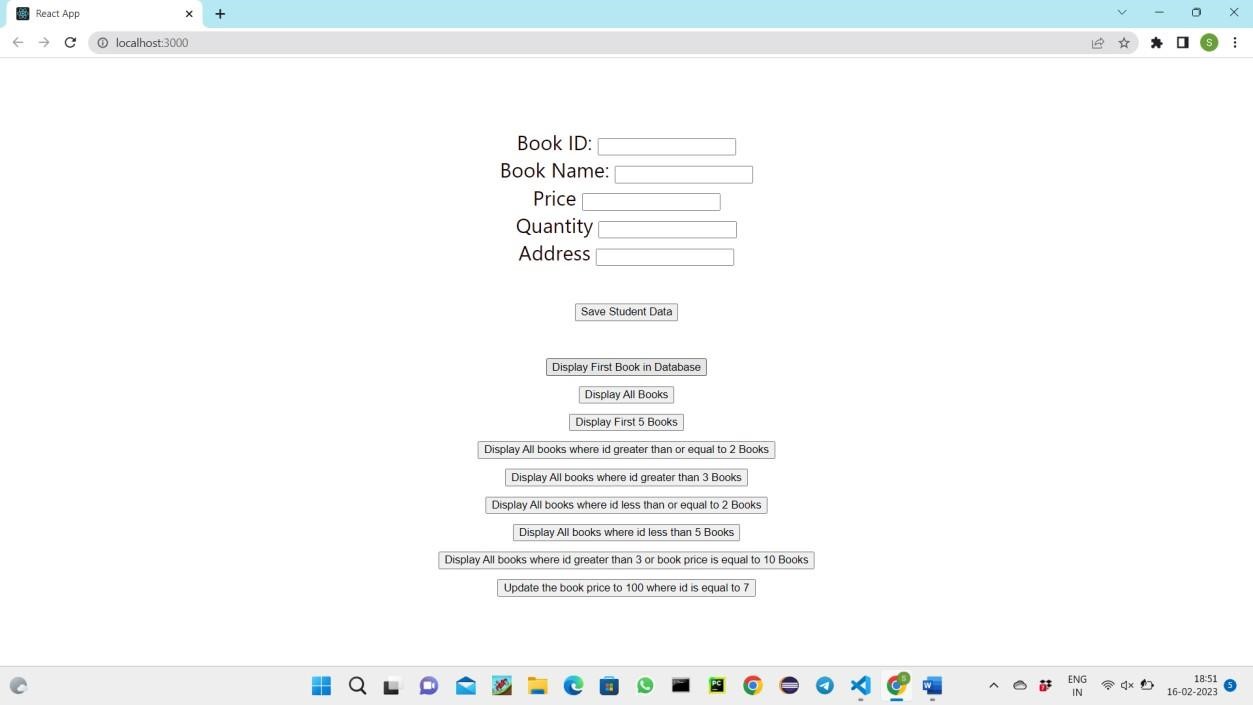
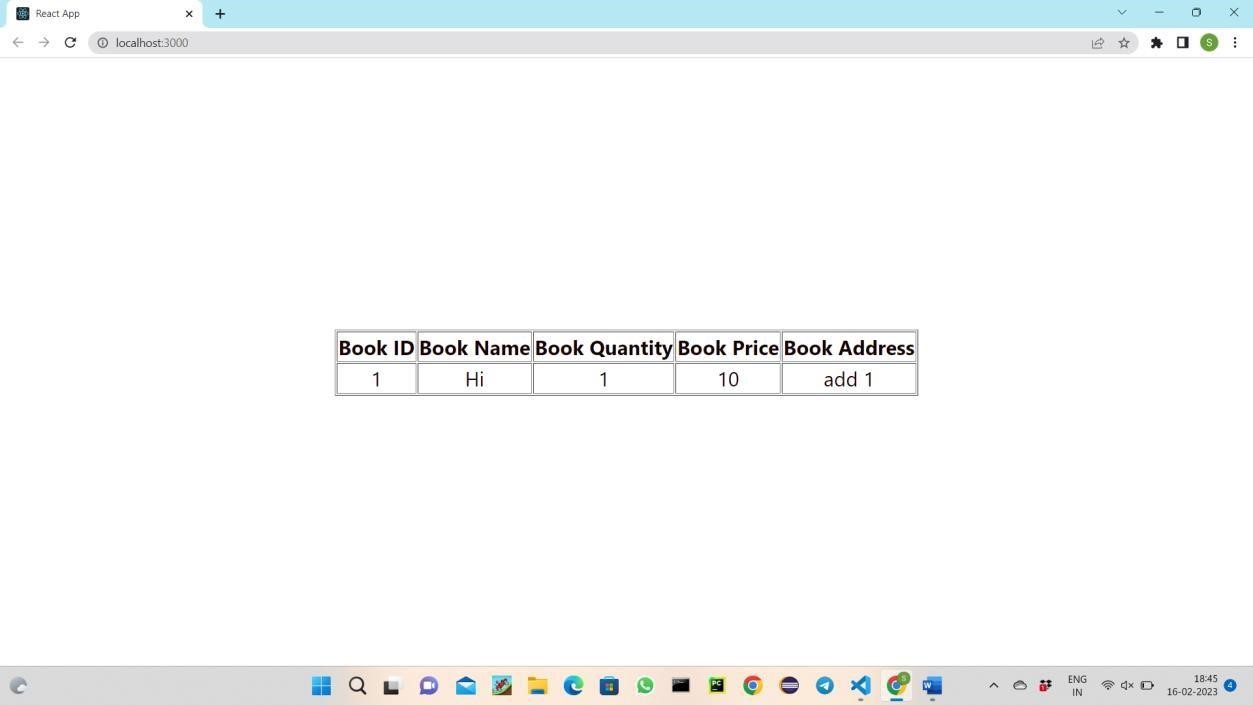
**SKILL-11**

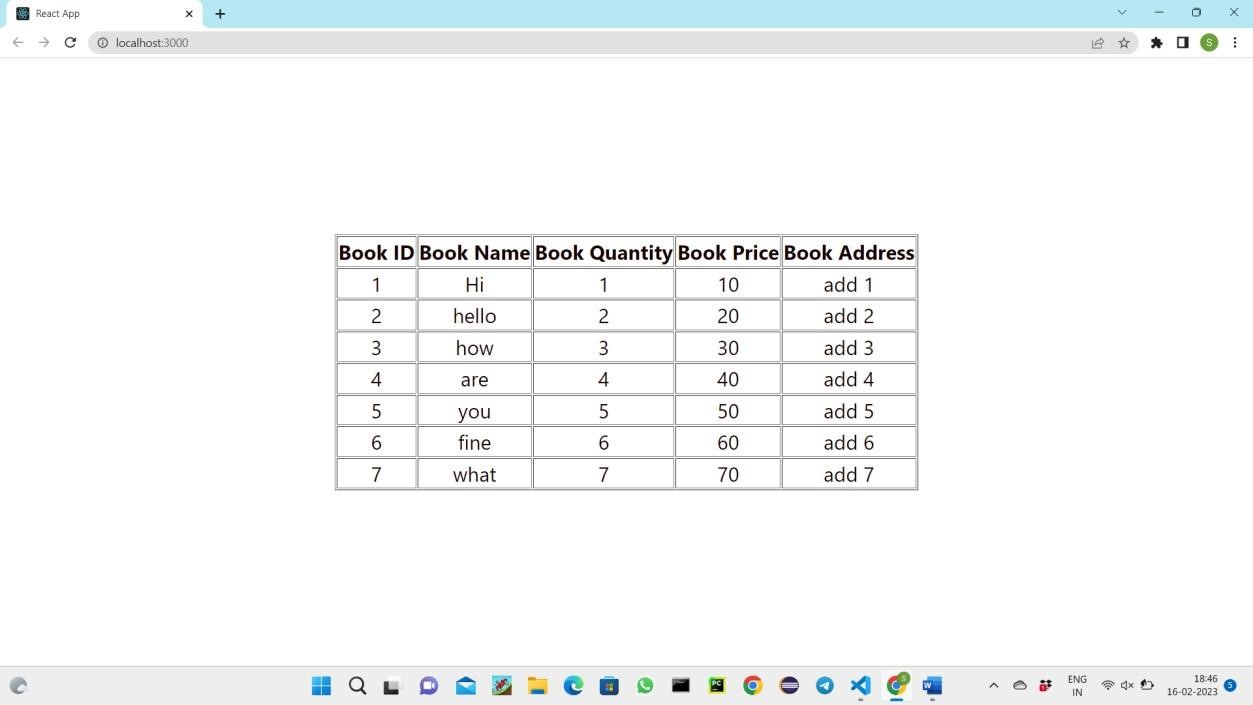
**ID: 2100030923**

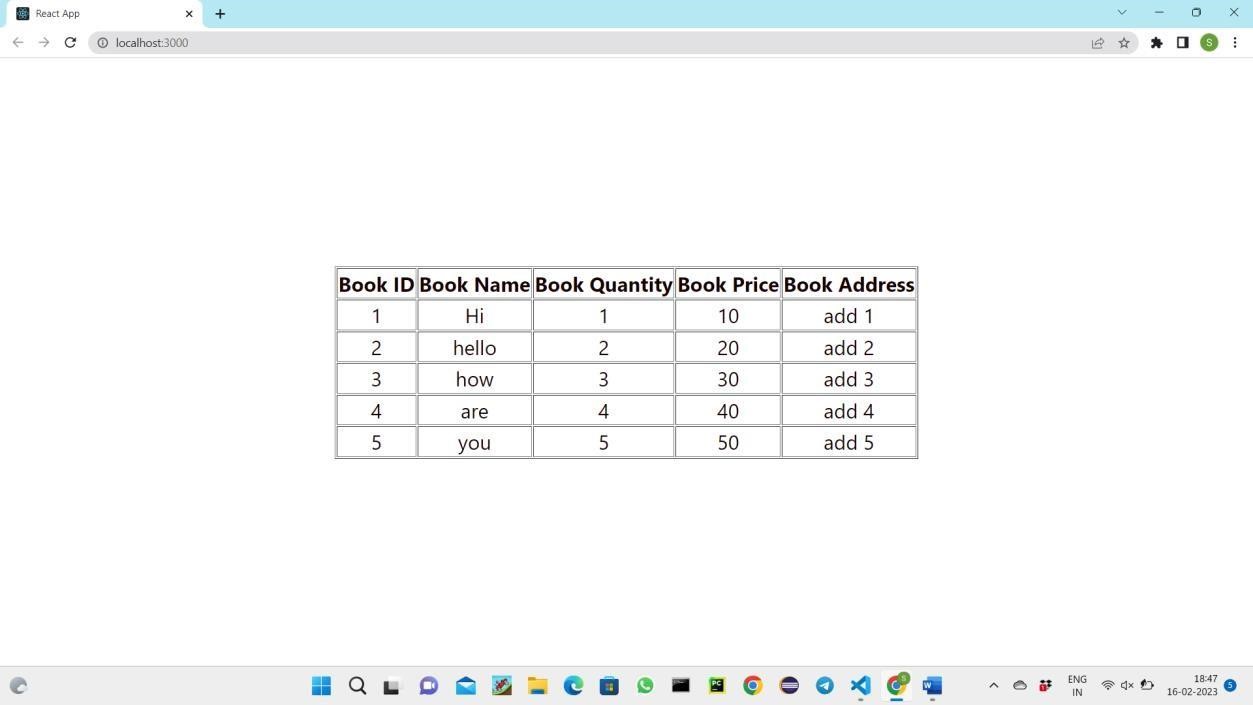
**Name: K V N K D Sai Chandana**

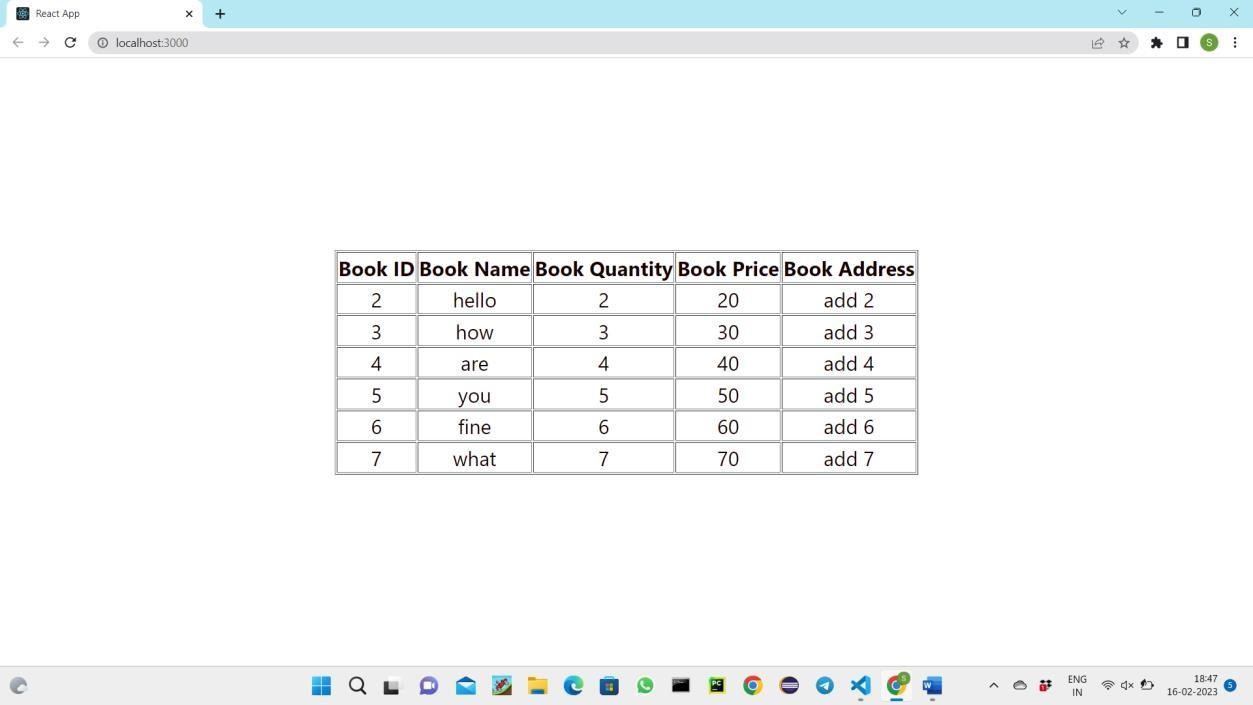
**OUTPUT:**

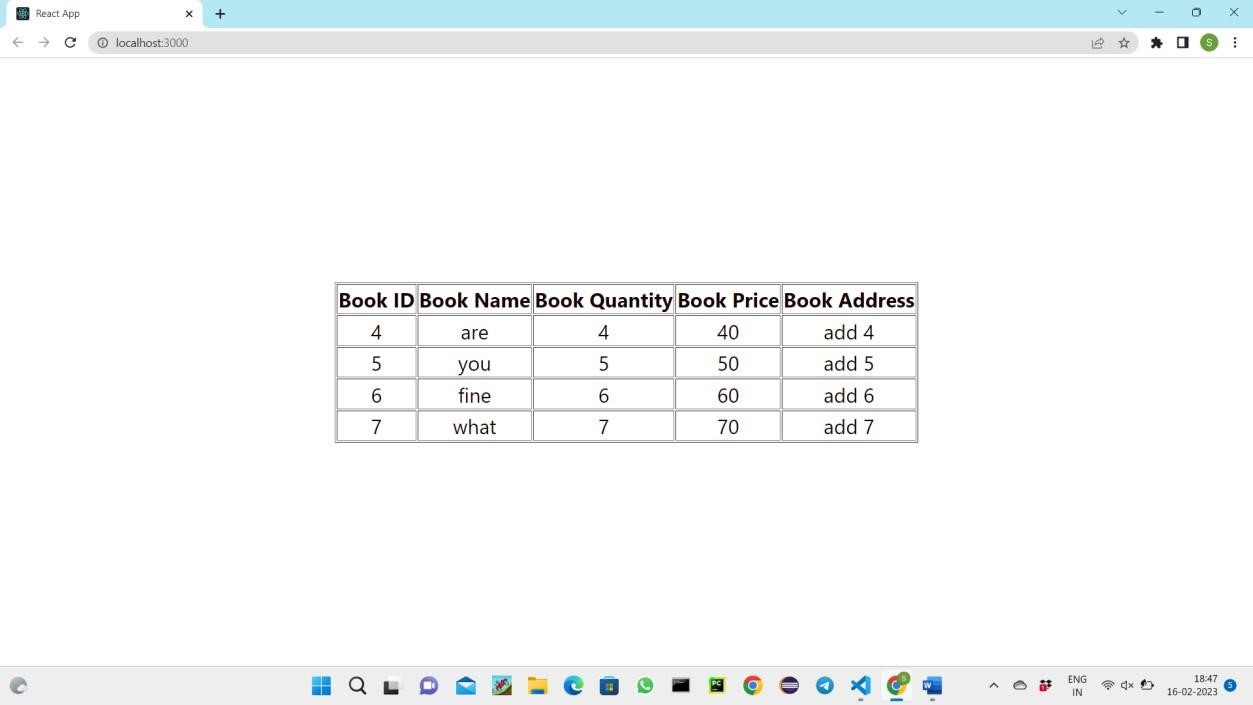


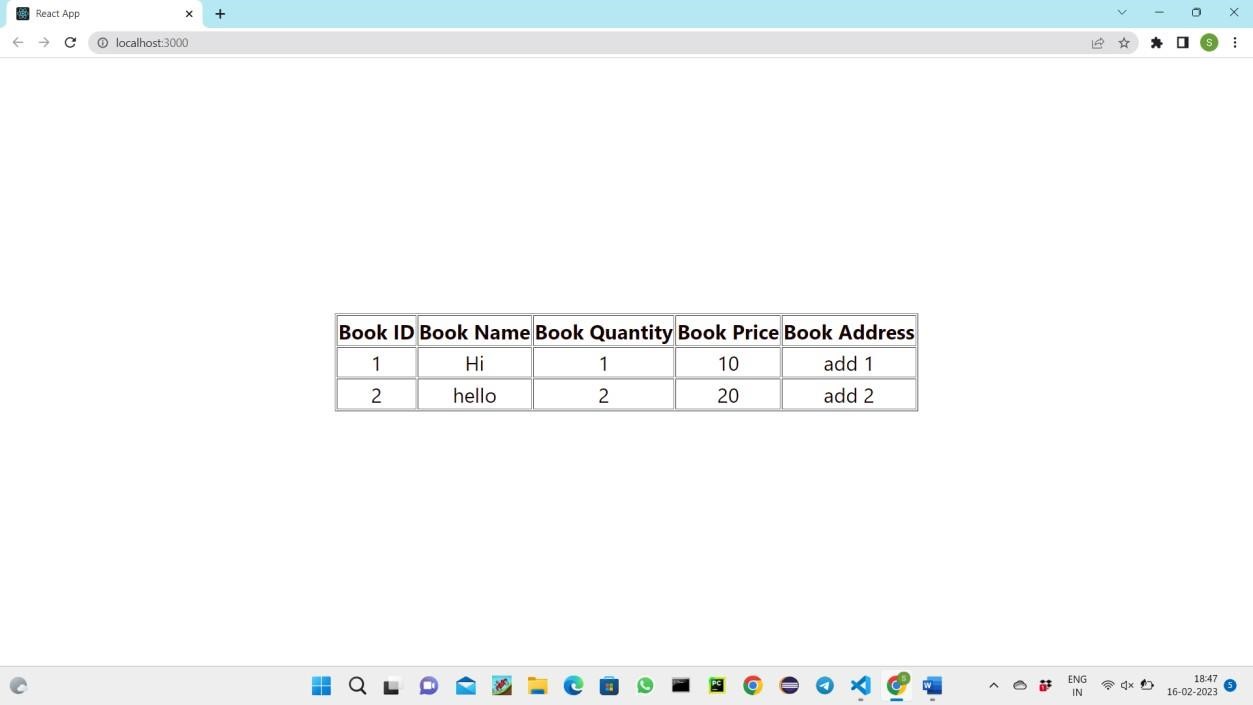


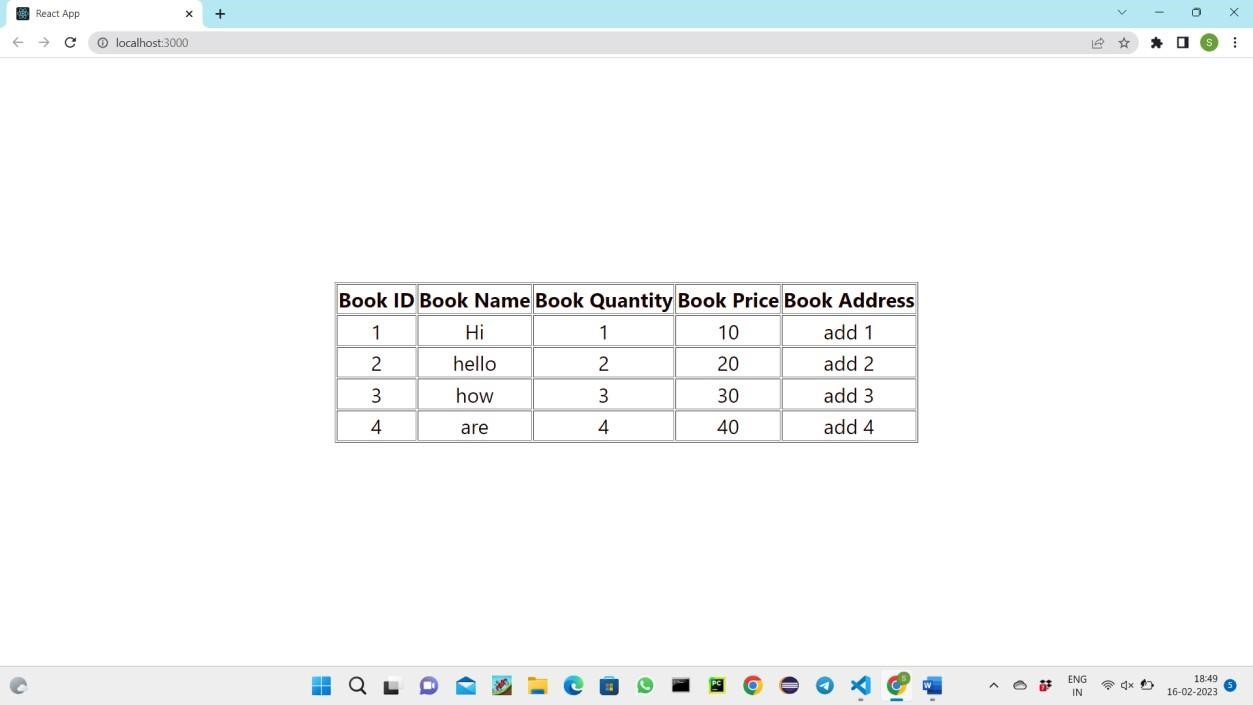


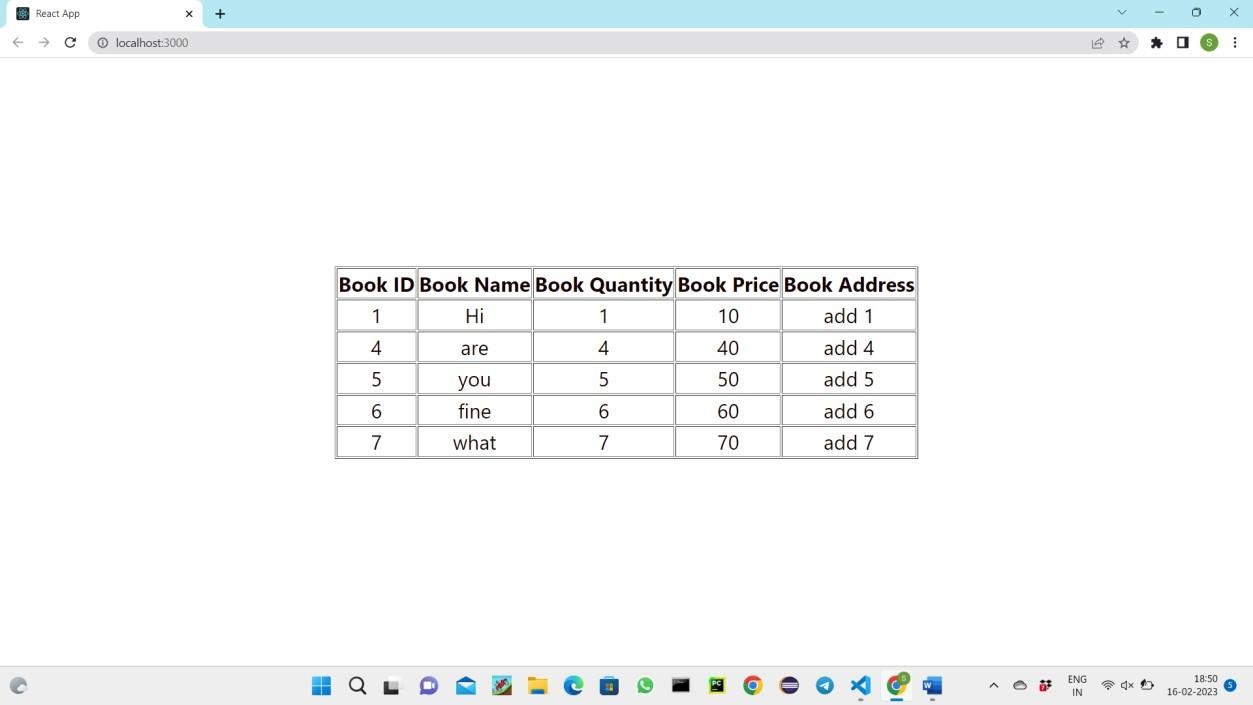


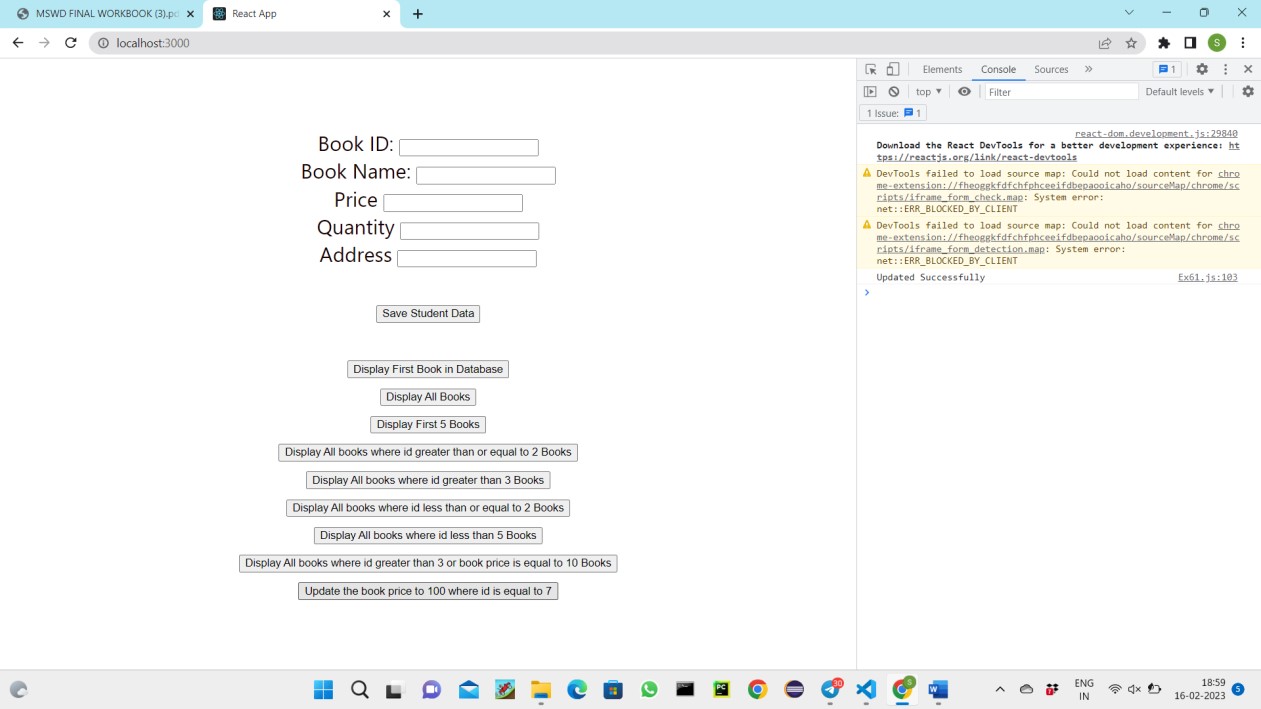


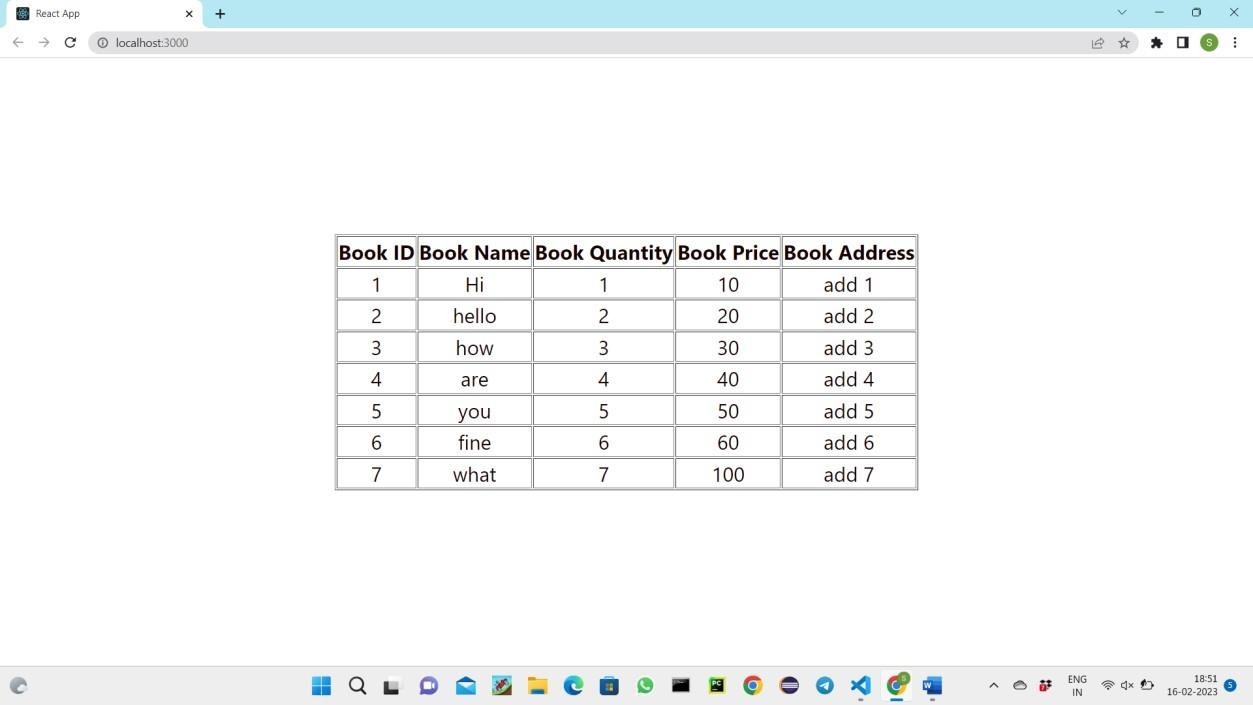












**App.js:**

|  |
| --- |
| // import logo from './logo.jpg'; // import klu from './klu.png'; import './App.css'; import Ex61 from './components/Ex61' import Ex63 from './components/Ex63' import { BrowserRouter, Routes, Route } from 'react-router-dom'; //npm i react-router-dom to install react-routerdom package in client cmd  //BrowserRouter, Routes and Route are used to route from one page to another like from signup to login etc.  // import { Container, Toolbar, AppBar } from '@material-ui/core'; //npm i  @material-ui/core --force to install @material-ui/core package in client cmd    function App() {  // function opetarion(){  // let vl=document.getElementById("Value");  // vl.value="1";  // } return(  <div className="App">    <div className='App-body'>  <BrowserRouter>  <Routes>  <Route path='/' element={<Ex61 />} />  {/\* <Route path='/ex63' element={<Ex63 />} /> \*/}  </Routes>  </BrowserRouter>  </div>  </div>    );    } export default  App; |

**App.css:**

.App { text-align: center;

}

.App-logo {

|  |
| --- |
| height: 20vmin; pointerevents: none;  }    .App-klu { height: 50px; width: 150px; pointer-events: none; margin-left: 80px; border-radius: 30px; margin-right: 5px; margin-bottom: 10px;  }    .App-kl{ display: flex; flexdirection: row; align-items: center; text-align: center;  }    /\* .App-header { backgroundcolor: #282c34; min-height: 20vh; display: flex; alignitems: center; justifycontent: center; font-size: calc(10px + 2vmin); color:  white;  } \*/    .App-Header{ fontsize: 20px; marginbottom: 10px;  }    .App-para{ fontsize: 15px; display: flex; flex-direction: row; align-items: center; text-align: center;  }    .App-color{ backgroundcolor: white;  /\* background-color: #15b1ff; \*/ borderradius: 30px; |

|  |
| --- |
| border :7px solid rgb(116, 116, 197); padding: 10px;  }    .App-para1{ display: flex; align-items: center; margin-left: 60px; color: blue; fontweight: bold; fontsize: 20px;  }    .App-para2{ display: flex; align-items: center; color: green; margin-left: 70px; font-weight: bold;  }    .App-para3{ display: flex; align-items: center; margin-left: 20px; color: green; fontweight: bold;  }    .App-para4{ display: flex; align-items: center; margin-left: 130px; color: red; fontweight: bold;  }    .App-body { backgroundcolor:white; min-height: 100vh; display: flex; align-items: center; justifycontent: center; font-size: calc(10px + 2vmin); color:  rgb(21, 1, 1);  }    .App-idcard { |

|  |
| --- |
|  |

/\* border:1px solid red; \*/

|  |
| --- |
| background-color: white; padding: 20px; height: 420px; width: 350px; display: block; align-items: center; justifycontent: center; font-size: calc(10px + 2vmin); color:  rgb(21, 1, 1);  /\* box-shadow: 10px 10px 5px rgb(60, 206, 255) inset; \*/ box-shadow: 0 4px 8px 0 rgb(146, 252, 16), 0 6px 20px 0 rgb(16, 186, 233); borderradius: 30px; border :7px solid rgb(70, 255, 53); }    .App-inputval{ width:200px; height: 50px; margin-bottom:  10px;  }    .App-plus{ height: 50px; width: 50px;  }  .App-zero{ height: 50px; width: 50px; margin-right: 50px;  margin-left: 50px;  }    .App-link { color: #61dafb;  }    @keyframes App-logo-spin { from { transform:  rotate(0deg);  } to { transform:  rotate(360deg);  }  } |

**Ex61.js:**

|  |
| --- |
| import { RepeatOneSharp } from "@mui/iconsmaterial"; import axios from "axios"; import { useState } from "react"; function Ex61()  { const [result,setResult] = useState(null); const isDict = dict => { return typeof dict === "object" && !Array.isArray(dict); }; function handleSubmit(event) { event.preventDefault(); const data = new FormData(event.currentTarget); axios.post('http://localhost:8086/insert\_book', { book\_id: parseInt(data.get("book\_id")), book\_name: data.get("book\_name"), book\_price: data.get("book\_price"), book\_quantity: data.get("book\_quantity"), book\_address:  data.get("book\_address") }).then((resposnse)=>{  console.log(resposnse.data);  })  } function showOne() {  axios.get('http://localhost:8086/show\_one\_book', {params:{  }}).then((response)=>{ console.log(response.data) setResult(response.data)  }) } function showAll() { axios.get('http://localhost:8086/show\_all\_book', {params:{  }}).then((response)=>{ console.log(response.data) setResult(response.data)  })  } function showAlllte5() {  axios.get('http://localhost:8086/show\_all\_lte5\_book', {params:{ |

}}).then((response)=>{

|  |
| --- |
| console.log(response.data) |
| setResult(response.data)  })  } function showAllgte2() {  axios.get('http://localhost:8086/show\_all\_gte2\_book', {params:{  }}).then((response)=>{ console.log(response.data) setResult(response.data)  })  } function showAllgt3() {  axios.get('http://localhost:8086/show\_all\_gt3\_book', {params:{  }}).then((response)=>{ console.log(response.data) setResult(response.data)  })  } function showAlllte2() {  axios.get('http://localhost:8086/show\_all\_lte2\_book', {params:{  }}).then((response)=>{ console.log(response.data) setResult(response.data)  })  } function showAlllt5() { axios.get('http://localhost:8086/show\_all\_lt5\_book', {params:{  }}).then((response)=>{ console.log(response.data) setResult(response.data)  })  } function showAllgt3orbookpriceet10() {  axios.get('http://localhost:8086/show\_all\_gt3\_or\_bookprice\_equalto10\_b ook', {params:{    }}).then((response)=>{ |

console.log(response.data)

|  |
| --- |
| setResult(response.data) |

|  |
| --- |
| })  } function updateid7() { console.log("upd") axios.patch('http://localhost:8086/show\_all\_update', {params:{  }}).then((response)=>{ console.log(response.data)  // setResult(response.data)  })  } if(result == null) { return(  <div>  <form onSubmit={handleSubmit}>  Book ID: <input type="text" name="book\_id" />  <br/>  Book Name: <input type="text" name="book\_name" />  <br/>  Price <input type="text" name="book\_price" /> <br/>  Quantity <input type="text" name="book\_quantity" />  <br/>  Address <input type="text" name="book\_address" /> <br/><br/>  <input type="submit" value="Save Student Data" />  </form>  <br/>  <button onClick={()=>showOne()}>Display First Book in  Database</button>  <br/>  <button onClick={()=>showAll()}>Display All Books</button> <br/>  <button onClick={()=>showAlllte5()}>Display First 5 Books</button>  <br/>  <button onClick={()=>showAllgte2()}>Display All books where id greater than or equal to 2 Books</button>  <br/>  <button onClick={()=>showAllgt3()}>Display All books where id greater than 3 Books</button>  <br/>  <button onClick={()=>showAlllte2()}>Display All books where id less than or equal to 2 Books</button>  <br/> |

<button onClick={()=>showAlllt5()}>Display All books where id

|  |
| --- |
| less than 5 Books</button>  <br/>  <button onClick={()=>showAllgt3orbookpriceet10()}>Display All books where id greater than 3 or book price is equal to 10 Books</button>  <br/>  <button onClick={()=>updateid7()}>Update the book price to 100 where id is equal to 7 </button>  <br/>  </div>  ); } else { if(isDict(result)) { return (  <div>  <table border="1">  <tr>  <th> Book ID </th>  <th> Book Name </th>  <th> Book Quantity </th>  <th> Book Price </th>  <th> Book Address </th>  </tr>  <tr>  <td> {result.book\_id} </td>  <td> {result.book\_name} </td>  <td> {result.book\_quantity} </td>  <td> {result.book\_price} </td>  <td> {result.book\_address} </td>  </tr>  </table>  </div>  );  } else { console.log("java"); return (  <div>  <table border="1">  <tr>  <th> Book ID </th>  <th> Book Name </th>  <th> Book Quantity </th>  <th> Book Price </th>  <th> Book Address </th> |

</tr> {result.map((obj)=>{

return(

<tr>

<td> {obj.book\_id} </td>

<td> {obj.book\_name} </td>

<td> {obj.book\_quantity} </td>

<td> {obj.book\_price} </td>

<td> {obj.book\_address} </td>

</tr>

);

})}

</table>

</div>

);

}

}

} export default Ex61;

**Server.js:**

|  |
| --- |
| const express = require('express') const cors = require('cors') const { MongoClient } = require('mongodb') const { request, response } = require('express') const app = express() app.use(cors()) app.use(express.json()) const uri  =  "mongodb+srv://admin:rocky9886@cluster0.nczlwja.mongodb.net/?retryWrites=true& w=majority" const client = new MongoClient(uri) client.connect()  //Experiment 6-1 var db = client.db("ex6")  var col = db.collection("costumer")    app.get('/',(request,response) => { response.send('This is a Server')  }) app.post('/insert\_book', (request,response) => { |

col = db.collection("costumer")

|  |
| --- |
| console.log(request.body) |
| col.insertOne(request.body) response.send(request.body)  }) app.get('/show\_one\_book', (req, res)=>{ col = db.collection("costumer") console.log(req.query) async function find(){ try{ const result=await col.findOne() console.log(result) if(result  == null)  { res.send({"Data\_Retrieval":"Fail"})  } else{ res.send(result)  }  } finally{} } find().catch(console.dir)  }) app.get('/show\_all\_book', (req, res)=>{ col = db.collection("costumer") console.log(req.query) async function find(){ try{ const result=await col.find().toArray()  console.log(result) if(result == null)  { console.log("sever"); res.send({"Data\_Retrieval":"Fail"})  } else{ res.send(result)  }  } finally{} }  find().catch(console.dir)  }) |

app.get('/show\_all\_lte5\_book', (req, res)=>{

|  |
| --- |
| col = db.collection("costumer") console.log(req.query) |

|  |
| --- |
| async function find(){ try{ const result=await col.find({book\_id:{$lte:5}}).toArray() console.log(result) if(result == null)  { res.send({"Data\_Retrieval":"Fail"})  } else{ res.send(result)  }  } finally{} } find().catch(console.dir)  })  app.get('/show\_all\_gte2\_book', (req, res)=>{ col = db.collection("costumer") console.log(req.query) async function find(){ try{ const result=await col.find({book\_id:{$gte:2}}).toArray() console.log(result) if(result == null)  { res.send({"Data\_Retrieval":"Fail"})  } else{ res.send(result)  }  } finally{}  } find().catch(console.dir)  }) app.get('/show\_all\_gt3\_book', (req, res)=>{ col = db.collection("costumer") console.log(req.query) async function find(){ try{ const result=await col.find({book\_id:{$gt:3}}).toArray() console.log(result) if(result == null) { |

|  |
| --- |
| res.send({"Data\_Retrieval":"Fail"}) } else{ res.send(result)  }  } finally{} }  find().catch(console.dir)  }) app.get('/show\_all\_lte2\_book', (req, res)=>{ col = db.collection("costumer") console.log(req.query) async function find(){ try{ const result=await col.find({book\_id:{$lte:2}}).toArray() console.log(result) if(result == null)  { res.send({"Data\_Retrieval":"Fail"})  } else{ res.send(result)  }  } finally{} }  find().catch(console.dir)  })  app.get('/show\_all\_lt5\_book', (req, res)=>{ col = db.collection("costumer") console.log(req.query) async function find(){ try{ const result=await col.find({book\_id:{$lt:5}}).toArray() console.log(result) if(result == null)  { res.send({"Data\_Retrieval":"Fail"}) } else{ res.send(result)  }  } finally{}  } |

|  |
| --- |
| find().catch(console.dir)  }) app.get('/show\_all\_gt3\_or\_bookprice\_equalto10\_book', (req, res)=>{ col = db.collection("costumer") console.log(req.query) async function find(){ try{ const result=await col.find({$or:[{book\_price:"10"},{book\_id:{$gt:3}}]}).toArray() console.log(result) if(result == null)  { res.send({"Data\_Retrieval":"Fail"})  } else{ res.send(result)  }  } finally{} }  find().catch(console.dir)  }) app.patch('/show\_all\_update',(req, res)=>{ console.log("update") col  = db.collection("costumer")  console.log(req.body) const doc = {  $set: {book\_price:"100"}  } col.updateOne({book\_id:7},doc) res.send("Updated Successfully")    // console.log(req.query)  // async function find(){  // try{  // const result=await col.find().toArray()  // console.log(result)  // if(result == null)  // {  // console.log("sever");  // res.send({"Data\_Retrieval":"Fail"})  // }  // else{  // res.send(result)  // }  // } |

// finally{}

// }

// find().catch(console.dir)

})

app.listen(8086) console.log("server started")