

TUGAS PERORANGAN/INDIVIDU
REDUX DAN HOC

Disusun sebagai
MATA KULIAH : Pemrograman Berbasis Framework

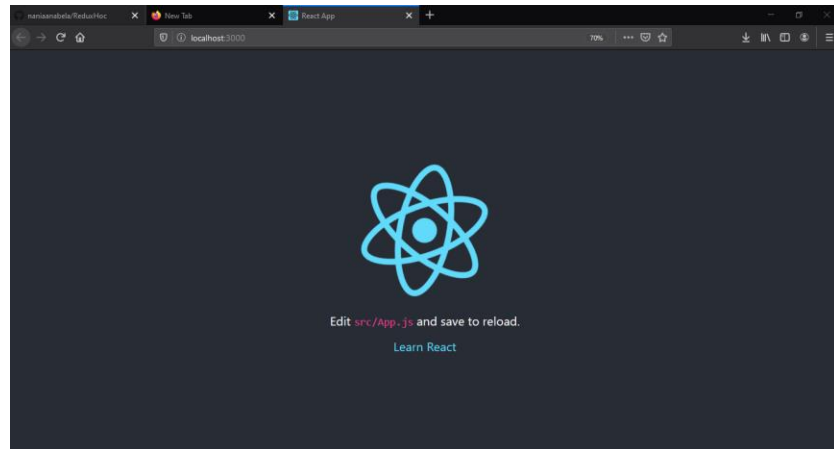
Oleh :
Nurcahya Nania Anabela
TI-3A / 24
1741720156



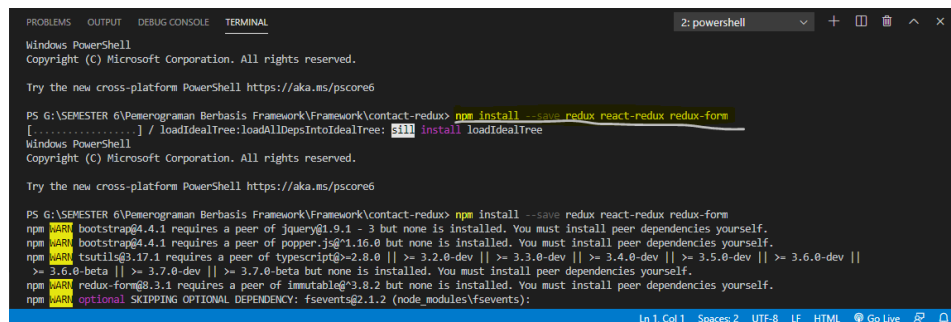
PROGRAM STUDI D-IV T.INFORMATIKA
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG
2020

LATIHAN PRAKTIKUM

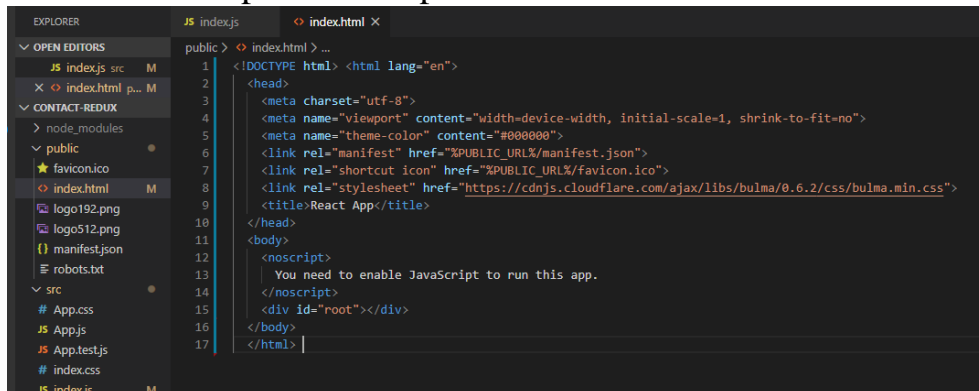
1. Buat dan Install Redux



2. Install Form



3. Ubah file index.html pada folder public/index.html



4. Ubah file App.js pada folder src/App.js

```
JS index.js    index.html    JS App.js    X
src > JS App.js > [0] SignInForm
1 import React, { Component } from 'react';
2 import { reduxForm, Field } from 'redux-form';
3 import logo from './logo.svg'; import './App.css';
4
5 let SignInForm = props => {
6   const { handleSubmit } = props;
7   return <form onSubmit={handleSubmit} className="form">
8     <div className="field">
9       <div className="control">
10        <Field name="firstName" component={renderField} type="text" label="First Name"/>
11      </div>
12    </div>
13
14    <div className="field">
15      <div className="control">
16        <Field name="lastName" component={renderField} type="text" label="Last Name"/>
17      </div>
18    </div>
19
20    <div className="field">
21      <div className="control">
22        <Field name="email" component={renderField} type="email" label="Email Address"/>
23      </div>
```

```
JS index.js    index.html    JS App.js    X
src > JS App.js > [0] SignInForm
24 </div>
25
26 <div className="field">
27   <div className="control">
28     <Field name="age" component={renderField} type="number" label="Age"/>
29   </div>
30 </div>
31
32 <div className="field">
33   <div className="control">
34     <label className="label">Proficiency</label>
35     <div className="select">
36       <Field className="input" name="proficiency" component="select">
37         <option />
38         <option value="beginner">Beginner Dev</option>
39         <option value="intermediate">Intermediate Dev</option>
40         <option value="expert">Expert Dev</option>
41       </Field>
42     </div>
43   </div>
44 </div>
45
```

```
JS index.js    index.html    JS App.js    X
src > JS App.js > [0] SignInForm
46 <div className="field">
47   <div className="control">
48     <label className="label">Gender</label>
49     <label className="radio">
50       <Field name="gender" component="input" type="radio" value="male" />
51       { ' ' }
52       Male
53     </label>
54     <label className="radio">
55       <Field name="gender" component="input" type="radio" value="female" />
56       { ' ' }
57       Female
58     </label>
59   </div>
60 </div>
61
62 <div className="field">
63   <div className="control">
64     <label className="checkbox">
65       <Field name="saveDetails" id="saveDetails" component="input" type="checkbox"/>
66       Save Details
67     </label>
68   </div>
```

```
JS indexjs  index.html JS Appjs X
src > JS Appjs > [0] SignInForm
69     </div>
70
71     <div className="field">
72       <div className="control">
73         <label className="label">Message</label>
74         <Field className="textarea" name="message" component="textarea" />
75       </div>
76     </div>
77
78     <div className="field">
79       <div className="control">
80         <button className="button is-link">Submit</button>
81       </div>
82     </div>
83
84   </form>;
85 };
86
87 const validate = val => {
88   const errors = {};
89   if (!val.firstName) {
90     console.log('First Name is required');
```

```
JS indexjs X  index.html JS Appjs X
src > JS Appjs > [0] SignInForm
91   errors.firstName = 'Required';  }
92   if (!val.lastName) {
93     console.log('Last Name is required');
94     errors.lastName = 'Required';
95   }
96   if (!val.email) {
97     console.log('email is required');
98     errors.email = 'Required';
99   } else if (!/^.+@.+$/.test(val.email)) {
100     console.log('email is invalid');
101     errors.email = 'Invalid email address';
102   }
103   if (!val.age) {
104     errors.age = 'Required'
105   } else if (isNaN(Number(val.age)))
106   {
107     errors.age = 'Must be a number'
108   } else if (Number(val.age) < 18) {
109     errors.age = 'Sorry, you must be at least 18 years old'
110   }   return errors;
111 };
112
```

```
JS indexjs  index.html JS Appjs X
src > JS Appjs > [0] SignInForm
113 const renderField = ({ input, label, type, meta: { touched, error, warning } }) => (
114   <div>
115     <div className="control">
116       <label className="field">{label}</label>
117       <input className="input" {...input} placeholder={label} type={type}/>
118       {touched && ((error && <span>{error}</span>) || (warning && <span>{warning}</span>))}
119     </div>
120   </div>
121 )
122 SignInForm = reduxForm({
123   form: 'signIn',
124   validate, })(SignInForm);
125
126 class App extends Component {
127
128   handleSignIn = values => {
129     console.log(values);  };
130
131   render() {
132     return (
133       <div className="App">
134         <header className="App-header">
135           <img src={logo} className="App-logo" alt="logo" />
```

```
JS index.js  X index.html  JS App.js  X
src > JS App.js > | SignInForm
136 |         <h1 className="App-title">Welcome to React x redux-form</h1>
137 |       </header>
138 |       <div className="container">
139 |         <p className="App-intro">
140 |           Contact Form
141 |         </p>
142 |         <SignInForm onSubmit={this.handleSignIn} />
143 |       </div>
144 |     </div>
145 |   );
146 | }
147 | }
148 |
149 | export default App;
```

5. Ubah file src/index.js

```
JS index.js  X index.html  JS App.js
src > JS index.js > ...
1 | import React from 'react';
2 | import ReactDOM from 'react-dom';
3 |
4 | import { createStore, combineReducers } from 'redux';
5 | import { Provider } from 'react-redux';
6 | import { reducer as formReducer } from 'redux-form';
7 |
8 | import './index.css';
9 | import App from './App';
10 | import registerServiceWorker from './registerServiceWorker';
11 |
12 | const rootReducer = combineReducers({
13 |   form: formReducer,
14 | });
15 |
16 | const store = createStore(rootReducer);
17 |
18 | ReactDOM.render(
19 |   <Provider store={store}>
20 |     <App />
21 |   </Provider>,
22 |   document.getElementById('root') );
23 |
24 | registerServiceWorker();
```

6. Output hasil setelah program dijalankan.

