**ФОРМА РЕГИСТРАЦИИ**

import React from 'react';

// BEGIN (write your solution here)

export default class MyForm extends React.Component {

  constructor(props) {

    super(props);

    this.state = {

      form: {

        email: '',

        password: '',

        address: '',

        city: '',

        country: '',

        acceptRules: false,

      },

      submition: 'filingForm',

    };

  }

  handleSubmit = (e) => {

    e.preventDefault();

    this.setState({ submition: 'showTable' });

  };

  handleBack = () => {

    this.setState({ submition: 'filingForm' });

  };

  handleChange = ({ target }) => {

    const { form } = this.state;

    const value = target.type === 'checkbox' ? target.checked : target.value;

    this.setState({ form: { ...form, [target.name]: value } });

  };

  renderResult() {

    const { form } = this.state;

    return (

      <div>

        <button type="button" className="btn btn-primary" onClick={this.handleBack}>Back</button>

        <table className="table">

          <tbody>

            <tr>

              <td>acceptRules</td>

              <td>{form.acceptRules.toString()}</td>

            </tr>

            <tr>

              <td>address</td>

              <td>{form.address}</td>

            </tr>

            <tr>

              <td>city</td>

              <td>{form.city}</td>

            </tr>

            <tr>

              <td>country</td>

              <td>{form.country}</td>

            </tr>

            <tr>

              <td>email</td>

              <td>{form.email}</td>

            </tr>

            <tr>

              <td>password</td>

              <td>{form.password}</td>

            </tr>

          </tbody>

        </table>

      </div>

    );

  }

  renderForm() {

    const { form } = this.state;

    return (

      <form name="myForm" onSubmit={this.handleSubmit}>

        <div className="col-md-6 mb-3">

          <label htmlFor="email" className="col-form-label">Email</label>

          <input

            type="email"

            name="email"

            className="form-control"

            id="email"

            placeholder="Email"

            onChange={this.handleChange}

            value={form.email}

          />

        </div>

        <div className="form-group col-md-6">

          <label htmlFor="password" className="col-form-label">Password</label>

          <input

            type="password"

            name="password"

            className="form-control"

            id="password"

            placeholder="Password"

            onChange={this.handleChange}

            value={form.password}

          />

        </div>

        <div className="col-md-6 mb-3">

          <label htmlFor="address" className="col-form-label">Address</label>

          <textarea

            type="text"

            className="form-control"

            name="address"

            id="address"

            placeholder="1234 Main St"

            onChange={this.handleChange}

            value={form.address}

          />

        </div>

        <div className="col-md-6 mb-3">

          <label htmlFor="city" className="col-form-label">City</label>

          <input

            type="text"

            className="form-control"

            name="city"

            id="city"

            onChange={this.handleChange}

            value={form.city}

          />

        </div>

        <div className="col-md-6 mb-3">

          <label htmlFor="country" className="col-form-label">Country</label>

          <select

            id="country"

            name="country"

            className="form-control"

            onChange={this.handleChange}

            value={form.country}

          >

            <option value="">Choose</option>

            <option value="argentina">Argentina</option>

            <option value="russia">Russia</option>

            <option value="china">China</option>

          </select>

        </div>

        <div className="col-md-6 mb-3">

          <div className="form-check">

            <label className="form-check-label" htmlFor="rules">

              <input

                id="rules"

                type="checkbox"

                name="acceptRules"

                className="form-check-input"

                onChange={this.handleChange}

                checked={form.acceptRules}

              />

              Accept Rules

            </label>

          </div>

        </div>

        <button type="submit" className="btn btn-primary">Sign in</button>

      </form>

    );

  }

  render() {

    const { submition } = this.state;

    switch (submition) {

      case 'filingForm':

        return this.renderForm();

      case 'showTable':

        return this.renderResult();

      default:

        throw new Error('What a FACK!!');

    }

  }

}

РЕШЕНИЕ УЧИТЕЛЯ

export default class MyForm extends React.Component {

  constructor(props) {

    super(props);

    this.state = {

      form: {

        email: '',

        password: '',

        city: '',

        country: '',

        address: '',

        acceptRules: false,

      },

      submittingState: 'fillingForm',

    };

  }

  handleChangeField = ({ target }) => {

    const { form } = this.state;

    const value = target.type === 'checkbox' ? target.checked : target.value;

    this.setState({ form: { ...form, [target.name]: value } });

  }

  handleBackToForm = () => {

    this.setState({ submittingState: 'fillingForm' });

  }

  handleSubmitForm = (e) => {

    e.preventDefault();

    this.setState({ submittingState: 'submitted' });

  }

  renderRow = (key) => {

    const { form } = this.state;

    return (

      <tr key={key}>

        <td>{key}</td>

        <td>{form[key].toString()}</td>

      </tr>

    );

  };

  renderResult() {

    const { form } = this.state;

    const keys = Object.keys(form).sort();

    return (

      <div>

        <button type="button" className="btn btn-primary" onClick={this.handleBackToForm}>Back</button>

        <table className="table">

          <tbody>

            {keys.map(this.renderRow)}

          </tbody>

        </table>

      </div>

    );

  }

  renderForm() {

    const { form } = this.state;

    return (

      <form onSubmit={this.handleSubmitForm} name="myForm">

        <div className="col-md-6 mb-3">

          <label htmlFor="email" className="col-form-label">Email</label>

          <input

            type="email"

            name="email"

            onChange={this.handleChangeField}

            value={form.email}

            className="form-control"

            id="email"

            placeholder="Email"

          />

        </div>

        <div className="col-md-6 mb-3">

          <label htmlFor="password" className="col-form-label">Password</label>

          <input

            type="password"

            onChange={this.handleChangeField}

            value={form.password}

            name="password"

            className="form-control"

            id="password"

            placeholder="Password"

          />

        </div>

        <div className="col-md-6 mb-3">

          <label htmlFor="address" className="col-form-label">Address</label>

          <textarea type="text" name="address" value={form.address} onChange={this.handleChangeField} className="form-control" id="address" placeholder="1234 Main St" />

        </div>

        <div className="col-md-6 mb-3">

          <label htmlFor="city" className="col-form-label">City</label>

          <input type="text" name="city" onChange={this.handleChangeField} value={form.city} className="form-control" id="city" />

        </div>

        <div className="col-md-6 mb-3">

          <label htmlFor="country" className="col-form-label">Country</label>

          <select id="country" name="country" onChange={this.handleChangeField} className="form-control" value={form.country}>

            <option value="">Choose</option>

            <option value="argentina">Argentina</option>

            <option value="russia">Russia</option>

            <option value="china">China</option>

          </select>

        </div>

        <div className="col-md-6 mb-3">

          <div className="form-check">

            <label className="form-check-label" htmlFor="rules">

              <input id="rules" name="acceptRules" className="form-check-input" onChange={this.handleChangeField} type="checkbox" checked={form.acceptRules} />

              Accept Rules

            </label>

          </div>

        </div>

        <button type="submit" className="btn btn-primary">Sign in</button>

      </form>

    );

  }

  render() {

    const { submittingState } = this.state;

    switch (submittingState) {

      case 'fillingForm':

        return this.renderForm();

      case 'submitted':

        return this.renderResult();

      default:

        throw new Error(`'${submittingState}' - unknown state`);

    }

  }

}

**Удаление компонентов**

class List extends React.Component {

removeItem = (id) => (e) => {

e.preventDefault();

const newItems = this.state.items.filter((item) => item.id !== id);

this.setState({ items: newItems });

};

constructor(props) {

super(props);

const items = [1, 2, 3, 4, 5].map((item, index) => ({ item, id: index + 1 }));

this.state = { items };

}

renderItem({ id }) {

return <li><a href="#" onClick={this.removeItem(id)}>{id}</a></li>;

}

render() {

return <ul>

{this.state.items.map((item) => this.renderItem(item))}

</ul>;

}

}

**Использование immutability-helper**

**const** index **=** 5;

**const** newItems **=** update(items, {$splice: [[index, 1]]});

**КНОПКИ + -**

export default class Component extends React.Component {

  constructor(props) {

    super(props);

    this.state = { items: [] };

  }

  handleCount = (value) => {

    const { items } = this.state;

    const currentValue = get(items, [0, 'value'], 0) + value;

    const current = { id: uniqueId(), value: currentValue };

    this.setState({ items: [current, ...items] });

  }

  handleDec = () => this.handleCount(-1);

  handleInc = () => this.handleCount(1);

  handleRemove = (currentId) => () => {

    const { items } = this.state;

    this.setState({ items: items.filter(({ id }) => id !== currentId) });

  }

  renderLog() {

    const { items } = this.state;

    if (items.length === 0) {

      return null;

    }

    return (

      <div className="list-group">

        {items.map(({ id, value }) => (

          <button type="button" className="list-group-item list-group-item-action" key={id} onClick={this.handleRemove(id)}>

            {value}

          </button>

        ))}

      </div>

    );

  }

  render() {

    return (

      <div>

        <div className="btn-group font-monospace" role="group">

          <button type="button" className="btn btn-outline-success" onClick={this.handleInc}>+</button>

          <button type="button" className="btn btn-outline-danger" onClick={this.handleDec}>-</button>

        </div>

        {this.renderLog()}

      </div>

    );

  }

}