# Lab 13: More Component Communication

## Objectives

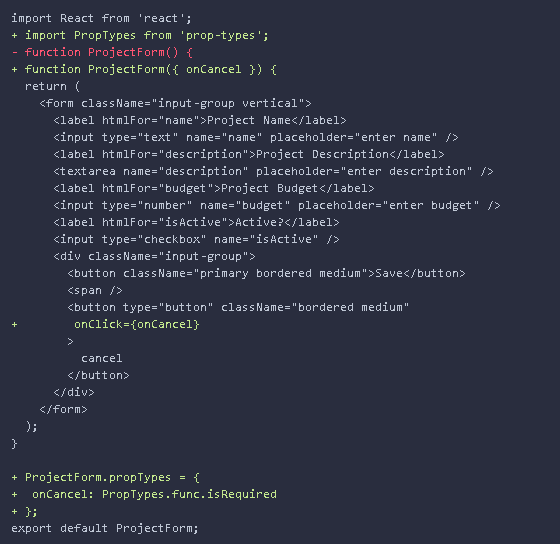
*  In a child component, accept a function as a prop and invoke it
*  In a parent component, implement a function and pass it as a prop to a child component

## Steps

### In a child component, accept a function as a prop and invoke it

1. **Open** the **file** src\projects\ProjectForm.js.
2. In the ProjectFormProps interface, **add** an onCancel **event handler** that takes no parameters and returns void.
3. Update the cancel button and add a click event to invoke the function passed into the onCancel prop.

#### src\projects\ProjectForm.js



### In a parent component, implement a function and pass it as a prop to a child component

1. In src\projects\ProjectList.js **add** a cancelEditing**event handler** toProjectListthat sets the state of the component so that editingProject is an empty object {}.
2. **Wire** up the **onCancel** **event** of the <ProjectForm /> component rendered in the ProjectList to the cancelEditing event handler.

#### src\projects\ProjectList.js



1. **Verify the application is working by following these steps:**
   1. **Open** the application in your browser and refresh the page.
   2. **Click** the **edit** button for a project.
   3. **Verify** the <ProjectCard /> is removed and replaced by the <ProjectForm/>.

The <ProjectForm/> will be empty at this point. We will fill in the data in a future lab.

* 1. **Click** the **cancel** button on the form.
  2. **Verify** the <ProjectForm/> is removed and replaced by the <ProjectCard />.

This lab is conceptually very similar to Lab 11 in that we are invoking a function on a parent component in a child component.

### ✔ You have completed Lab 13