

# Assignment: Form Handling

---

## Simple Registration Form

### Scenario:

You are tasked with building a simple user registration form where users can enter their details to sign up. The form will include fields for name, email, and password.

This assignment will help you practice form handling in React with TypeScript, focusing on how to manage form state, handle form submission, and validate input data.

### Requirements:

#### ➤ Project Setup:

- Create a new React project with a TypeScript template and name it RegistrationApp.

#### ➤ Component Structure:

##### ❖ Create a RegistrationForm Component:

- This component should include input fields for the user's name, email, and password.
- The component should manage the form state using React's useState hook.
- It should handle form submission and validate that all fields are filled out.

#### ➤ Form Fields:

- Name: A text input where users can enter their full name.
- Email: An email input where users can enter their email address.
- Password: A password input where users can enter their password.

#### ➤ Form Validation:

- Ensure that all fields are required. If a user tries to submit the form with any empty fields, display an appropriate error message.

➤ Form Submission:

- When the form is submitted, display a summary of the entered details below the form.
- If any field is left empty, prevent the form from submitting and display an error message.

➤ Expected Outcome:

❖ A simple registration form where users can:

- Enter their name, email, and password.
- Submit the form to see a summary of their entered details.
- Receive feedback if they try to submit the form without filling out all fields.

## Tasks:

➤ Form Handling:

- Implement the RegistrationForm component to manage the form state using React's useState hook.
- Handle input changes and update the form state accordingly.

➤ Form Validation:

- Implement simple validation to ensure that all fields are required.
- If validation fails, display an error message and prevent form submission.

➤ Form Submission:

- On form submission, display a summary of the entered details.
- Ensure the form only submits if all fields are valid.

➤ Styling:

- Apply basic styling to make the form user-friendly and visually appealing.
-