Sukkur IBA University

Final Examination
Practical Exam

Course: Enterprise Application Development | **Class:** BS-VI
Date: 3rd June 2025 | **Time:** 1 Hour
Instructor: Khalid Hussain | **Marks:** 15 Marks

Instructions:

- No Mobile Phones are allowed.
- Save your files after any changes.
- Submit **a single React component file (not an entire project) and provide the GitHub repository link.** The link must be pasted directly in the LMS text submission box, not inside a text file.
- Once done, submit it in LMS (Section: Part-B).

Part-B (Practical)

(No Internet, Copilot/ChatGPT, or any AI model/extension is allowed)

Build a **single React component** that replicates the exact behavior shown in the **provided GIF demo (attached)**.

Deliverables:

- A **single React component file** (e.g., **TrafficLightSimulator.jsx**)
- A **GitHub repository link**

Refer to the GIF Demo

Use the provided GIF to observe:

- The sequence of light changes (Red → Yellow → Green → Red…)
- Each light stays active for 3 seconds
- Only one light is active at a time
- Smooth transitions and style

Requirements

1. React Component Setup

- Create a file like **TrafficLightSimulator.jsx**
- Component must be self-contained (all JSX + CSS in one file)
- Match the design shown in the GIF

2. Light Logic

- Use **useState** to track the current light (red/yellow/green)
- Use **useEffect** with setInterval to change light every 3 seconds
- Apply active CSS class to the current light

3. Styling

- Match the GIF's light casing and glow effects
- Red, yellow, and green circles should visibly change
- Add smooth CSS transitions

Git Commit Plan (Minimum 4 Commits)

Commit #	Commit Message	Task
1	`Initial commit: add static HTML and CSS for traffic light`	Setup project and layout (3 lights, no logic)
2	`Refactor: convert to functional React component`	JSX structure for lights, apply classes
3	`Feature: implement light cycle logic with useEffect`	Cycle lights every 3s using state + effect
4	`Style: enhance UI with transitions and active styles`	