**Лабораторна робота № 8**

**Тема:** Основи роботи із бібліотекою React.js. State i Props

**Мета:** Навчитись працювати з state і props.

**Завдання**

1. Створити додаток на реакт згідно варіанту, в якому необхідно обов’язково використати props,

state, створити мінімум дві компоненти, що мають бути вкладеними.

Телефона книга. Можливість додавати контакти до книги, переглядати усі і тд.

**Хід роботи**

App.js

import PhoneBook from "./PhoneBook";

function App() {  
 return (  
 <div className="App">  
 <PhoneBook/>  
 </div>  
 );  
}  
  
export default App;

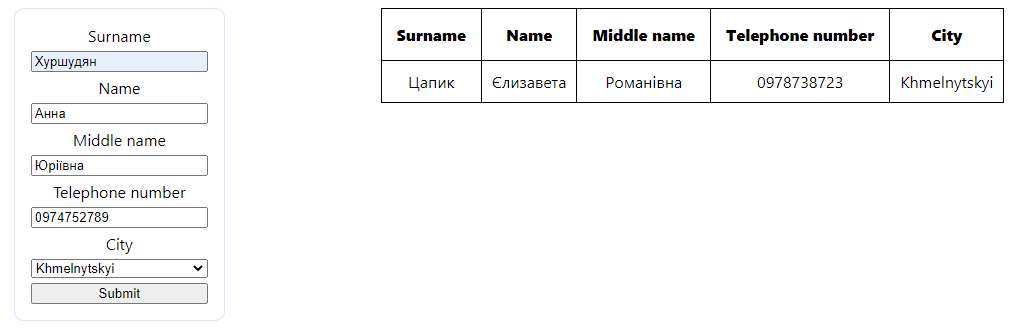
PhoneBook.js

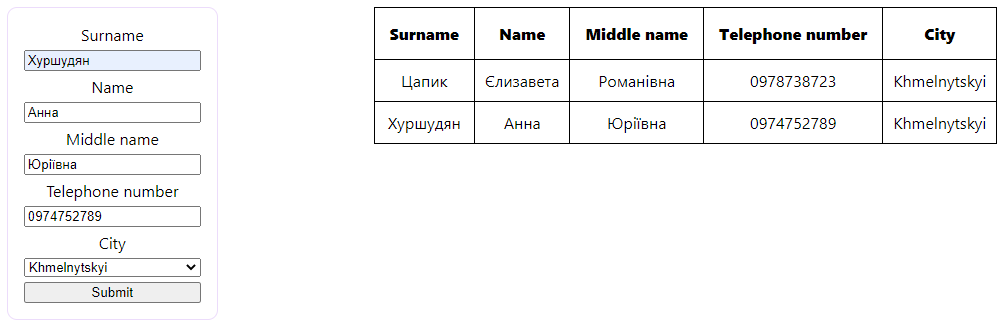
import ***React***, {useCallback, useState} from 'react';  
import './PhoneBook.css';  
  
function PhoneBook() {  
 let [contacts, setContacts] = useState([]);  
  
 const submitForm = useCallback((e) => {  
 e.preventDefault();  
 const surnameInput = ***document***.getElementById('surname');  
 const nameInput = ***document***.getElementById('name');  
 const middle\_nameInput = ***document***.getElementById('middle\_name');  
 const telephone\_numberInput = ***document***.getElementById('telephone\_number');  
 const cityInput = ***document***.getElementById('city');  
 setContacts([...contacts, {  
 "surname": surnameInput.value,  
 "name": nameInput.value,  
 "middle\_name": middle\_nameInput.value,  
 "telephone\_number": telephone\_numberInput.value,  
 "city": cityInput.value,  
 }])  
 }, [contacts, setContacts]);  
 return (  
 <div className="Student flex justify-content\_space-evenly align-items\_flex-start m\_2rem">  
 <form className="form flex flex-direction\_column" onSubmit={submitForm}>  
 <label htmlFor="surname">Surname</label>  
 <input id="surname" type="text" required/>  
 <label htmlFor="name">Name</label>  
 <input id="name" type="text" required/>  
 <label htmlFor="middle\_name">Middle name</label>  
 <input id="middle\_name" type="text" required/>  
 <label htmlFor="telephone\_number">Telephone number</label>  
 <input id="telephone\_number" type="tel" required/>  
 <label htmlFor="city">City</label>  
 <select id="city">  
 <option value="Khmelnytskyi">Khmelnytskyi</option>  
 <option value="Vinnitsa">Vinnitsa</option>  
 <option value="Shepetivka">Shepetivka</option>  
 <option value="Dunaivtsi">Dunaivtsi</option>  
 </select>  
 <button type="submit">Submit</button>  
 </form>  
 <Contacts contacts={contacts}/>  
 </div>  
 );  
}  
  
function Contacts(props) {  
 return (  
 <div className="Contacts">  
 {  
 props.contacts.length > 0 ?  
 <table>  
 <thead>  
 <tr>  
 <th>Surname</th>  
 <th>Name</th>  
 <th>Middle name</th>  
 <th>Telephone number</th>  
 <th>City</th>  
 </tr>  
 </thead>  
 <tbody>  
 {  
 props.contacts.map((s, index) =>  
 <tr key={index}>  
 <td>{s.surname}</td>  
 <td>{s.name}</td>  
 <td>{s.middle\_name}</td>  
 <td>{s.telephone\_number}</td>  
 <td>{s.city}</td>  
 </tr>  
 )  
 }  
 </tbody>  
 </table> : ''}  
 </div>  
 );  
}  
  
function Contact(props) {  
 return (  
 <div className="Contact">  
 <h1>{props.surname} {props.name} {props.middle\_name}</h1>  
 <h2>{props.telephone\_number}</h2>  
 <h2>{props.city}</h2>  
 </div>  
 );  
}  
  
export default PhoneBook;

PhoneBook.css

.flex {  
 display: flex;  
}  
  
.align-items\_flex-start{  
 align-items: flex-start;  
}  
  
.justify-content\_center {  
 justify-content: center;  
}  
  
.justify-content\_space-evenly {  
 justify-content: space-evenly;  
}  
  
.flex-direction\_column {  
 flex-direction: column;  
}  
  
.m\_2rem {  
 margin: 2rem;  
}  
  
.form {  
 width: fit-content;  
 border: #ecdcfb 1px solid;  
 padding: 1rem;  
 border-radius: 10px;  
 display: grid;  
 gap: 5px;  
 position: sticky;  
 top: 20px;  
}  
  
table, th, td {  
 border: 1px solid black;  
 border-collapse: collapse;  
}  
  
td {  
 padding: 10px;  
}  
  
th {  
 padding: 15px;  
}

Результат:





**Висновок:** Навчилась працювати з state і props.