

Nama : Annisa Jufe Aryani

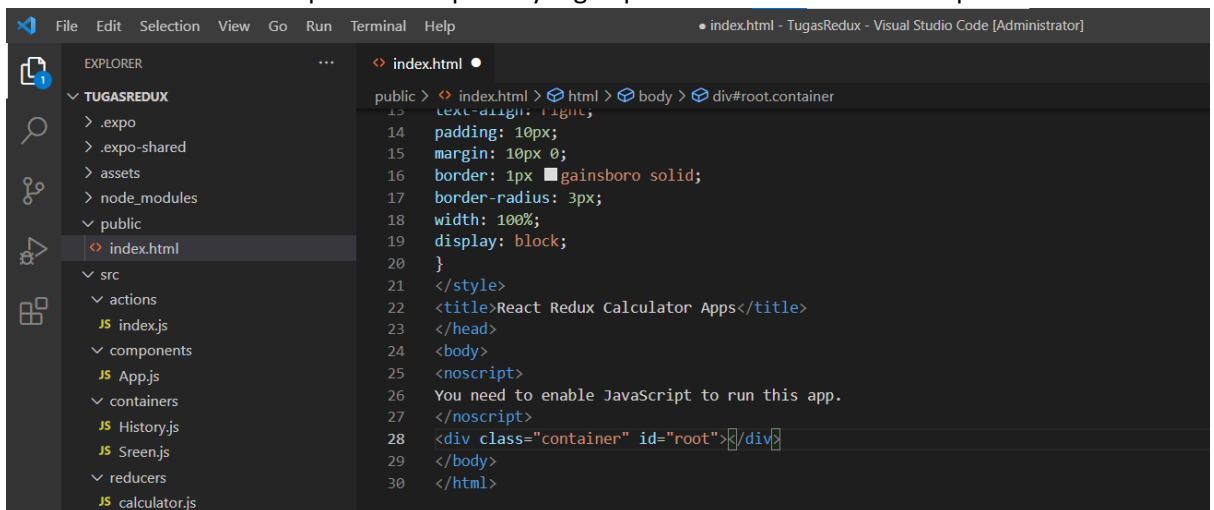
NIM : 118140124

Kelas : PAM RB

Tugas Redux

Membuat Program Aplikasi Calculator Sederhana dengan Menggunakan Redux

1. Membuat file index.html pada folder public yang dapat menambahkan bootstrap



```
File Edit Selection View Go Run Terminal Help
• index.html - TugasRedux - Visual Studio Code [Administrator]

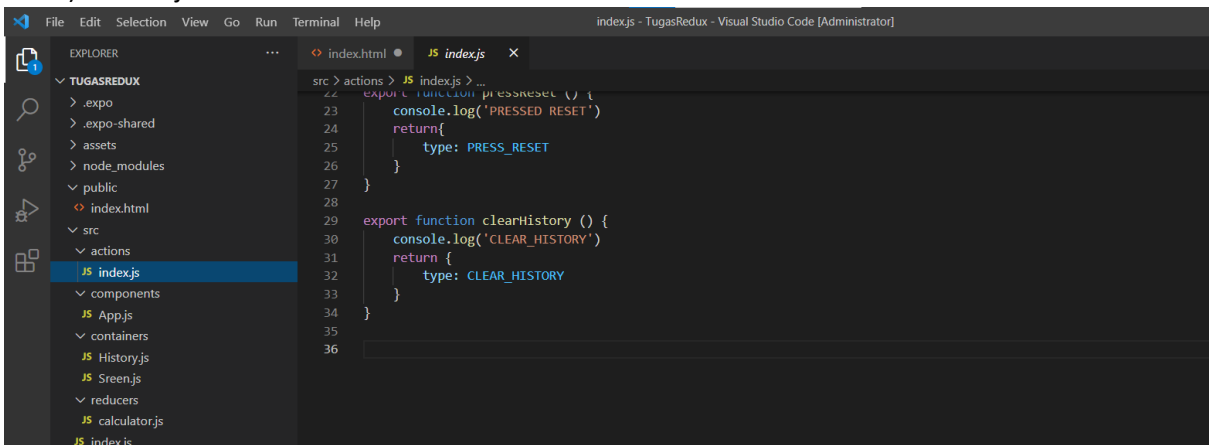
EXPLORER
TUGASREDUX
  > .expo
  > .expo-shared
  > assets
  > node_modules
  > public
    index.html
  > src
    actions
    index.js
    components
    App.js
    containers
    History.js
    Screen.js
    reducers
    calculator.js

index.html
13 text-align: right;
14 padding: 10px;
15 margin: 10px 0;
16 border: 1px solid gainsboro;
17 border-radius: 3px;
18 width: 100%;
19 display: block;
20 }
21 </style>
22 <title>React Redux Calculator Apps</title>
23 </head>
24 <body>
25 <noscript>
26 You need to enable JavaScript to run this app.
27 </noscript>
28 <div class="container" id="root"></div>
29 </body>
30 </html>
```

2. Membuat beberapa folder pada folder src

a. Folder actions

a) Index.js



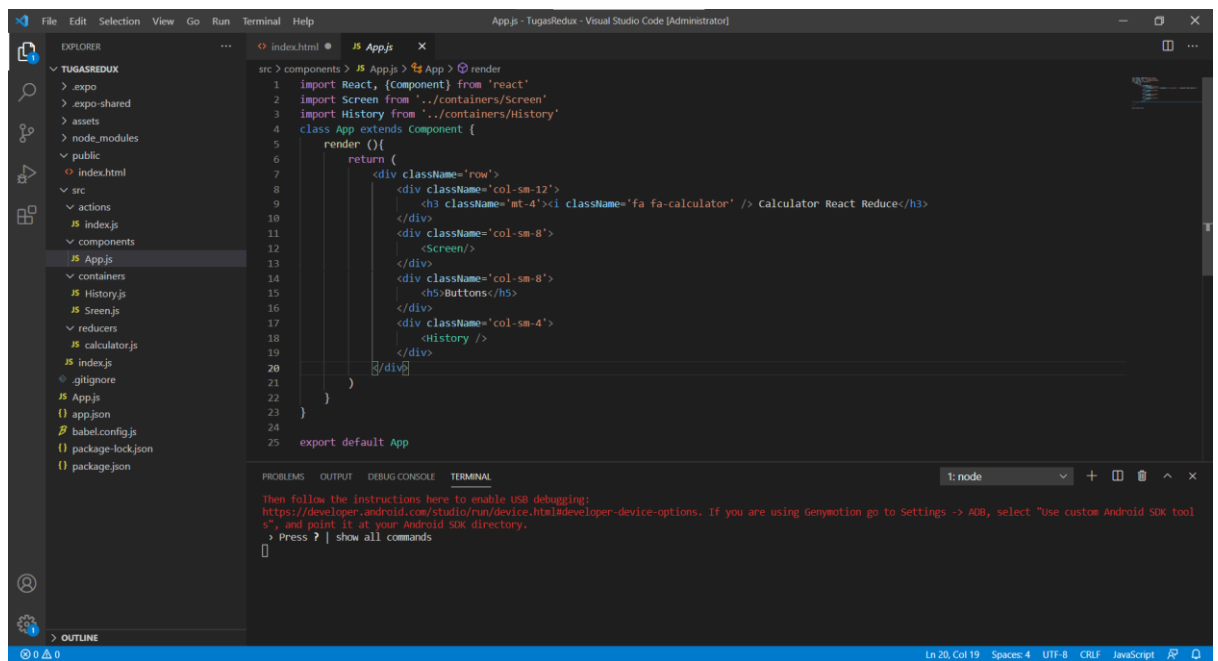
```
File Edit Selection View Go Run Terminal Help
index.js - TugasRedux - Visual Studio Code [Administrator]

EXPLORER
TUGASREDUX
  > .expo
  > .expo-shared
  > assets
  > node_modules
  > public
    index.html
  > src
    actions
      index.js
    components
    App.js
    containers
    History.js
    Screen.js
    reducers
    calculator.js
    index.js

src > actions > JS index.js
23 export function pressReset () {
24   console.log('PRESSED RESET')
25   return {
26     type: PRESS_RESET
27   }
28 }
29 export function clearHistory () {
30   console.log('CLEAR_HISTORY')
31   return {
32     type: CLEAR_HISTORY
33   }
34 }
35
36
```

b. Components

a) App.js



```
src > components > JS App.js > App > render
1 import React, {Component} from 'react'
2 import Screen from '../containers/Screen'
3 import History from '../containers/History'
4 class App extends Component {
5   render () {
6     return (
7       <div className='row'>
8         <div className='col-sm-12'>
9           <h3 className='mt-4'><i className='fa fa-calculator' /> Calculator React Reduce</h3>
10         </div>
11         <div className='col-sm-8'>
12           <Screen/>
13         </div>
14         <div className='col-sm-8'>
15           <h5>Buttons</h5>
16         </div>
17         <div className='col-sm-4'>
18           <History />
19         </div>
20       </div>
21     )
22   }
23 }
24
25 export default App
```

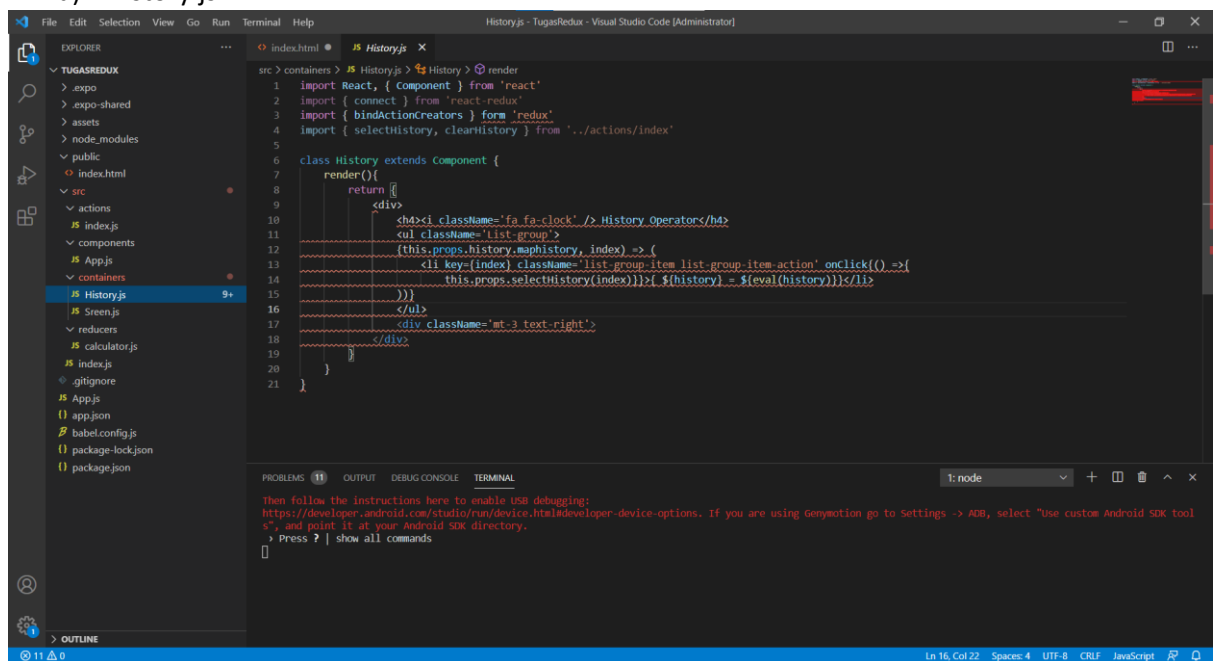
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

1: node

Then follow the instructions here to enable USB debugging:
<https://developer.android.com/studio/run/device.html#developer-device-options>. If you are using Genymotion go to Settings -> ADB, select "Use custom Android SDK tool s", and point it at your Android SDK directory.
> Press ? | show all commands

c. Containers

a) History.js



```
src > containers > JS History.js > History > render
1 import React, {Component} from 'react'
2 import { connect } from 'react-redux'
3 import { bindActionCreators } from 'redux'
4 import { selectHistory, clearHistory } from '../actions/index'
5
6 class History extends Component {
7   render() {
8     return (
9       <div>
10         <h4><i className='fa fa-clock' /> History Operators</h4>
11         <ul className='list-group'>
12           {this.props.history.map(history, index) => (
13             <li key={index} className='list-group-item list-group-item-action' onClick={() => {
14               this.props.selectHistory(index)}}>{ $(history) = $(eval(history))}</li>
15             )}
16         </ul>
17         <div className='mt-3 text-right'>
18           <button className='btn btn-danger'>Clear History</button>
19         </div>
20       </div>
21     )
22   }
23 }
```

PROBLEMS 11 OUTPUT DEBUG CONSOLE TERMINAL

1: node

Then follow the instructions here to enable USB debugging:
<https://developer.android.com/studio/run/device.html#developer-device-options>. If you are using Genymotion go to Settings -> ADB, select "Use custom Android SDK tool s", and point it at your Android SDK directory.
> Press ? | show all commands

b) Screen.js

```
File Edit Selection View Go Run Terminal Help
Screen.js - TugasRedux - Visual Studio Code [Administrator]

EXPLORER
TUGASREDUX
  > expo
  > expo-shared
  > assets
  > node_modules
  > public
  > index.html
  > src
  > actions
  > index.js
  > components
  > App.js
  > containers
  > History.js
  > Screen.js
  > reducers
  > calculator.js
  > index.js
  > .gitignore
  > App.js
  > app.json
  > babel.config.js
  > package-lock.json
  > package.json

src > containers > Screen.js > mapStateToProps
1 import React, { Component } from 'react'
2 import { connect } from 'react-redux'
3
4 class Screen extends Component{
5   render () {
6     return (
7       <div className='row'>
8         <div className='screen'>
9           {(this.props.screen)}
10         </div>
11       </div>
12     )
13   }
14 }
15
16 function mapStateToProps (state) {
17   return{
18     screen: state.calculator.screen
19   }
20 }
21
22 export default connect(mapStateToProps)(Screen)

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
1: node
Then follow the instructions here to enable USB debugging:
https://developer.android.com/studio/run/device.html#developer-device-options. If you are using Genymotion go to Settings -> ADB, select "Use custom Android SDK tool
s", and point it at your Android SDK directory.
> Press ? | show all commands
[]

Ln 19, Col 6 Spaces: 4 UTF-8 CRLF JavaScript
```

d. Reducers

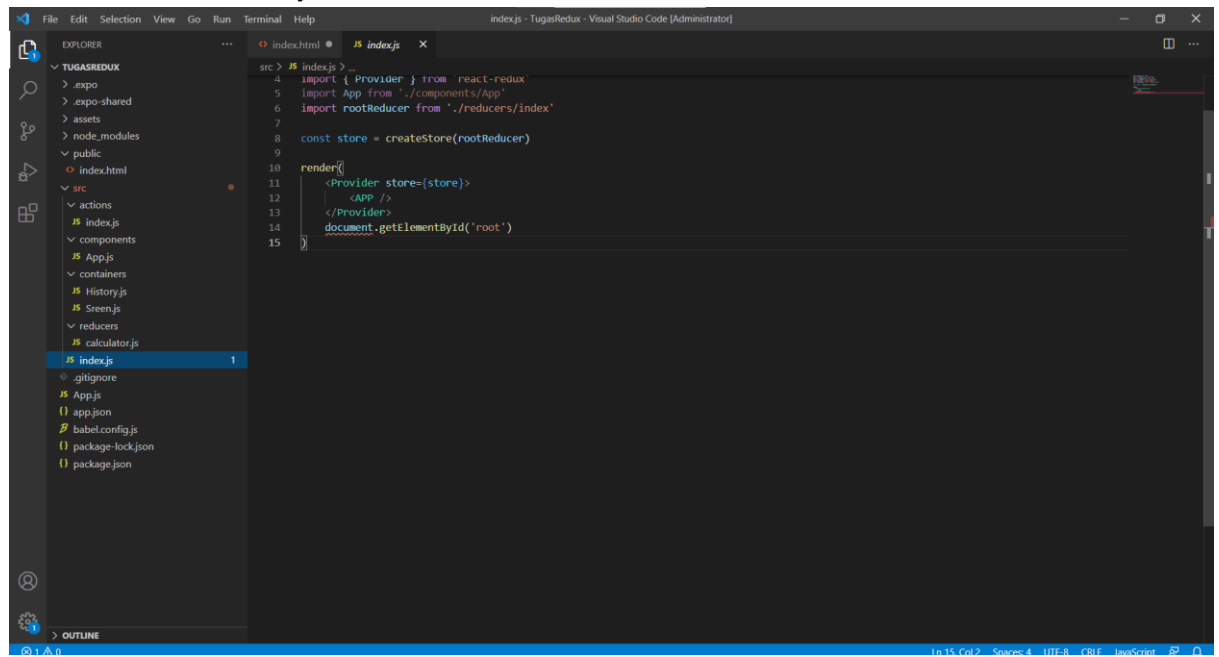
a) Calculator.js

```
File Edit Selection View Go Run Terminal Help
calculator.js - TugaxRedux - Visual Studio Code [Administrator]

EXPLORER
TUGASREDUX
  > expo
  > expo-shared
  > assets
  > node_modules
  > public
  > index.html
  > src
  > actions
  > index.js
  > components
  > App.js
  > containers
  > History.js
  > Screen.js
  > reducers
  > calculator.js
  > index.js
  > .gitignore
  > App.js
  > app.json
  > babel.config.js
  > package-lock.json
  > package.json

src > reducers > calculator.js > default
1 import { PRESS_BUTTON, PRESS_EQUALS, PRESS_RESET, SELECT_HISTORY, CLEAR_HISTORY } from '../actions/index'
2 const INITIAL_STATE = { screen: '', history: [] }
3 const calculator = (state = INITIAL_STATE, action) => {
4   switch (action.type) {
5     case PRESS_BUTTON:
6       return {
7         ...state,
8         screen: (state.screen + ' + ' + action.value)
9       }
10     case PRESS_EQUALS:
11       const newState = { ...state }
12       newState.screen = (eval(state.screen))
13       newState.history = state.history.concat(`${state.screen}`)
14       return newState
15     case PRESS_RESET:
16       return {
17         ...state,
18         screen: ''
19       }
20     case SELECT_HISTORY:
21       return {
22         ...state,
23         screen: state.history[action.id]
24       }
25     case CLEAR_HISTORY:
26       return {
27         ...state,
28         history: []
29       }
30     default:
31       return state
32   }
33 }
34 export default calculator
```

e. Membuat file index.js



The screenshot shows the Visual Studio Code editor with a project named 'TugasRedux'. The Explorer sidebar on the left shows the file structure, with 'index.js' selected under the 'src' directory. The main editor area displays the content of 'index.js'.

```
src > # index.js > ...
4 import { Provider } from 'react-redux'
5 import App from './components/App'
6 import rootReducer from './reducers/index'
7
8 const store = createStore(rootReducer)
9
10 render()
11   <Provider store={store}>
12     <App />
13   </Provider>
14   document.getElementById('root')
15 }
```