

Front End Bootcamp PT. Mahardika Solusi Teknologi

Ariel Arliyanus, S.Kom

Chapter 15 – React JS Lists & Forms

2021 © All Right Reserved

Dilarang memperbanyak dan/atau meng-copy sebagian atau seluruh material dalam dokumen ini tanpa persetujuan tertulis dari PT. Mahardika Solusi Teknologi

Lists & Forms

Dalam sebuah aplikasi tentunya kita sering menemukan sebuah list atau table yang menampilkan data-data dan disana kita bisa mengolah datanya dengan menambahkan data, menghapus data dan update data. tentunya di reactjs pun perlu hal tersebut karna itu merupakan hal yang lumrah dalam aplikasi. lalu bagaimana cara mengimplementasikannya? ya mari kita simak pemaparan di bawah ini

Lists

lists atau daftar, atau sering juga kita melihat daftar dalam bentuk table merupakan salah satu opsi untuk menampilkan data, nah mari kita coba dengan kasus di bawah ini

misal kita memiliki data array of object seperti di bawah ini:

```
let pesertaLomba = [ 'Budi', 'Susi', 'Lala', 'Agung' ]
```

lalu kita ingin menampilkan data tersebut kita olah di masukkan ke dalam state lalu kita tampilkan dalam tabel. berikut contoh kode penggunaanya:

```
class Lists extends Component{
  constructor(props){
    super(props)
    this.state = {
      pesertaLomba : [ 'Budi', 'Susi', 'Lala', 'Agung' ]
    }
  }

  render(){
    return(
      <>
        <h1>Daftar Peserta Lomba</h1>
        <table>
          <thead>
            <tr>
              <th>No</th>
              <th>Nama</th>
            </tr>
          </thead>
        </table>
      </>
    )
  }
}
```

```

        <tbody>
        {
            this.state.pesertaLomba.map((val, index)=>{
                return(
                    <tr>
                        <td>{index+1}</td>
                        <td>{val}</td>
                    </tr>
                )
            })
        }
    </tbody>
</table>
</>
)
}
}

```

berikut ini tampilannya:

Daftar Peserta Lomba

No	Nama
1	Budi
2	Susi
3	Lala
4	Agung

ya begitulah kurang lebih cara menampilkan lists di dalam class components

Forms

lalu lists sendiri itu erat kaitannya dengan penambahan data lalu bagaimana penerapannya, pertama mari kita coba buat form terlebih dahulu berdasarkan data diatas, itu berarti kita membutuhkan form dengan input nama

```
<form >
  <label>
    Masukkan nama peserta:
  </label>
  <input type="text" />
  <input type="submit" value="Submit" />
</form>
```

lalu langkah berikutnya adalah kita tambahkan state baru seperti di bawah ini:

```
constructor(props){
  super(props)
  this.state = {
    pesertaLomba : [ 'Budi', 'Susi', 'Lala', 'Agung' ],
    inputName : ""
  }
}
```

lalu tambahkan value input text nya dengan this.state.inputName, tambahkan handle di input text dan tambahkan handle di submit seperti di bawah ini:

```
<form onSubmit={this.handleSubmit}>
  <label>
    Masukkan nama peserta:
  </label>
  <input type="text" value={this.state.inputName}
onChange={this.handleChange}/>
  <input type="submit" value="Submit" />
</form>
```

lalu tambahkan method handle submit dan handlechange di dalam class dan bind method tersebut dalam constructor

```
constructor(props){
  super(props)
  this.state = {
    pesertaLomba : [ 'Budi', 'Susi', 'Lala', 'Agung' ],
    inputName : ""
  }

  this.handleChange = this.handleChange.bind(this);
  this.handleSubmit = this.handleSubmit.bind(this);
}

handleChange(event){
  this.setState({inputName: event.target.value});
}

handleSubmit(event){
  event.preventDefault()
  this.setState({
    pesertaLomba: [...this.state.pesertaLomba, this.state.inputName],
    inputName: ""
  })
}
```

berikut ini contoh penerepan membuat lists dan forms:

```
import React, {Component} from "react"
import "./Lists.css"

class Lists extends Component{

  constructor(props){
    super(props)
    this.state ={
      pesertaLomba : [ 'Budi', 'Susi', 'Lala', 'Agung' ],
      inputName : ""
    }

    this.handleChange = this.handleChange.bind(this);
    this.handleSubmit = this.handleSubmit.bind(this);
  }

  handleChange(event){
    this.setState({inputName: event.target.value});
  }

  handleSubmit(event){
    event.preventDefault()
    console.log(this.state.inputName)
    this.setState({
      pesertaLomba: [...this.state.pesertaLomba, this.state.inputName],
      inputName: ""
    })
  }

  render(){
    return(
      <>
        <h1>Daftar Peserta Lomba</h1>
        <table>
          <thead>
            <tr>
              <th>No</th>
              <th>Nama</th>
              <th></th>
            </tr>
          </thead>
          <tbody>
            {
              this.state.pesertaLomba.map((val, index)=>{
                return(
                  <tr>
                    <td>{index+1}</td>
                    <td>{val}</td>
                    <td></td>
                  </tr>
                )
              })
            }
          </tbody>
        </table>
      </>
    )
  }
}
```

```

    </tbody>
  </table>
  { /* Form */ }
  <h1>Form Peserta</h1>
  <form onSubmit={this.handleSubmit}>
    <label>
      Masukkan nama peserta:
    </label>
    <input type="text" value={this.state.inputName}
onChange={this.handleChange}/>
    <button>submit</button>
  </form>
</>
  )
}
}

export default Lists

```

Tugas

buatlah form untuk menambahkan data ke tabel harga buah dengan inputan nama, hargaTotal dan beratTotal (dalam satuan gram)

untuk inisialisasi data awal gunakan data di bawah ini:

```

var daftarBuah = [
  {nama: "Nanas", hargaTotal: 100000, beratTotal: 4000 },
  {nama: "Manggis", hargaTotal: 350000, beratTotal: 10000},
  {nama: "Nangka", hargaTotal: 90000, beratTotal: 2000},
  {nama: "Durian", hargaTotal: 400000, beratTotal: 5000},
  {nama: "Strawberry", hargaTotal: 120000, beratTotal: 6000}
]

```

isi variabel diatas bisa langsung di masukkan ke state awal

lalu pada tabel tersebut tambahkan kolom aksi yang berisi button **edit** (yang dimana ketika klik edit maka akan muncul datanya di inputan form untuk bisa diedit) dan **delete**

lalu tampilkan dalam App.js untuk tugas-tugas sebelumnya yang di tampilkan silahkan komen saja, jadi yang di tampilkan hanya tugas 12 saja

kurang lebih outputnya seperti ini (stylingnya sudah boleh bebas asalkan **tidak** menggunakan **bootstrap** dan sejenisnya):

Daftar Harga Buah

No	Nama	Harga total	Berat total	Harga per kg	Aksi
1	Nanas	100000	4 kg	25000	<button>Edit</button> <button>Delete</button>
2	Manggis	350000	10 kg	35000	<button>Edit</button> <button>Delete</button>
3	Nangka	90000	2 kg	45000	<button>Edit</button> <button>Delete</button>
4	Durian	400000	5 kg	80000	<button>Edit</button> <button>Delete</button>
5	Strawberry	120000	6 kg	20000	<button>Edit</button> <button>Delete</button>

Form Daftar Harga Buah

Nama:	<input type="text"/>
Harga Total:	<input type="text"/>
Berat Total(dalam gram):	<input type="text" value="0"/>
<button>submit</button>	

ketentuan pada form adalah:

- semua inputan wajib diisi
- inputan total berat minimal 2 kg

pada tugas ini **tidak di perbolehkan** menggunakan **bootstrap** ataupun sejenisnya