# Lab 16: Form Validation

## Objectives

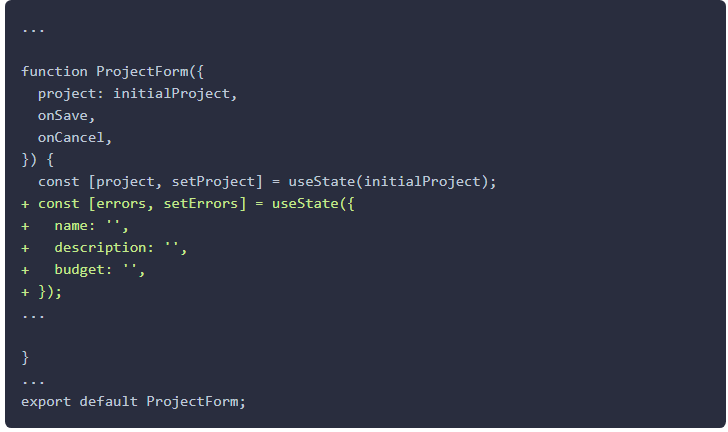
*  Add Form Validation

## Steps

### Add Form Validation

1. **Open** the file src\project\ProjectForm.js
2. **Initialize** an errors object as a state variable to {name: '', description: '', budget: ''} so we can hold form errors in the component's state.

#### src\projects\ProjectForm.js

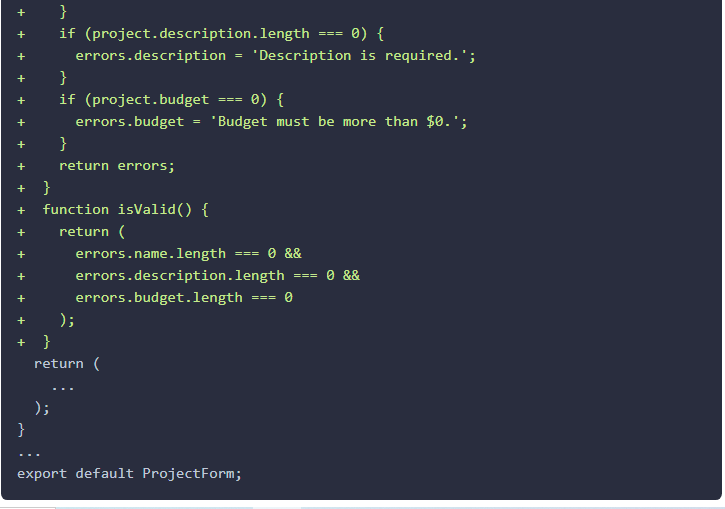


1. **Implement** a validate function in the ProjectForm component that meets these **requirements**.
2. Name is required.
3. Name needs to be at least 3 characters long.
4. Description is required.
5. Budget must be greater than \$0.

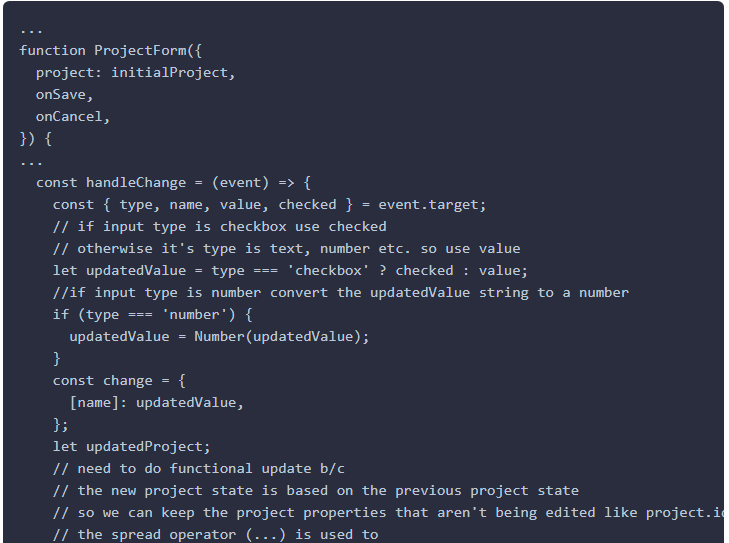
Also, **implement** an isValid function in the ProjectForm component that checks whether there are any validation errors.

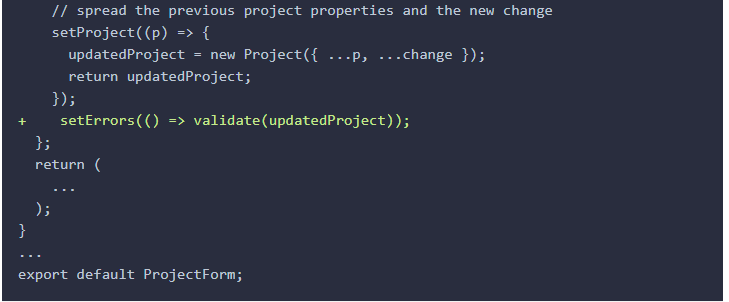
#### src\projects\ProjectForm.js





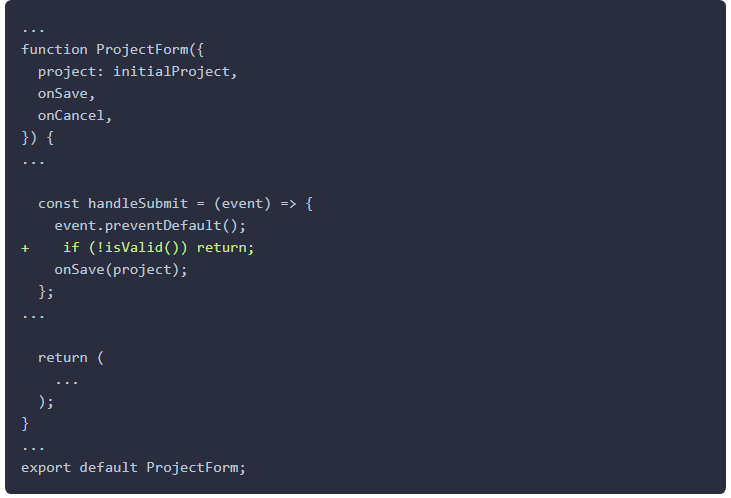
1. **Call the validate function inside handleChange to determine if there are any errors and then set them into the errors state variable.**





1. **Call the isValid function on form submit and return back out of the function before saving changes if the form is invalid.**

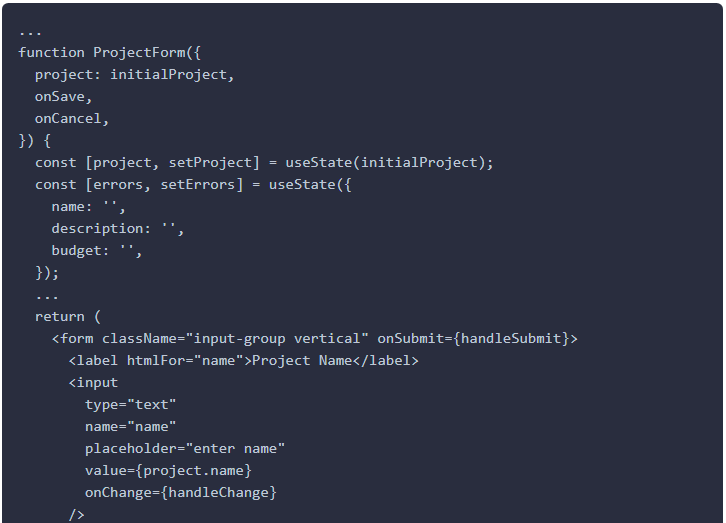
#### src\projects\ProjectForm.js

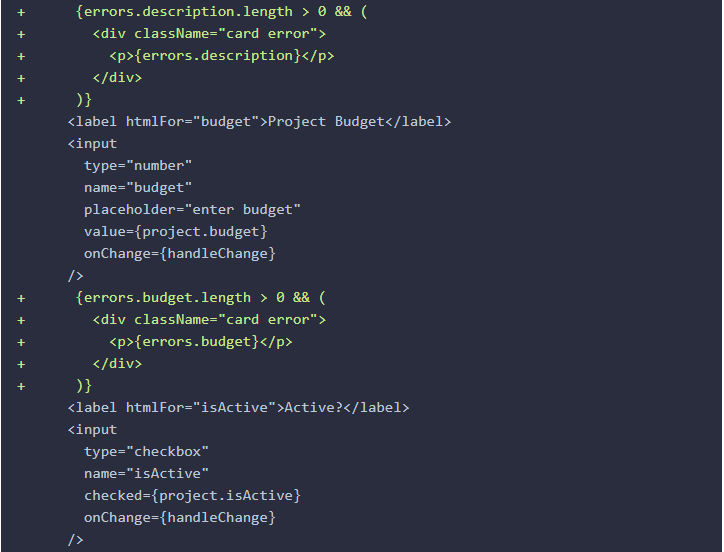


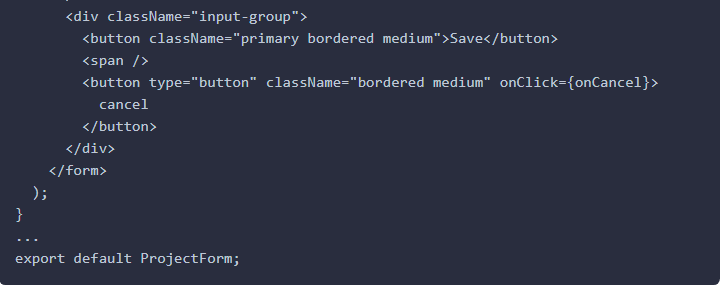
1. **In the JSX returned by the component: display the validation messages using the following HTML template.**



#### src\projects\ProjectForm.js







1. **Verify** the application is working by **following these steps**.
   1. **Click** the **edit** **button** on any project.
   2. **Delete** the contents of the **project name** textbox.
   3. **Verify** the **error message** displays.
   4. **Test** the **other** validation rules.

### ✔ You have completed Lab 16