PRAKTIKUM MINGGU 7

Mata kuliah : PBF(Pemrograman Berbasis Framework)
Alif Mohammad Sultan S (08)



D-IV Teknik Informatika Teknologi Informasi Politeknik Negeri Malang

```
import React, {Component} from 'react'
import { connect } from 'react-redux'
import { addTodo } from '../Action/actionCreator'
import {bindActionCreators} from 'redux'
class CreateTodo extends Component {
constructor(props){
super(props)
this.state = {
todotext: '',
this.onChangeTodoText = this.onChangeTodoText.bind(this)
 onChangeTodoText(e){
this.setState({
 todotext: e.target.value
 })
render(){
   return (
   <div className="form-group row">
   <div className="col-sm-10">
    <input onChange={this.onChangeTodoText} value={this.state.todotext}</pre>
    type="text" className="form-
control" id="inputEmail3" placeholder="add todo here"/>
    <button type="button" onClick={ () => this.setState({ todotext: '' }) }
    style={{marginTop: "25px", marginRight: "15px"}} className="btn btndanger">
Cancel</button>
    <button type="button" onClick={() =>{
    this.props.addTodo(this.state.todotext); this.setState({ todotext: '' }) }
    style={{marginTop: "25px"}} className="btn btn-success">Add Todo</button>
    </div>
    </div>
    );
const mapDispatchToProps = (dispatch) => {
   return bindActionCreators({
    addTodo
    }, dispatch)
   export default connect(null, mapDispatchToProps)(CreateTodo)
```

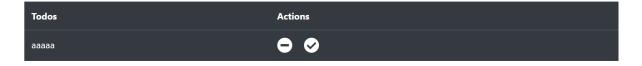
add todo here

Cancel

Add Todo

```
import { ADD_TODO, REMOVE_TODO, TOGGLE_TODO, SET_VISIBILITY_FILTER } from
'./actionTypes'
let TodoId = 2
export const addTodo = text => ({
 type: ADD_TODO,
 id: TodoId++,
 text
export const deleteTodo = (id) => ({
 type: REMOVE_TODO,
 id: id
})
export const toggleTodo = (id) => ({
type: TOGGLE_TODO,
id: id
})
export const setVisibilityFilter = filter => ({
 type: SET_VISIBILITY_FILTER,
 filter
```

Untuk Mengimport todo yang telah dibuat dari CreateTodo.js kemudian akan dipanggil berdasarkan id dari delete hingga add



```
import React, { Component } from "react";
import { connect } from "react-redux";
import {
  deleteTodo,
  toggleTodo,
  setVisibilityFilter
} from "../Action/actionCreator";
```

```
import { SHOW_ALL, SHOW_COMPLETED, SHOW_ACTIVE } from "../Action/actionTypes";
import { bindActionCreators } from "redux";
class Table extends Component {
render() {
return (
<div className="col-lg-10 offset-lg-1 col-md-10 col-sm-12 col-xs-12">
<nav style={{ marginTop: "60px" }}>
 <li
className={"breadcrumb-item "+ (this.props.visibilityFilter === SHOW_ALL ?
onClick={() => this.props.setVisibilityFilter(SHOW ALL)}
A11
<1i
className={"breadcrumb-item "+ (this.props.visibilityFilter ===
SHOW COMPLETED ? 'active' : '') }
onClick={() => this.props.setVisibilityFilter(SHOW COMPLETED)}
Completed
<li
 className={"breadcrumb-item "+ (this.props.visibilityFilter === SHOW ACTIVE ?
active' : '') }
 onClick={() => this.props.setVisibilityFilter(SHOW ACTIVE)}
Active
</nav>
{this.props.todos.length !== 0 ? (
<table
 style={{ marginTop: "60px" }}
 className="table table-hover table-dark"
<thead>
Todos
Actions
</thead>
{this.props.todos.map(todo => (
```

```
<td
 style={{
textDecoration: todo.completed ? "line-through" : "none"
 }}
 {todo.text} {todo.completed === true ? "(completed)" : ""}
 className="fas fa-minus-circle"
 onClick={() => this.props.deleteTodo(todo.id)}
 style={{
color: "white",
fontSize: "20pt",
 marginRight: "20px"
 }}
<span
className="fas fa-check-circle"
 onClick={() => this.props.toggleTodo(todo.id)}
 style={{ color: "white", fontSize: "20pt" }}
))}
<div
 style={{ marginTop: "50px" }}
 className="col-lg-10 col-md-10 col-xs-12 col-sm-12 offset-lg-1"
<div className="alert alert-danger" role="alert">
 Todo List is empty or Filter results show no results
</div>
 )}{" "}
 );
const getVisibleTodos = (todos, filter) => {
   switch (filter) {
   case SHOW_ALL:
   return todos;
```

```
case SHOW COMPLETED:
    return todos.filter(t => t.completed);
    case SHOW_ACTIVE:
    return todos.filter(t => !t.completed);
    default:
    throw new Error("Unknown filter: " + filter);
};
const mapStateToProps = state => {
    return { todos: getVisibleTodos(state.todos, state.visibilityFilter),
    visibilityFilter: state.visibilityFilter
   };
};
const mapDispatchToProps = dispatch => {
    return bindActionCreators(
    deleteTodo,
    toggleTodo,
    setVisibilityFilter
    dispatch
    );
export default connect(mapStateToProps, mapDispatchToProps)(Table);
```

Untuk menampilan table yang telah dibuat dengan memanggil bootstrap maka table akan berjalan

