\* Contact Form Section \*/

#contact {

text-align: center;

background-color: #edf2f4;

padding: 30px;

margin-top: 40px;

}

button {

background-color: #ef233c;

color: white;

border: none;

padding: 10px 20px;

border-radius: 5px;

cursor: pointer;

}

button:hover {

background-color: #d90429;

}

/\* Responsive Design \*/

@media (max-width: 768px) {

#products {

flex-direction: column;

align-items: center;

}

.product {

width: 80%;

}

}

</style>

</head>

<body>

<header id="header">

<h1>ShopEase</h1>

<p>Your One-Stop Online Shopping Destination</p>

<nav>

<span class="nav-item"><a href="#">Home</a></span>

<span class="nav-item"><a href="#">Products</a></span>

<span class="nav-item"><a href="#">Deals</a></span>

<span class="nav-item"><a href="#">Contact</a></span>

</nav>

</header>

<main>

<section id="intro" style="padding: 20px; text-align:center;">

<h2>Welcome to ShopEase</h2>

<p>Find amazing deals on electronics, fashion, and more! Enjoy free delivery and fast checkout.</p>

</section>

<section id="products">

<div class="product">

<img src="https://via.placeholder.com/250x180.png?text=Smartphone" alt="Smartphone">

<h3>Smartphone</h3>

<p class="price">$499</p>

<button>Add to Cart</button>

</div>

<div class="product">

<img src="https://via.placeholder.com/250x180.png?text=Headphones" alt="Headphones">

<h3>Wireless Headphones</h3>

<p class="price">$129</p>

<button>Add to Cart</button>

</div>

<div class="product">

<img src="https://via.placeholder.com/250x180.png?text=Laptop" alt="Laptop">

<h3>Laptop</h3>

<p class="price">$899</p>

<button>Add to Cart</button>

</div>

</section>

<section id="contact">

<h2>Subscribe to Our Newsletter</h2>

<p>Stay updated with our latest offers and discounts!</p>

<form>

<input type="text" placeholder="Enter your name" required> <br><br>

<input type="email" placeholder="Enter your email" required> <br><br>

<button type="submit">Subscribe</button>

</form>

</section>

</main>

<footer>

<p>&copy; 2025 ShopEase | All Rights Reserved</p>

</footer>

</body>

</html>

Exp4

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8" />

<meta name="viewport" content="width=device-width, initial-scale=1" />

<title>Tribute to Sharath Kamal</title>

<link

href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css"

rel="stylesheet"

/>

</head>

<body>

<header class="bg-primary text-white text-center py-5">

<div class="container">

<h1 class="display-4">Tribute to Achanta Sharath Kamal</h1>

Department of Information Technology | APSIT

<p class="lead">Indian Table Tennis Legend & Multiple CWG Gold Medalist</p>

</div>

</header>

<section class="container my-5">

<div class="row align-items-center">

<div class="col-md-4 text-center">

<img src="sharath.jpeg" alt="Sharath Kamal"

class="img-fluid rounded-circle shadow" style="max-height: 300px;" />

</div>

<div class="col-md-8">

<div class="card shadow">

<div class="card-body">

<h3 class="card-title">About Sharath Kamal</h3>

<p class="card-text">

Achanta Sharath Kamal is one of India's most celebrated table tennis players.

He has won numerous medals in the Commonwealth Games and represented

India in multiple Olympics.

A true inspiration for aspiring athletes!

</p>

<ul>

<li><strong>Born:</strong> July 12, 1982</li>

<li><strong>Sport:</strong> Table Tennis</li>

<li><strong>Awards:</strong> Padma Shri, Arjuna Award, CWG Gold

Medals</li>

</ul>

</div>

</div>

</div>

</div>

Department of Information Technology | APSIT

</section>

<section class="bg-light py-5">

<div class="container">

<h2 class="text-center mb-4">Key Achievements</h2>

<div class="row g-4 justify-content-center">

<div class="col-md-4">

<div class="card border-success h-100 shadow-sm">

<div class="card-body">

<h5 class="card-title">Commonwealth Games Success</h5>

<p class="card-text">Multiple Gold Medals across several editions.</p>

</div>

</div>

</div>

<div class="col-md-4">

<div class="card border-primary h-100 shadow-sm">

<div class="card-body">

<h5 class="card-title">Olympic Representation</h5>

<p class="card-text">Represented India in 4 consecutive Olympics (2004

2016).</p>

</div>

</div>

</div>

<div class="col-md-4">

<div class="card border-warning h-100 shadow-sm">

<div class="card-body">

<h5 class="card-title">National Champion</h5>

<p class="card-text">Multiple-time National Table Tennis Champion of

India.</p>

</div>

Department of Information Technology | APSIT

</div>

</div>

</div>

<h3 class="mt-5 mb-3 text-center">Career Highlights & Stats</h3>

<div class="table-responsive">

<table class="table table-striped table-bordered">

<thead class="table-dark">

<tr>

<th>Year</th>

<th>Event</th>

<th>Medal/Result</th>

<th>Location</th>

</tr>

</thead>

<tbody>

<tr>

<td>2006</td>

<td>Commonwealth Games</td>

<td>Gold (Team Event)</td>

<td>Melbourne, Australia</td>

</tr>

<tr>

<td>2010</td>

<td>Commonwealth Games</td>

<td>Gold (Men's Singles)</td>

<td>Delhi, India</td>

</tr>

<tr>

Department of Information Technology | APSIT

<td>2014</td>

<td>Commonwealth Games</td>

<td>Gold (Men's Singles)</td>

<td>Glasgow, Scotland</td>

</tr>

<tr>

<td>2018</td>

<td>Commonwealth Games</td>

<td>Silver (Men's Singles)</td>

<td>Gold Coast, Australia</td>

</tr>

<tr>

<td>2004-2016</td>

<td>Olympics</td>

<td>Participant</td>

<td>Various Locations</td>

</tr>

</tbody>

</table>

</div>

</div>

</section>

<script

src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/

bootstrap.bundle.min.js"

></script>

<script>

document.getElementById('backToTopBtn').addEventListener('click', () => {

window.scrollTo({ top: 0, behavior: 'smooth' });

});

</script>

<footer class="bg-dark text-white py-4 mt-5">

<div class="container text-center">

<p class="mb-2">2025 Tribute Page. All Rights Reserved.</p>

<p class="mb-2">Created with pride to honor Achanta Sharath Kamal's

legacy.</p>

</div>

</footer>

</body>

</html>

Exp5

// Program 1: Leap Year Checker (Regular Function)

function isLeapYear(year) {

if ((year % 4 === 0 && year % 100 !== 0) || year % 400 === 0) {

console.log(`${year} is a Leap Year`);

} else {

console.log(`${year} is NOT a Leap Year`);

}

}

// Example usage:

isLeapYear(2024);

isLeapYear(2025);

// Program 2: Sum using Rest Parameters (ES6 Feature)

function sumAll(...numbers) {

let total = 0;

for (let num of numbers) {

total += num;

}

console.log(`Sum = ${total}`);

}

// Example usage:

sumAll(10, 20, 30);

sumAll(5, 15, 25, 35, 45);

// Program 3: Prime Number Checker (Regular Function)

function isPrime(num) {

if (num <= 1) {

console.log(`${num} is NOT a Prime Number`);

return;

}

for (let i = 2; i <= Math.sqrt(num); i++) {

if (num % i === 0) {

console.log(`${num} is NOT a Prime Number`);

return;

}

}

console.log(`${num} is a Prime Number`);

}

// Example usage:

isPrime(7);

isPrime(12);

isPrime(19);

exp6

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Registration Form with Validation</title>

<style>

body {

font-family: Arial, sans-serif;

background-color: #f4f7fa;

margin: 0;

padding: 0;

}

.container {

width: 400px;

background-color: #fff;

padding: 25px;

margin: 50px auto;

border-radius: 10px;

box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);

}

h2 {

text-align: center;

color: #333;

}

label {

font-weight: bold;

display: block;

margin-top: 10px;

}

input, select {

width: 100%;

padding: 8px;

margin-top: 5px;

margin-bottom: 10px;

border: 1px solid #ccc;

border-radius: 5px;

}

input[type="submit"] {

background-color: #007bff;

color: white;

cursor: pointer;

border: none;

}

input[type="submit"]:hover {

background-color: #0056b3;

}

.error {

color: red;

font-size: 14px;

}

</style>

</head>

<body>

<div class="container">

<h2>Registration Form</h2>

<form id="regForm" onsubmit="return validateForm()">

<label for="name">Full Name:</label>

<input type="text" id="name" placeholder="Enter your full name">

<span id="nameError" class="error"></span>

<label for="email">Email:</label>

<input type="text" id="email" placeholder="Enter your email">

<span id="emailError" class="error"></span>

<label for="password">Password:</label>

<input type="password" id="password" placeholder="Enter password">

<span id="passwordError" class="error"></span>

<label for="confirmPassword">Confirm Password:</label>

<input type="password" id="confirmPassword" placeholder="Confirm password">

<span id="confirmError" class="error"></span>

<label for="gender">Gender:</label>

<select id="gender">

<option value="">--Select--</option>

<option value="Male">Male</option>

<option value="Female">Female</option>

<option value="Other">Other</option>

</select>

<span id="genderError" class="error"></span>

<label for="phone">Phone Number:</label>

<input type="text" id="phone" placeholder="Enter 10-digit number">

<span id="phoneError" class="error"></span>

<input type="submit" value="Register">

</form>

</div>

<script>

function validateForm() {

// Get form fields

const name = document.getElementById("name").value.trim();

const email = document.getElementById("email").value.trim();

const password = document.getElementById("password").value;

const confirmPassword = document.getElementById("confirmPassword").value;

const gender = document.getElementById("gender").value;

const phone = document.getElementById("phone").value.trim();

// Error spans

const nameError = document.getElementById("nameError");

const emailError = document.getElementById("emailError");

const passwordError = document.getElementById("passwordError");

const confirmError = document.getElementById("confirmError");

const genderError = document.getElementById("genderError");

const phoneError = document.getElementById("phoneError");

// Clear old errors

nameError.textContent = "";

emailError.textContent = "";

passwordError.textContent = "";

confirmError.textContent = "";

genderError.textContent = "";

phoneError.textContent = "";

// Validation flags

let valid = true;

// Name validation

if (name === "") {

nameError.textContent = "Please enter your name";

valid = false;

}

// Email validation (simple regex)

const emailPattern = /^[^ ]+@[^ ]+\.[a-z]{2,3}$/;

if (!email.match(emailPattern)) {

emailError.textContent = "Enter a valid email address";

valid = false;

}

// Password validation

if (password.length < 6) {

passwordError.textContent = "Password must be at least 6 characters";

valid = false;

}

// Confirm password validation

if (confirmPassword !== password) {

confirmError.textContent = "Passwords do not match";

valid = false;

}

// Gender validation

if (gender === "") {

genderError.textContent = "Please select your gender";

valid = false;

}

// Phone number validation

const phonePattern = /^[0-9]{10}$/;

if (!phone.match(phonePattern)) {

phoneError.textContent = "Enter a valid 10-digit phone number";

valid = false;

}

if (valid) {

alert("Registration Successful!");

}

return valid; // Stop form if invalid

}

</script>

</body>

</html>

Exp7

// Program 1: Using ES6 Promise to check Positive or Negative

function checkNumber(num) {

return new Promise((resolve, reject) => {

if (typeof num !== "number") {

reject("Invalid input! Please provide a number.");

} else if (num >= 0) {

resolve(`${num} is a Positive Number`);

} else {

resolve(`${num} is a Negative Number`);

}

});

}

// Example usage:

checkNumber(25)

.then(result => console.log(result))

.catch(error => console.log(error));

checkNumber(-10)

.then(result => console.log(result))

.catch(error => console.log(error));

checkNumber("abc")

.then(result => console.log(result))

.catch(error => console.log(error));

// Program 2: ES6 Iterator for Fibonacci Series

function fibonacciIterator(limit) {

let a = 0, b = 1, count = 0;

return {

next: function () {

if (count < limit) {

let value = a;

[a, b] = [b, a + b];

count++;

return { value: value, done: false };

} else {

return { done: true };

}

}

};

}

// Example usage:

const fib = fibonacciIterator(10); // Generate first 10 Fibonacci numbers

let result = fib.next();

while (!result.done) {

console.log(result.value);

result = fib.next();

}

Exp8 class:

// ClassComponentTodo.js

import React, { Component } from "react";

class ClassComponentTodo extends Component {

constructor() {

super();

this.state = {

todos: [],

input: "",

};

}

handleChange = (e) => {

this.setState({ input: e.target.value });

};

addTodo = () => {

if (this.state.input !== "") {

this.setState({

todos: [...this.state.todos, this.state.input],

input: "",

});

}

};

deleteTodo = (index) => {

const todos = this.state.todos.filter((\_, i) => i !== index);

this.setState({ todos });

};

render() {

return (

<div>

<h2>Class Component To-Do List</h2>

<input

type="text"

value={this.state.input}

onChange={this.handleChange}

placeholder="Enter task"

/>

<button onClick={this.addTodo}>Add</button>

<ul>

{this.state.todos.map((todo, index) => (

<li key={index}>

{todo} <button onClick={() => this.deleteTodo(index)}>Delete</button>

</li>

))}

</ul>

</div>

);

}

}

export default ClassComponentTodo;

function:

// FunctionalComponentTodo.js

import React, { useState } from "react";

function FunctionalComponentTodo() {

const [todos, setTodos] = useState([]);

const [input, setInput] = useState("");

const addTodo = () => {

if (input !== "") {

setTodos([...todos, input]);

setInput("");

}

};

const deleteTodo = (index) => {

setTodos(todos.filter((\_, i) => i !== index));

};

return (

<div>

<h2>Functional Component To-Do List</h2>

<input

type="text"

value={input}

onChange={(e) => setInput(e.target.value)}

placeholder="Enter task"

/>

<button onClick={addTodo}>Add</button>

<ul>

{todos.map((todo, index) => (

<li key={index}>

{todo} <button onClick={() => deleteTodo(index)}>Delete</button>

</li>

))}

</ul>

</div>

);

}

export default FunctionalComponentTodo;

App.js

// App.js

import React from "react";

import ClassComponentTodo from "./ClassComponentTodo";

import FunctionalComponentTodo from "./FunctionalComponentTodo";

function App() {

return (

<div>

<ClassComponentTodo />

<hr />

<FunctionalComponentTodo />

</div>

);

}

export default App;

exp9 App.js

import React, { useState } from "react";

import TodoList from "./TodoList";

import AddTodo from "./AddTodo";

function App() {

// State to hold list of todos

const [todos, setTodos] = useState([]);

// Function to add a new todo

const addTodo = (todo) => {

setTodos([...todos, { id: Date.now(), text: todo, completed: false }]);

};

// Function to toggle completion

const toggleTodo = (id) => {

setTodos(

todos.map((todo) =>

todo.id === id ? { ...todo, completed: !todo.completed } : todo

)

);

};

return (

<div style={{ padding: "20px", maxWidth: "400px", margin: "0 auto" }}>

<h2>React To-Do List</h2>

<AddTodo addTodo={addTodo} />

<TodoList todos={todos} toggleTodo={toggleTodo} />

</div>

);

}

export default App;

AddTodo.js (Child Component using props)

import React, { useState } from "react";

function AddTodo({ addTodo }) {

const [input, setInput] = useState("");

const handleSubmit = (e) => {

e.preventDefault();

if (input.trim()) {

addTodo(input); // pass data to parent via props

setInput(""); // reset input field

}

};

return (

<form onSubmit={handleSubmit} style={{ marginBottom: "20px" }}>

<input

type="text"

value={input}

onChange={(e) => setInput(e.target.value)}

placeholder="Add a todo"

style={{ padding: "8px", width: "70%" }}

/>

<button type="submit" style={{ padding: "8px" }}>Add</button>

</form>

);

}

export default AddTodo;

App.js

import React from "react";

import PropsTodoApp from "./PropsTodoApp"; // Import the parent component for props example

// import StateTodo from "./StateTodo"; // Uncomment this line to test the state example

function App() {

return (

<div>

{/\* Use one of the examples at a time \*/}

<PropsTodoApp />

{/\* <StateTodo /> \*/}

</div>

);

}

export default App;

exp10

import React, { useRef, useState } from "react";

function App() {

const nameRef = useRef(null); // Ref for input field

const [showName, setShowName] = useState(true); // State to control visibility

const handleClick = () => {

if (nameRef.current) {

alert(`Hello, ${nameRef.current.value}`); // Access value via ref

}

};

const handleHide = () => {

setShowName(false); // Hide the text

};

return (

<div style={{ padding: "20px", maxWidth: "400px", margin: "0 auto" }}>

<h2>React Refs Example</h2>

<input

type="text"

placeholder="Enter your name"

ref={nameRef} // Assign ref

style={{ padding: "8px", width: "70%", marginBottom: "10px" }}

/>

<div style={{ marginTop: "10px" }}>

<button onClick={handleClick} style={{ padding: "8px", marginRight: "10px" }}>

Click Me

</button>

<button onClick={handleHide} style={{ padding: "8px" }}>

Hide

</button>

</div>

{showName && nameRef.current?.value && (

<p style={{ marginTop: "20px", fontWeight: "bold" }}>

Name: {nameRef.current.value}

</p>

)}

</div>

);

}

export default App;

exp11

var express = require('express');

var app = express();

app.get('/', function (req, res) {

console.log("Got a GET request for the homepage");

res.send('Welcome to Express JS!!!');

});

app.post('/', function (req, res) {

console.log("Got a POST request for the homepage");

res.send('I am Impossible! ');

});

app.delete('/del\_student', function (req, res) {

console.log("Got a DELETE request for /del\_student");

res.send('I am Deleted!');

Department of Information Technology | APSIT

});

app.get('/enrolled\_student', function (req, res) {

console.log("Got a GET request for /enrolled\_student");

res.send('I am an enrolled student.');

});

app.get('/ab\*cd', function (req, res) {

console.log("Got a GET request for /ab\*cd");

res.send('Pattern Matched.');

});

var server = app.listen(5000, function () {

var host = server.address().address;

var port = server.address().port;

console.log("Example app listening at http://localhost:%s", port);

});