TUGAS 2



MUHAMMAD AUDYA FADHLURROHMAN

202310041

PARIWISATA DAN INFORMATIKA

TEKNOLOGI INFORMASI

INSTITUT BISNIS DAN INFORMATIKA KESATUAN

|  |  |
| --- | --- |
| **Script** | **Screen Capture** |
| Widget1 Class Component  import React, { Component } from "react";  import Layout from "./Layout";  class Widget1 extends Component {  constructor(props) {  super(props);  this.state = {  title: "My Apps",  };  }  input = (value) => {  this.setState({ title: value });  };  render() {  return (  <Layout title={this.state.title}>  <form method="post" autoComplete="off" className="form w-100">  <div className="text-center mb-10">  <h1 className="text-dark mb-3">Widget One</h1>  <div className="text-gray-400 fw-bold fs-5">Please fill up this form with correctly</div>  </div>  <div className="fv-row mb-10 fv-plugins-icon-container">  <label className="form-label fs-6 fw-bolder text-dark">Type the title of this web</label>  <input type="text" name="title" className="form-control form-control-lg form-control-solid" defaultValue={this.state.title} onChange={(e) => this.input(e.target.value)} />  </div>  <div className="text-center mt-4">  <button type="button" className="btn btn-lg btn-primary w-100 mb-5">  Submit  </button>  </div>  </form>  </Layout>  );  }  }  export default Widget1; |  |
| FormsMultiple.js  import React, { useState } from "react";  import { Modal } from "react-bootstrap";  import DatePicker from "react-datepicker";  import "react-datepicker/dist/react-datepicker.css";  export default function FormMultiple() {  var today = new Date();  const currDate = formatDate(today);  const objProfile = {  npm: 0,  fname: "",  mname: "",  lname: "",  birthdate: currDate,  };  const [postData, setPostData] = useState([objProfile]);  const [mymodal, setMymodal] = useState({  show: false,  message: "",  });  const modalHandleClose = () => {  setMymodal({ ...mymodal, show: false });  };  const RowItem = ({ data, index }) => {  return (  <tr>  <td>  <input  type="text"  name="npm"  className="form-control"  placeholder="type numeric"  pattern="^[0-9]\*$"  defaultValue={data.npm ? data.npm : ""}  onMouseLeave={(e) =>  setPostData((prev) => {  const newData = [...prev];  newData[index].npm = parseInt(e.target.value);  return newData;  })  }  />  </td>  <td>  <input  type="text"  name="fname"  className="form-control"  placeholder="enter firstname"  defaultValue={data.fname}  onMouseLeave={(e) =>  setPostData((prev) => {  const newData = [...prev];  newData[index].fname = e.target.value;  return newData;  })  }  />  </td>  <td>  <input  type="text"  name="mname"  className="form-control"  placeholder="enter middlename"  defaultValue={data.mname}  onMouseLeave={(e) =>  setPostData((prev) => {  const newData = [...prev];  newData[index].mname = e.target.value;  return newData;  })  }  />  </td>  <td>  <input  type="text"  name="lname"  className="form-control"  placeholder="enter lastname"  defaultValue={data.lname}  onMouseLeave={(e) =>  setPostData((prev) => {  const newData = [...prev];  newData[index].lname = e.target.value;  return newData;  })  }  />  </td>  <td>  <DatePicker  selected={new Date(data.birthdate)}  onChange={(value) =>  setPostData((prev) => {  const newData = [...prev];  newData[index].birthdate = formatDate(value);  return newData;  })  }  className="form-control"  dateFormat="yyyy-MM-dd"  placeholderText="Choose Date"  />  </td>  {index !== 0 ? (  <td>  <button  className="btn btn-icon btn-sm btn-danger"  type="button"  onClick={(e) =>  setPostData((prev) => {  const newData = [...prev];  delete newData[index];  return newData;  })  }>  <i className="bi bi-trash"></i>  </button>  </td>  ) : (  ""  )}  </tr>  );  };  const resultPost = (obj) => {  return (  <div className="result">  {obj.map((v, index) =>  v ? (  <div className="border-bottom mb-3" key={index}>  <p>  <span className="fw-bolder">NPM:</span> {v.npm}  </p>  <p>  <span className="fw-bolder">Fullname:</span> {v.fname} {v.mname} {v.lname}{" "}  </p>  <p>  <span className="fw-bolder">Birthdate:</span> {v.birthdate}{" "}  </p>  <p>  <span className="fw-bolder">Age:</span> {calculateAge(v.birthdate)} years old{" "}  </p>  </div>  ) : (  ""  )  )}  </div>  );  };  const handlerSubmit = (e) => {  e.preventDefault();  let myresult = resultPost(postData);  setMymodal({ ...mymodal, show: true, message: myresult });  };  return (  <div id="form-multiple" className="card mb-5 mb-xl-10">  <div className="card-header">  <div className="card-title">  <h3>Widget Two</h3>  </div>  </div>  <div className="card-body p-0">  <form method="post" autoComplete="off" onSubmit={(e) => handlerSubmit(e)}>  <table className="table table-flush align-middle table-row-bordered table-row-solid gy-4 gs-9">  <thead className="border-gray-200 fs-5 fw-bold bg-lighten">  <tr>  <th>NPM</th>  <th>First Name</th>  <th>Middle Name</th>  <th>Last Name</th>  <th>Birthdate</th>  <th></th>  </tr>  </thead>  <tbody>{postData.map((v, index) => (v ? <RowItem key={index} data={v} index={index} /> : ""))}</tbody>  <tfoot>  <tr>  <td colSpan={5}>  <button  className="btn btn-sm btn-light-primary"  type="button"  onClick={() =>  setPostData((prev) => {  return [...prev, objProfile];  })  }>  <i className="bi bi-plus-circle me-2"></i>  Add new row  </button>  </td>  </tr>  </tfoot>  </table>  <div className="text-end p-10">  <div className="btn-group">  <button className="btn btn-lg btn-secondary" type="button">  Clear  </button>  <button className="btn btn-lg btn-primary" type="submit">  Save  </button>  </div>  </div>  </form>  </div>  <ModalPopUP show={mymodal.show} handleClose={modalHandleClose} message={mymodal.message} />  </div>  );  }  const formatDate = (datestring) => {  const today = new Date(datestring);  const yyyy = today.getFullYear();  let mm = today.getMonth() + 1; // Months start at 0!  let dd = today.getDate();  if (dd < 10) dd = "0" + dd;  if (mm < 10) mm = "0" + mm;  return yyyy + "-" + mm + "-" + dd;  };  const calculateAge = (birthdate) => {  var today = new Date();  var birthDate = new Date(birthdate);  var age = today.getFullYear() - birthDate.getFullYear();  var m = today.getMonth() - birthDate.getMonth();  if (m < 0 || (m === 0 && today.getDate() < birthDate.getDate())) {  age--;  }  return age;  };  const ModalPopUP = ({ show, handleClose, message }) => {  return (  <Modal show={show} onHide={handleClose}>  <Modal.Header closeButton>  <Modal.Title>Information</Modal.Title>  </Modal.Header>  <Modal.Body>{message}</Modal.Body>  </Modal>  );  };  ============================  Layout.js  import React from "react";  export default function Layout(props) {  return (  <div className="d-flex flex-column flex-root vh-100" style={{ backgroundColor: "purple" }}>  <div className="d-flex flex-column flex-column-fluid">  <div className="d-flex flex-center flex-column flex-column-fluid p-10 pb-lg-20">  <h1 className="mb-12 text-white text-center my-4">{props.title}</h1>  <div className="w-50 bg-body rounded shadow-sm p-4 p-lg-15 mx-auto">{props.children}</div>  </div>  </div>  </div>  );  } |  |