DAY 26 – FORMS IN REACT JS

PROJECT 1 — “Job Application Portal Form”

Objective:

Build a complete job application form for candidates to apply for a company job posting.

Features & Requirements:

1. **Personal Details Section**
   - Fields: Full Name, Email, Mobile, Gender (radio), Date of Birth (date picker)
   - Controlled components using useState
   - Validation: Required + Email format
2. **Professional Details Section**
   - Dropdown: Select Job Role (Developer, Designer, PM)
   - Textarea: Cover Letter
   - Upload Resume (file input with preview)
   - Conditional field: If “Developer” → show “Preferred Tech Stack” input
3. **Validation**
   - Custom validation on submit
   - Show red borders and error text for invalid fields
4. **UX Enhancements**
   - Disable Submit button until all mandatory fields are filled
   - Loading spinner after submission
   - Success message (conditional rendering)
5. **Integration**
   - Send POST request to /api/apply (mock API)
   - Reset form on success

Concepts Reinforced:
Controlled inputs, conditional rendering, file upload, form validation, disabled states, reset.

PROJECT 2 — “Online Course Enrollment Form”

Objective:

Create a form that lets students enroll in online courses with dynamic input fields.

Features & Requirements:

1. **Student Details**
   - Name, Email, Country (dropdown), Gender (radio)
   - Controlled form inputs
2. **Course Selection**
   - Dynamic Fields:
     → User can "Add Course" or "Remove Course"
     → Each course has dropdown: Course Name, Duration, and Fees
3. **Validation**
   - Ensure no empty course fields
   - Check if email already exists (simulate async API check with useEffect)
4. **Submission**
   - On Submit → Show "Review your data" section
   - Then confirm submission (conditional rendering)
5. **UX Enhancements**
   - Progress indicator: Step 1 → Step 2 → Step 3
   - Reset form after final confirmation

Concepts Reinforced:
Dynamic form fields, async validation, multi-step forms, conditional rendering, form review logic.

PROJECT 3 — "E-Commerce Checkout Form"

Objective:

Simulate a complete checkout page for an online shopping website.

Features & Requirements:

1. **Shipping Address**
   - Name, Phone, Street, City, State, Pincode (controlled)
   - Validate: required fields and pincode numeric
2. **Payment Details**

   - Radio: Payment Mode (Card, UPI, COD)
   - If Card → show Card Number, Expiry, CVV inputs (conditional rendering)
   - If UPI → show UPI ID input only

3. **Form Behavior**
   - Disable form after successful order placement
   - “Edit Address” button to toggle editable mode (controlled states)

4. **Validation**
   - Regex-based validation for Card and UPI
   - Show live validation errors as user types

5. **Integration**
   - Simulate POST /api/checkout and show “Order Placed” message
   - Reset after 5 seconds (via useEffect timer)

Concepts Reinforced:
Conditional rendering, regex validation, toggle state, controlled vs disabled inputs, post-submit UX.

PROJECT 4 — “Event Registration with Live Validation”

Objective:

A form where users register for an event and see validation feedback instantly.

Features & Requirements:

1. **Form Fields**
   - Full Name, Email, Phone, Event Type (dropdown), Number of Tickets (counter input)
   - Checkbox: “I agree to terms”

2. **Live Validation**
   - Real-time input validation using useEffect on every change
   - Display green borders for valid inputs and red for invalid ones
   - Disable Submit until all validations pass

3. **UX Improvements**
   - Show a live “Form Progress” indicator (e.g., “4 of 5 fields valid”)
   - Show loader animation while validating email via fake API

4. **Submission**
    - Submit button sends POST /api/register
    - Show dynamic thank-you message with user name
5. **Reset**
    - Add “Clear Form” button to reset all fields to default

Concepts Reinforced:
Real-time validation, controlled inputs, dynamic button disabling, UX feedback, progress tracking.

PROJECT 5 — “Feedback & Rating Form with API”

Objective:

Collect user feedback and ratings for a product or service.

Features & Requirements:

1. **Form Fields**
    - Name, Email (controlled)
    - Rating (1 to 5 stars — clickable icons)
    - Feedback message (textarea)
    - File upload (optional screenshot)
2. **Validation**
    - Required fields: Name, Email, Rating
    - Email validation using regex
    - Optional image validation (file size < 2MB)
3. **UX & Conditional Rendering**
    - Disable submit until rating > 0
    - Show “Thank you” message after submission
    - “Submit another feedback” button resets the form
4. **API Integration**
    - POST data to /api/feedback
    - Simulate success/error with random outcomes
    - Show toast or alert on success/error

5. **UI Enhancements**
    - Stars turn gold when selected
    - Text counter for message box
    - Smooth transitions for success message

Concepts Reinforced:
Custom UI controls, controlled components, validation, conditional rendering, API interaction, success/error handling.
