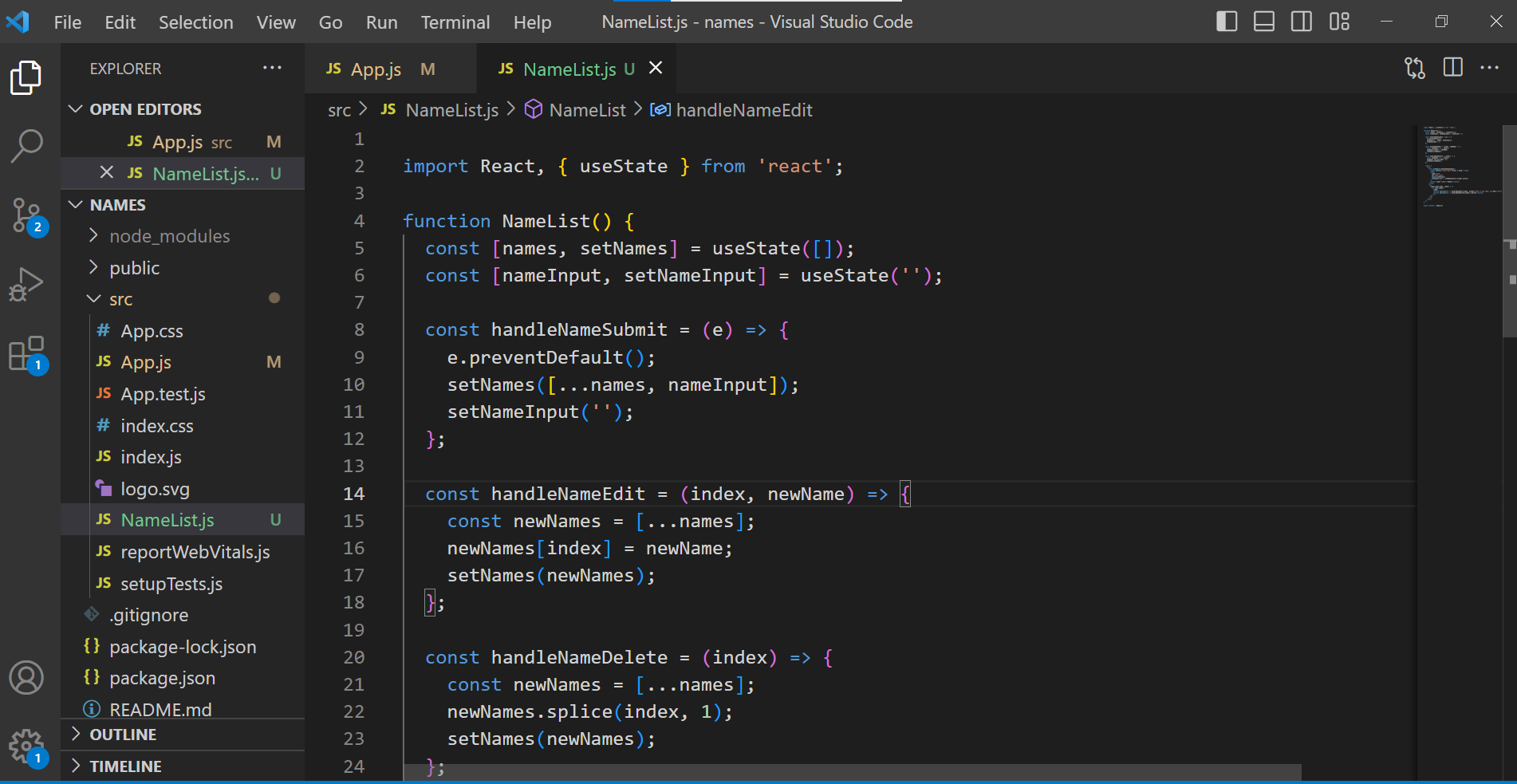
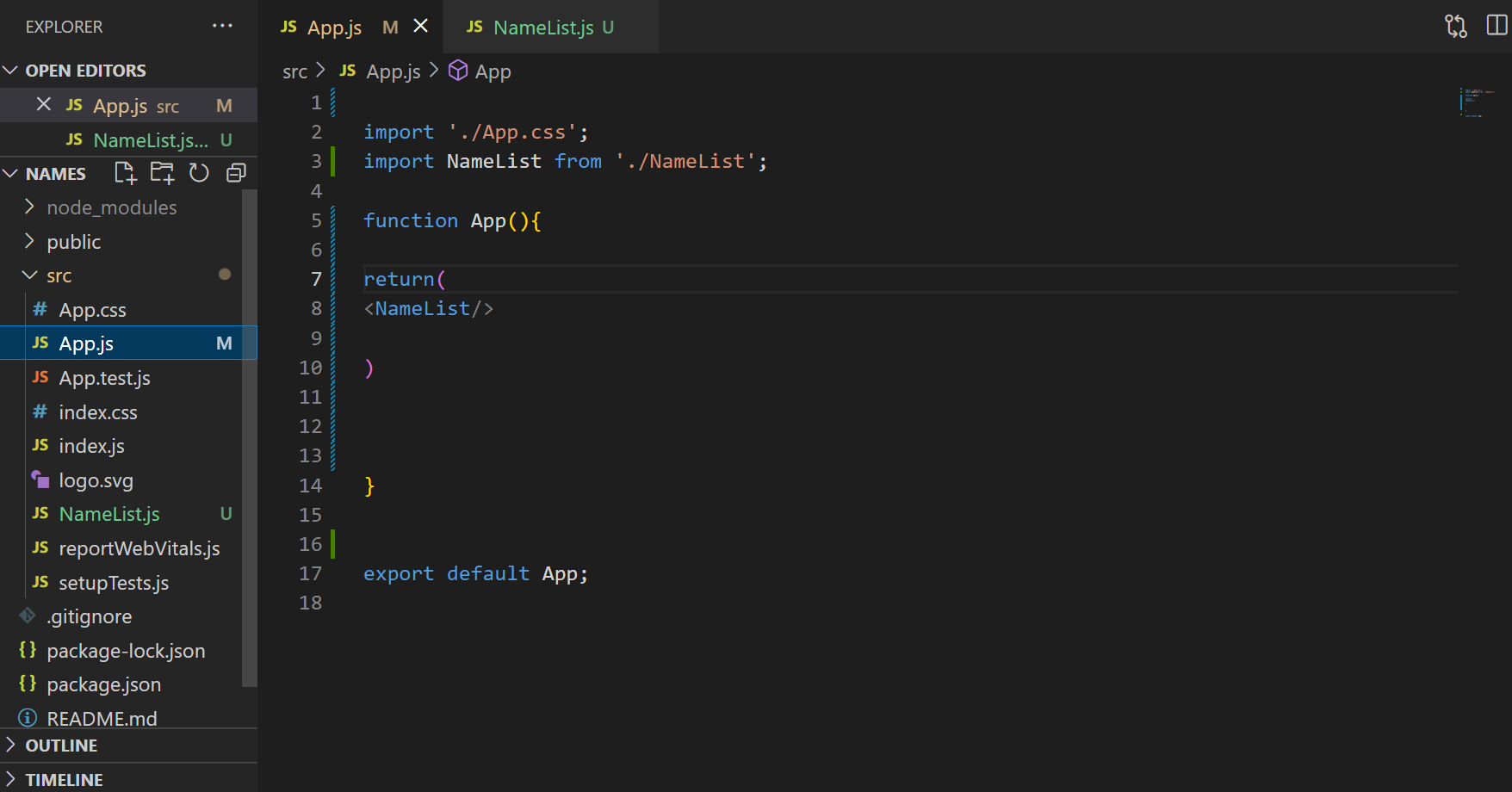
1. Create a component which has 2 parts. The first part has a form where user can enter a name and submit. When the name is submitted, it's added to the list of names in the parent components state. The list names components get the name list as a props. It maps through each name and prints the name in a list.

Additional: Provide edit and delete buttons in the list name component. When edit is clicked - the selected name is updated in the form and the user can update it. Delete will remove the element from the list.

See this demo how to code in vs

* Create a file NameList and create a child function component (code is at bottom)
* Once created go the parent function component and the child component





**(Creating child component )**

import React, { useState } from 'react';

function NameList() {

const [names, setNames] = useState([]);

const [nameInput, setNameInput] = useState('');

const handleNameSubmit = (e) => {

e.preventDefault();

setNames([...names, nameInput]);

setNameInput('');

};

const handleNameEdit = (index, newName) => {

const newNames = [...names];

newNames[index] = newName;

setNames(newNames);

};

const handleNameDelete = (index) => {

const newNames = [...names];

newNames.splice(index, 1);

setNames(newNames);

};

return (

<div>

<form onSubmit={handleNameSubmit}>

<label htmlFor="name-input">Enter a name:</label>

<input

type="text"

id="name-input"

value={nameInput}

onChange={(e) => setNameInput(e.target.value)}

/>

<button type="submit">Submit</button>

</form>

<ul>

{names.map((name, index) => (

<li key={index}>

{name}

<button onClick={() => handleNameEdit(index, prompt('Enter a new name:'))}>Edit</button>

<button onClick={() => handleNameDelete(index)}>Delete</button>

</li>

))}

</ul>

</div>

);

}

export default NameList

**(calling child component in parent component)**

import './App.css';

import NameList from './NameList';

function App(){

return(

<NameList/>

)

}

export default App;

This code is a React component called NameList that manages a list of names.

It uses the useState hook to define two state variables: names, which is initialized as an empty array, and nameInput, which is initialized as an empty string.

The component renders a form with an input field and a submit button. When the form is submitted, the handleNameSubmit function is called, which adds the current nameInput value to the names array using the setNames function and then clears the nameInput value.

The component also renders a list of names using the map function. For each name in the names array, a list item is created with the name, an "Edit" button, and a "Delete" button.

When the "Edit" button is clicked, it calls the handleNameEdit function, which prompts the user for a new name and updates the corresponding name in the names array using the setNames function.

When the "Delete" button is clicked, it calls the handleNameDelete function, which removes the corresponding name from the names array using the splice method and updates the state using the setNames function.

Overall, this component allows users to add, edit, and delete names in a list, and uses React's state management to keep the UI synchronized with the underlying data.