**Week 8 Plan**

**Day 1: React Components**

**Theory**

* Understanding React Components
* Functional vs. Class Components
* Props & State
* Component Composition
* Lifecycle Methods (Brief)
* Hands-on Example: Creating a Simple Bank Component

**Assignments (Classroom Practical Session)**

**1. Fixed Deposit Summary Component**

* Create a FixedDepositSummary component that takes props:
  + depositAmount, interestRate, tenure, maturityAmount
* Display these details in a styled card.
* Add a button to toggle between "Summary View" and "Detailed View."

**2. Recurring Deposit Contribution Tracker**

* Create an RDContributionTracker component showing:
  + Monthly Contribution, Tenure, Total Invested, Estimated Maturity
* Allow users to dynamically change the contribution.

**3. Customer Investment Profile Component**

* Create a CustomerProfile component displaying:
  + Name, Total FDs, Total RDs, Total Investment Value
* Accept props for customer details and display them in a formatted way.

**Day 2: Forms & UI**

**Theory**

* Controlled & Uncontrolled Components
* Handling Form Events
* Form Validation & Submission
* UI Components (Buttons, Inputs, Dropdowns)
* Basic CSS & Styling in React

**Assignments (Classroom Practical Session)**

**1. FD Investment Form**

* Create a form with fields:
  + Deposit Amount, Tenure, Interest Rate (auto-filled)
* Display **maturity amount** after form submission.

**2. RD Plan Creation Form**

* Create a form with fields:
  + Monthly Contribution, Tenure (Months), Bank Name
* Show a summary after submission.

**3. Bank Branch Selector UI**

* Create a dropdown to select a bank branch.
* Display branch details (address, branch code, IFSC) dynamically when selected.

**Day 3: React Routing**

**Theory**

* Introduction to React Router
* Route Setup & Navigation
* Dynamic Routing
* Using useParams and useNavigate

**Assignments (Classroom Practical Session)**

**1. FD & RD Management Portal**

* Implement routing for:
  + / → Home
  + /fixed-deposit → FD Calculator
  + /recurring-deposit → RD Summary
  + /investment-history → Display previous FDs/RDs

**2. User Dashboard Navigation**

* Implement routes for:
  + /dashboard → Overview
  + /create-fd → FD Creation
  + /create-rd → RD Opening
  + /support → Customer Support

**3. Investment Detail Page (Dynamic Routing)**

* Clicking on an FD/RD from the investment list navigates to a detailed page showing full investment details (/investment/:id).

**Day 4: Event Handling & Styling**

**Theory**

* Event Handling in React
* Handling Button Clicks & Form Inputs
* Inline & External CSS in React
* Using Styled Components

**Assignments (Classroom Practical Session)**

**1. FD Maturity Calculator with Event Handling**

* User inputs Deposit Amount and Tenure.
* Clicking "Calculate Maturity" displays maturity amount.
* Interest rate auto-adjusts based on tenure.

**2. RD Contribution Tracker with Events**

* Users adjust their **monthly contribution** dynamically using (+ / -).
* Display updated **total investment** and **maturity amount** in real-time.

**3. Theme Switcher for Investment Dashboard**

* Add a button to toggle between **light mode** and **dark mode** for styling using React state.

**Day 5: React Redux**

**Theory**

* Introduction to Redux
* Redux Store, Actions, Reducers
* Using Redux with Functional Components
* Managing Global State

**Assignments (Classroom Practical Session)**

**1. FD Account State Management using Redux**

* Store FD list in Redux with:
  + amount, tenure, interest rate, status
* Allow users to:
  + **Add** a new FD
  + **Mark** FD as matured

**2. RD Investment Tracker with Redux**

* Store **monthly RD contributions** in Redux.
* Allow users to:
  + Add a new RD
  + Increase/decrease monthly contributions
  + Calculate total maturity amount dynamically

**3. Notifications for Investment Updates (Redux State Management)**

* Maintain a Redux state to show notifications when:
  + A new FD/RD is created
  + A maturity status is updated