**Exercise 1: 9. ReactJS-HOL**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>ES6 Hands-on Lab</title>

<script src="https://cdn.tailwindcss.com"></script>

<style>

body {

font-family: 'Inter', sans-serif;

}

</style>

</head>

<body class="bg-gray-100 min-h-screen flex items-center justify-center p-6">

<div class="bg-white p-8 rounded-xl shadow-2xl w-full max-w-4xl space-y-8">

<h1 class="text-4xl font-bold text-center text-gray-800">ES6 Hands-on Lab</h1>

<p class="text-center text-gray-600">

This page demonstrates the solutions for the ES6 lab exercises using standard JavaScript.

</p>

<div id="app" class="space-y-10">

<div id="list-players-container" class="bg-indigo-50 p-6 rounded-xl border-l-4 border-indigo-500">

<h2 class="text-2xl font-semibold text-indigo-700 mb-4">1. List of All Players</h2>

<div id="players-output" class="space-y-2"></div>

</div>

<div id="filtered-players-container" class="bg-pink-50 p-6 rounded-xl border-l-4 border-pink-500">

<h2 class="text-2xl font-semibold text-pink-700 mb-4">2. List of Players with Scores Below 70</h2>

<div id="filtered-output" class="space-y-2"></div>

</div>

<div id="indian-players-container" class="bg-green-50 p-6 rounded-xl border-l-4 border-green-500">

<h2 class="text-2xl font-semibold text-green-700 mb-4">3. Indian Players (Odd and Even Teams)</h2>

<div class="grid md:grid-cols-2 gap-6">

<div>

<h3 class="text-lg font-medium text-green-800 mb-2">Odd Team</h3>

<div id="odd-team-output" class="space-y-1"></div>

</div>

<div>

<h3 class="text-lg font-medium text-green-800 mb-2">Even Team</h3>

<div id="even-team-output" class="space-y-1"></div>

</div>

</div>

</div>

<div id="merged-players-container" class="bg-blue-50 p-6 rounded-xl border-l-4 border-blue-500">

<h2 class="text-2xl font-semibold text-blue-700 mb-4">4. Merged Players List</h2>

<div id="merged-output" class="space-y-2"></div>

</div>

</div>

</div>

<script>

const players = [

{ id: 1, name: "Player A", score: 85, isIndian: false },

{ id: 2, name: "Player B", score: 62, isIndian: true },

{ id: 3, name: "Player C", score: 95, isIndian: false },

{ id: 4, name: "Player D", score: 58, isIndian: true },

{ id: 5, name: "Player E", score: 71, isIndian: false },

{ id: 6, name: "Player F", score: 45, isIndian: true },

{ id: 7, name: "Player G", score: 88, isIndian: false },

{ id: 8, name: "Player H", score: 68, isIndian: true },

{ id: 9, name: "Player I", score: 92, isIndian: false },

{ id: 10, name: "Player J", score: 30, isIndian: true },

{ id: 11, name: "Player K", score: 70, isIndian: false },

];

**//ListOfPlayers**

function ListPlayers(playerList) {

return playerList.map(item => `<div class="p-2 bg-gray-100 rounded-lg flex justify-between items-center">

<span class="font-medium">${item.name}</span>

<span class="text-sm text-gray-500">Score: ${item.score}</span>

</div>`);

}

const allPlayersOutput = document.getElementById('players-output');

allPlayersOutput.innerHTML = ListPlayers(players).join('');

const playersWithLowScores = players.filter(player => player.score < 70);

const filteredOutput = document.getElementById('filtered-output');

filteredOutput.innerHTML = ListPlayers(playersWithLowScores).join('');

**//IndianPlayers**

const indianPlayers = players.filter(p => p.isIndian);

const [oddPlayer1, , oddPlayer2, , oddPlayer3] = indianPlayers;

const [, evenPlayer1, , evenPlayer2] = indianPlayers;

const oddTeamOutput = document.getElementById('odd-team-output');

const evenTeamOutput = document.getElementById('even-team-output');

oddTeamOutput.innerHTML = `

<p class="bg-green-100 p-2 rounded-lg">${oddPlayer1?.name || 'N/A'}</p>

<p class="bg-green-100 p-2 rounded-lg">${oddPlayer2?.name || 'N/A'}</p>

<p class="bg-green-100 p-2 rounded-lg">${oddPlayer3?.name || 'N/A'}</p>

`;

evenTeamOutput.innerHTML = `

<p class="bg-green-100 p-2 rounded-lg">${evenPlayer1?.name || 'N/A'}</p>

<p class="bg-green-100 p-2 rounded-lg">${evenPlayer2?.name || 'N/A'}</p>

`;

const T20Players = ["Virat Kohli", "Rohit Sharma", "Jasprit Bumrah"];

const RanjiTrophyPlayers = ["Priyank Panchal", "Sheldon Jackson", "Sarfaraz Khan"];

const IndianPlayersMerged = [...T20Players, ...RanjiTrophyPlayers];

const mergedOutput = document.getElementById('merged-output');

mergedOutput.innerHTML = IndianPlayersMerged.map(

player => `<div class="p-2 bg-blue-100 rounded-lg font-medium">${player}</div>`

).join('');

</script>

</body>

</html>

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

**Exercise 2: 10. ReactJS-HOL**

**//App.jsx**

function App() {

const officeSpaces = [

{

id: 1,

name: "Modern Downtown Suite",

rent: 55000,

address: "123 Main Street, Cityville, USA",

image: "https://placehold.co/800x600/bbf7d0/16a34a?text=Office+1"

},

{

id: 2,

name: "Sunny Loft Space",

rent: 75000,

address: "456 Market Avenue, Townburg, USA",

image: "https://placehold.co/800x600/a5f3fc/0891b2?text=Office+2"

},

{

id: 3,

name: "Creative Hub",

rent: 45000,

address: "789 Tech Park, Metropolis, USA",

image: "https://placehold.co/800x600/fed7aa/ea580c?text=Office+3"

},

{

id: 4,

name: "Executive Office",

rent: 90000,

address: "101 Business Blvd, Capital City, USA",

image: "https://placehold.co/800x600/c7d2fe/3730a3?text=Office+4"

},

];

return (

<div className="bg-gray-50 min-h-screen p-8">

<div className="max-w-6xl mx-auto">

<h1 className="text-5xl font-bold text-center text-gray-800 mb-12">

Office Space Rentals

</h1>

<div className="grid grid-cols-1 sm:grid-cols-2 lg:grid-cols-3 gap-8">

{officeSpaces.map(office => (

<div key={office.id} className="bg-white rounded-2xl shadow-xl overflow-hidden transition-transform duration-300 hover:scale-105">

<img

src={office.image}

alt={`Image of ${office.name}`}

className="w-full h-48 object-cover"

onError={(e) => { e.target.src = 'https://placehold.co/800x600/e5e7eb/6b7280?text=Image+Not+Found'; }}

/>

<div className="p-6">

<h2 className="text-2xl font-semibold text-gray-800 mb-2">

{office.name}

</h2>

<p className="text-xl font-bold">

Rent:

<span className={office.rent > 60000 ? "text-green-600" : "text-red-600"}>

${office.rent.toLocaleString()}

</span>

</p>

<p className="text-gray-500 mt-2">

Address: {office.address}

</p>

</div>

</div>

))}

</div>

</div>

</div>

);

}

export default App;

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

**Exercise 3: 11. ReactJS-HOL**

**//App.jsx**

import { useState } from 'react';

function App() {

const [count, setCount] = useState(0);

const incrementValue = () => {

setCount(prevCount => prevCount + 1);

};

const decrementValue = () => {

setCount(prevCount => prevCount - 1);

};

const sayWelcome = () => {

alert("Welcome! This is a static message.");

};

const handlePress = () => {

alert("I was clicked!");

};

const [amount, setAmount] = useState('');

const [currency, setCurrency] = useState('');

const [convertedAmount, setConvertedAmount] = useState('');

const handleConversion = () => {

const rate = 0.011; // A simple hardcoded conversion rate for demonstration

if (amount && currency.toLowerCase() === 'rupies') {

const euroAmount = parseFloat(amount) \* rate;

setConvertedAmount(`${euroAmount.toFixed(2)} Euros`);

alert(`The converted amount is: ${euroAmount.toFixed(2)} Euros`);

} else {

setConvertedAmount('Invalid input');

alert('Please enter a valid amount and "Rupies" as the currency.');

}

};

return (

<div className="bg-slate-900 min-h-screen p-8 text-white font-sans">

<div className="max-w-4xl mx-auto space-y-12">

<h1 className="text-5xl font-bold text-center text-indigo-400">

Event Handling Application

</h1>

<div className="bg-slate-800 p-8 rounded-2xl shadow-xl flex flex-col items-center gap-6">

<h2 className="text-3xl font-semibold text-white">Counter</h2>

<p className="text-7xl font-mono font-extrabold text-indigo-300">

{count}

</p>

<div className="flex gap-4">

<button

onClick={incrementValue}

className="bg-indigo-600 text-white font-semibold py-3 px-6 rounded-xl shadow-lg hover:bg-indigo-500 transition-colors active:scale-95"

>

Increment

</button>

<button

onClick={decrementValue}

className="bg-red-600 text-white font-semibold py-3 px-6 rounded-xl shadow-lg hover:bg-red-500 transition-colors active:scale-95"

>

Decrement

</button>

</div>

</div>

<div className="bg-slate-800 p-8 rounded-2xl shadow-xl flex flex-col items-center gap-6">

<h2 className="text-3xl font-semibold text-white">Static Message</h2>

<button

onClick={sayWelcome}

className="bg-green-600 text-white font-semibold py-3 px-6 rounded-xl shadow-lg hover:bg-green-500 transition-colors active:scale-95"

>

Say Welcome

</button>

</div>

<div className="bg-slate-800 p-8 rounded-2xl shadow-xl flex flex-col items-center gap-6">

<h2 className="text-3xl font-semibold text-white">Dynamic Message</h2>

<button

onClick={handlePress}

className="bg-purple-600 text-white font-semibold py-3 px-6 rounded-xl shadow-lg hover:bg-purple-500 transition-colors active:scale-95"

>

Press Me

</button>

</div>

<div className="bg-slate-800 p-8 rounded-2xl shadow-xl flex flex-col items-center gap-6">

<h2 className="text-3xl font-semibold text-white">Currency Converter</h2>

<p className="text-xl text-white">Amount to convert: <span className="text-indigo-400 font-bold">{convertedAmount}</span></p>

<div className="flex flex-col gap-4 w-full max-w-sm">

<input

type="text"

placeholder="Amount"

value={amount}

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

className="p-3 rounded-lg bg-slate-700 text-white border border-slate-600 focus:ring-2 focus:ring-indigo-500"

/>

<input

type="text"

placeholder="Currency (e.g., Rupies)"

value={currency}

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

className="p-3 rounded-lg bg-slate-700 text-white border border-slate-600 focus:ring-2 focus:ring-indigo-500"

/>

<button

onClick={handleConversion}

className="bg-yellow-500 text-slate-900 font-semibold py-3 px-6 rounded-xl shadow-lg hover:bg-yellow-400 transition-colors active:scale-95"

>

Convert to Euros

</button>

</div>

</div>

</div>

</div>

);

}

export default App;

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

**Exercise 4: 12. ReactJS-HOL**

**//App.jsx**

import { useState } from 'react';

function App() {

const [isLoggedIn, setIsLoggedIn] = useState(false);

const handleLoginClick = () => {

setIsLoggedIn(!isLoggedIn);

};

return (

<div className="min-h-screen bg-gray-50 flex flex-col items-center justify-center p-8 font-sans">

<div className="bg-white p-12 rounded-xl shadow-2xl text-center space-y-8 max-w-lg w-full">

<h1 className="text-4xl font-bold text-gray-800">

{isLoggedIn ? "Welcome back" : "Please sign up"}

</h1>

<button

onClick={handleLoginClick}

className={`

font-semibold py-3 px-8 rounded-lg shadow-md transition-colors duration-200

${isLoggedIn

? "bg-red-600 text-white hover:bg-red-700" // Red for Logout

: "bg-green-600 text-white hover:bg-green-700" // Green for Login

}

`}

>

{isLoggedIn ? "Logout" : "Login"}

</button>

</div>

</div>

);

}

export default App;

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

**Exercise 5: 13. ReactJS-HOL**

**//App.jsx**

import { useState } from 'react';

const booklist = [

{ id: 1, name: "I am aname", price: 500 },

{ id: 2, name: "I am another one", price: 800 },

{ id: 3, name: "I am a new one", price: 650 }

];

const BloagDetails = () => (

<div className="p-6 bg-yellow-50 rounded-lg shadow-inner">

<h3 className="text-xl font-bold text-yellow-800">React Learning</h3>

<p className="text-gray-600 mt-2">Welcome to learning React!</p>

<p className="text-sm text-gray-500 mt-1">Installation</p>

<p className="text-sm text-gray-500">You can install React from npm.</p>

</div>

);

const BookDetails = () => (

<div className="p-6 bg-green-50 rounded-lg shadow-inner">

<h3 className="text-xl font-bold text-green-800 mb-4">Book Details</h3>

{booklist.map(book => (

<div key={book.id} className="mb-2 p-3 bg-green-100 rounded-lg">

<p className="font-semibold">{book.name}</p>

<p className="text-sm text-green-700">Price: ${book.price}</p>

</div>

))}

</div>

);

const CourseDetails = () => (

<div className="p-6 bg-blue-50 rounded-lg shadow-inner">

<h3 className="text-xl font-bold text-blue-800">Course Details</h3>

<p className="text-gray-600 mt-2">Angular</p>

<p className="text-sm text-gray-500 mt-1">Master React</p>

<p className="text-sm text-gray-500">6/3/2017</p>

</div>

);

function App() {

const [selectedComponent, setSelectedComponent] = useState('course');

const renderComponent = () => {

switch (selectedComponent) {

case 'course':

return <CourseDetails />;

case 'book':

return <BookDetails />;

case 'blog':

return <BlogDetails />;

default:

return <p className="text-center text-gray-500">Please select an option.</p>;

}

};

return (

<div className="min-h-screen bg-gray-50 p-8 font-sans">

<div className="max-w-4xl mx-auto space-y-10">

<div className="flex justify-center gap-4">

<button

onClick={() => setSelectedComponent('course')}

className={`py-2 px-6 rounded-xl font-semibold transition-colors duration-200 ${

selectedComponent === 'course' ? 'bg-indigo-600 text-white shadow-lg' : 'bg-gray-200 text-gray-700 hover:bg-gray-300'

}`}

>

Course Details

</button>

<button

onClick={() => setSelectedComponent('book')}

className={`py-2 px-6 rounded-xl font-semibold transition-colors duration-200 ${

selectedComponent === 'book' ? 'bg-indigo-600 text-white shadow-lg' : 'bg-gray-200 text-gray-700 hover:bg-gray-300'

}`}

>

Book Details

</button>

<button

onClick={() => setSelectedComponent('blog')}

className={`py-2 px-6 rounded-xl font-semibold transition-colors duration-200 ${

selectedComponent === 'blog' ? 'bg-indigo-600 text-white shadow-lg' : 'bg-gray-200 text-gray-700 hover:bg-gray-300'

}`}

>

Blog Details

</button>

</div>

<div className="mt-8 border border-gray-300 p-6 rounded-2xl shadow-xl bg-white">

{renderComponent()}

</div>

</div>

</div>

);

}

export default App;

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.