

TUGAS PERTEMUAN 3

Disusun Guna Memenuhi Tugas Mata Kuliah Pemrograman Berbasis Framework

Dosen Pengampu : Dimas Wahyu Wibowo, S.T., M.T.



Disusun Oleh :

Akhmad Ramadani (2041720002)

TI-3B

**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG**

2023

1. Latihan

a. Import react menggunakan url

i. Kode Program

```
react0.html

<!-- *** Author : Akhmad Ramadani *** -->
<!-- *** NIM      : 2041720002 *** -->
<!-- *** Kelas    : TI-3B   *** -->

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible"
content="IE=edge">
        <meta name="viewport" content="width=device-width,
initial-scale=1.0">
        <title>React Uncover</title>
</head>

<body>
    <h1>Belajar React</h1>
    <script
src="https://unpkg.com/react@18/umd/react.development.js
" crossorigin></script>
    <script
src="https://unpkg.com/react-dom@18/umd/react-dom.develo
pment.js" crossorigin></script>
    <script>
        const myElement = React.createElement("h1", {
            id: "judul"
        }, "Hello React");

        console.log(myElement);
    </script>
</body>

</html>
```

ii. Hasil

Belajar React



A screenshot of the Chrome DevTools Console tab. The tab bar at the top includes Elements, Console (which is selected), Sources, Network, Performance, Memory, Application, Security, and Lighthouse. Below the tab bar are icons for back, forward, and search, followed by 'top ▾' and 'Filter'. The main console area displays the following text:

```
Download the React DevTools for a better development experience: https://reactjs.org/link/react-devtools
> {$$typeof: Symbol/react.element, type: 'h1', key: null, ref: null, props: {...}, ...}
```

iii. Penjelasan

1. Kode di atas adalah untuk mengimport React menggunakan link dari react.
2. Kemudian, untuk script digunakan untuk mengecek apakah react sudah terimport

b. Membuat Template HTML - File React di Local

i. Kode Program

```
react.html

<!-- *** Author : Akhmad Ramadani *** -->
<!-- *** NIM      : 2041720002 *** -->
<!-- *** Kelas    : TI-3B   *** -->

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible"
content="IE=edge">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>React Uncover</title>
</head>

<body>
    <h1>Belajar React</h1>
    <script src="js/react.development.js"
crossorigin></script>
    <script src="js/react-dom.development.js"
crossorigin></script>
    <script>
        const myElement = React.createElement("h1", {
            id: "judul"
        }, "Hello React");

        console.log(myElement);
    </script>
</body>

</html>
```

ii. Hasil

Belajar React



iii. Penjelasan

1. Kode di atas adalah untuk mengimport React menggunakan file local yang telah di-download dari link pada nomor 1.
2. Kemudian, untuk script digunakan untuk mengecek apakah react sudah terimport

c. Method React.createElement()

i. Kode Program

```
react.html

<!-- *** Author : Akhmad Ramadani *** -->
<!-- *** NIM      : 2041720002 *** -->
<!-- *** Kelas   : TI-3B   *** -->

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible"
content="IE=edge">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>React Uncover</title>
</head>

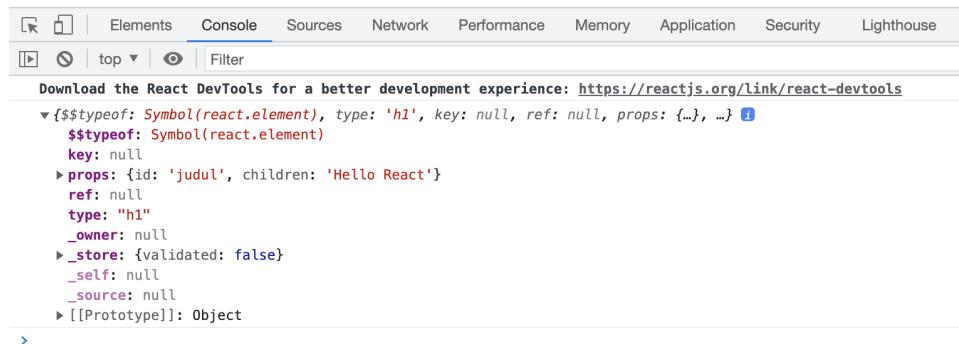
<body>
    <h1>Belajar React</h1>
    <script src="js/react.development.js"
crossorigin></script>
    <script src="js/react-dom.development.js"
crossorigin></script>
    <script>
        const myElement = React.createElement("h1", {
            id: "judul"
        }, "Hello React");

        console.log(myElement);
    </script>
</body>

</html>
```

ii. Hasil

Belajar React



iii. Penjelasan

1. Kode di atas menggunakan API React.createElement() untuk membuat elemen React. Fungsi React.createElement()

digunakan untuk membuat elemen React virtual. Elemen virtual ini kemudian akan diubah menjadi elemen DOM yang sesungguhnya dan ditampilkan pada halaman web.

2. Dalam kasus ini, elemen yang dibuat adalah tag h1 dengan atribut id bernilai "judul" dan teks "Hello React". Hasil dari pembuatan elemen ini kemudian dicetak ke konsol browser menggunakan fungsi console.log().

d. Method ReactDOM.createRoot().render()

i. Kode Program

```
react2.html

<!-- *** Author : Akhmad Ramadani *** -->
<!-- *** NIM      : 2041720002 *** -->
<!-- *** Kelas    : TI-3B   *** -->

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible"
content="IE=edge">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>React Uncover</title>
    <style>
        #judul {
            color: blueviolet;
            text-shadow: 1px 1px 2px;
            font-family: Verdana, Geneva, Tahoma,
sans-serif;
        }
    </style>
</head>

<body>
    <h1>Belajar React</h1>

    <div id="root"></div>

    <script src="js/react.development.js"
crossorigin></script>
    <script src="js/react-dom.development.js"
crossorigin></script>
    <script>
        const myElement = React.createElement("h1", {
            id: "judul"
        }, "Hello React");

        ReactDOM.createRoot(document.getElementById("root")).ren
der(myElement);
    </script>
</body>

</html>
```

ii. Hasil

Belajar React

Hello React

iii. Penjelasan

1. Kode tersebut adalah sebuah halaman HTML yang menggunakan React untuk menampilkan elemen <h1> yang berisi teks "Belajar React". Di dalam halaman tersebut terdapat sebuah <div> dengan id "root", yang nantinya akan digunakan untuk me-render elemen React.
2. Di dalam tag <script>, terdapat kode JavaScript yang menggunakan metode React.createElement() untuk membuat sebuah elemen <h1> dengan properti id bernilai "judul" dan teks "Hello React". Kemudian, elemen tersebut akan di-render menggunakan ReactDOM.createRoot() dan dilekatkan ke dalam <div> dengan id "root".

e. Method ReactDOM.createRoot().render()

i. Kode Program

```
react3.html

<!-- *** Author : Akhmad Ramadani *** -->
<!-- *** NIM      : 2041720002 *** -->
<!-- *** Kelas    : TI-3B   *** -->

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible"
content="IE=edge">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>React Uncover</title>
    <style>
        #judul {
            color: blueviolet;
            text-shadow: 1px 1px 2px;
            font-family: Verdana, Geneva, Tahoma,
sans-serif;
        }
    </style>
</head>

<body>
    <h1>Belajar React</h1>

    <div id="root"></div>

    <script src="js/react.development.js"
crossorigin></script>
    <script src="js/react-dom.development.js"
crossorigin></script>
    <script>
        const myElement = React.createElement("h1", {
            id: "judul"
        }, "Hello React");

        const root =
ReactDOM.createRoot(document.getElementById("root"));

        root.render(myElement);
    </script>
</body>

</html>
```

ii. Hasil

Belajar React

Hello React

iii. Penjelasan

1. Pada tag <script>, terdapat dua file JavaScript yang diambil dari React dan ReactDOM, serta kode JavaScript yang membuat sebuah elemen React dengan fungsi createElement() dan merender elemen tersebut ke dalam div dengan ID "root" menggunakan fungsi createRoot() dan render() dari ReactDOM.
2. Kemudian, akan tampil Hello React dengan dengan style dari id judul seperti di atas.

f. Method ReactDOM.render()

i. Kode Program

```
react4.html

<!-- *** Author : Akhmad Ramadani *** -->
<!-- *** NIM      : 2041720002 *** -->
<!-- *** Kelas    : TI-3B   *** -->

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible"
content="IE=edge">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>React Uncover</title>
    <style>
        #judul {
            color: blueviolet;
            text-shadow: 1px 1px 2px;
            font-family: Verdana, Geneva, Tahoma,
sans-serif;
        }
    </style>
</head>

<body>
    <h1>Belajar React</h1>

    <div id="root"></div>

    <script src="js/react.development.js"
crossorigin></script>
    <script src="js/react-dom.development.js"
crossorigin></script>
    <script>
        const myElement = React.createElement("h1", {
            id: "judul"
        }, "Hello React");

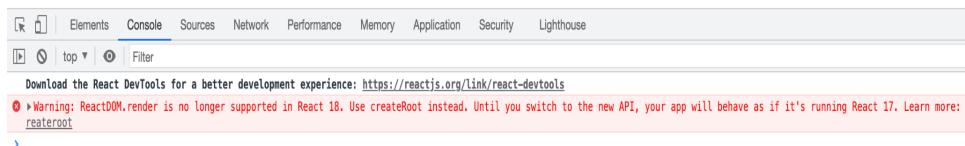
        ReactDOM.render(myElement,
document.getElementById("root"));
    </script>
</body>

</html>
```

ii. Hasil

Belajar React

Hello React



iii. Penjelasan

1. Sama seperti nomor atas, Hello React tetap tampil. Namun, kali ini ada peringatan dikarenakan kita menggunakan method yang ada pada React 17 dan sudah deprecated pada React 18. Sehingga muncullah peringatan seperti di atas.

g. Multiple ReactDOM.createRoot().render()

i. Kode Program

```
react5.html

<!-- *** Author : Akhmad Ramadani *** -->
<!-- *** NIM      : 2041720002 *** -->
<!-- *** Kelas    : TI-3B   *** -->

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible"
content="IE=edge">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>React Uncover</title>
</head>

<body>
    <h1>Belajar React</h1>

    <div id="root"></div>
    <div id="foo"></div>

    <script src="js/react.development.js"
crossorigin></script>
    <script src="js/react-dom.development.js"
crossorigin></script>
    <script>
        const myElement = React.createElement("h1", {
            id: "judul"
        }, "Hello React");

        ReactDOM.createRoot(document.getElementById("root")).ren
der(myElement);

        const myPic = React.createElement("img", {
            src: "img/capybara.jpeg"
        }, null);

        ReactDOM.createRoot(document.getElementById("foo")).rend
er(myPic);
    </script>
</body>

</html>
```

ii. Hasil

Belajar React

Hello React



iii. Penjelasan

1. Pada bagian body, terdapat dua div yaitu root dan foo yang akan digunakan untuk me-render elemen React.
2. Kemudian, pada bagian script, pertama-tama digunakan `React.createElement()` untuk membuat elemen `h1` dengan atribut `id` bernilai "judul" dan teks "Hello React". Setelah itu, elemen tersebut di-render menggunakan `ReactDOM.createRoot()` dengan id bernilai "root".
3. Selanjutnya, digunakan `React.createElement()` untuk membuat elemen `img` dengan atribut `src` yang menunjukkan alamat gambar "capybara.jpeg". Setelah itu, elemen tersebut di-render menggunakan `ReactDOM.createRoot()` dengan id bernilai "foo".
4. Dengan demikian, ketika halaman HTML di-load, maka akan muncul teks "Hello React" dengan font berwarna biru violet pada div dengan id "root", serta gambar capybara pada div dengan id "foo".

h. Nested React Element

i. Kode Program

```
react6.html

<!-- *** Author : Akhmad Ramadani *** -->
<!-- *** NIM      : 2041720002 *** -->
<!-- *** Kelas    : TI-3B   *** -->

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible"
content="IE=edge">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>React Uncover</title>
</head>

<body>
    <h1>Belajar React</h1>

    <div id="root"></div>

    <script src="js/react.development.js"
crossorigin></script>
    <script src="js/react-dom.development.js"
crossorigin></script>
    <script>
        const myMenu = React.createElement("ul",
            {},
            [
                React.createElement("li", {}, "Espresso"),
                React.createElement("li", {}, "Cappuccino"),
                React.createElement("li", {}, "Moccacino")
            ]
);

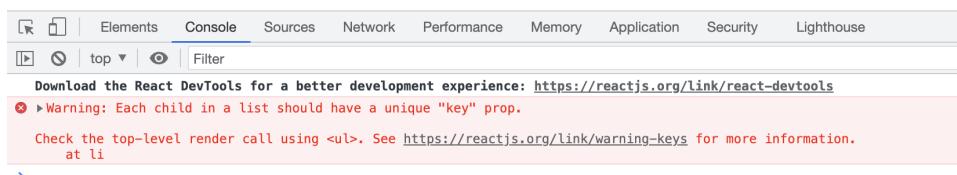
ReactDOM.createRoot(document.getElementById("root")).ren
der(myMenu);
    </script>
</body>

</html>
```

ii. Hasil

Belajar React

- Espresso
- Cappuccino
- Moccacino



iii. Penjelasan

1. Membuat sebuah elemen ul dengan tiga elemen li di dalamnya, masing-masing berisi nama kopi (Espresso, Cappucino, dan Moccacino). Elemen ul tersebut disimpan dalam variabel myMenu menggunakan metode React.createElement().
2. Kemudian, dengan menggunakan ReactDOM.createRoot(), elemen myMenu ditampilkan di dalam div dengan id "root". Saat dijalankan, hasilnya akan menampilkan daftar menu kopi seperti di atas
3. Akan terjadi peringatan seperti pada hasil di atas, karena setiap elemen pada li diharuskan untuk memiliki key yang unik.

i. Nested React Element

i. Kode Program

```
react7.html

<!-- *** Author : Akhmad Ramadani *** -->
<!-- *** NIM      : 2041720002 *** -->
<!-- *** Kelas    : TI-3B   *** -->

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible"
content="IE=edge">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>React Uncover</title>
</head>

<body>
    <h1>Belajar React</h1>

    <div id="root"></div>

    <script src="js/react.development.js"
crossorigin></script>
    <script src="js/react-dom.development.js"
crossorigin></script>
    <script>
        const myMenu = React.createElement("ul",
            {},
            [
                React.createElement("li", {key: 'a'},
                    "Espresso"),
                React.createElement("li", {key: 'b'},
                    "Cappuccino"),
                React.createElement("li", {key: 'c'},
                    "Moccacino")
            ]);
    
```



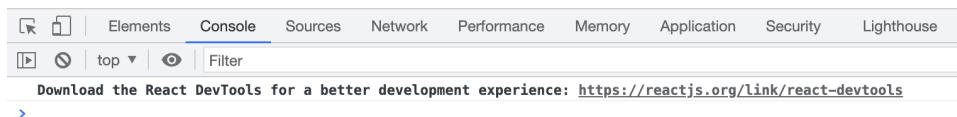
```
ReactDOM.createRoot(document.getElementById("root")).ren
der(myMenu);
</script>
</body>

</html>
```

ii. Hasil

Belajar React

- Espresso
- Cappuccino
- Moccacino



iii. Penjelasan

1. Sama dengan penjelasan di atas. Namun, ini diberi key pada setiap elemennya. Sehingga peringatan yang ada pada hasil di atas hilang.

2. Soal

a. 1

i. Kode Program

```
soal.html

<!-- *** Author : Akhmad Ramadani *** -->
<!-- *** NIM      : 2041720002 *** -->
<!-- *** Kelas    : TI-3B   *** -->

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible"
content="IE=edge">
        <meta name="viewport" content="width=device-width,
initial-scale=1.0">
        <title>React Uncover</title>
    </head>

<body>
    <h1>Belajar React</h1>

    <div id="root"></div>

    <script src="js/react.development.js"
crossorigin></script>
    <script src="js/react-dom.development.js"
crossorigin></script>
    <script>
        const myMenu = React.createElement("ul",
            {},
            [
                React.createElement("li", {key: 'a'},
                    React.createElement("a", {href:
"https://www.google.com"}, "Espresso")),
                React.createElement("li", {key: 'b'},
                    React.createElement("a", {href:
"https://www.google.com"}, "Cappucino")),
                React.createElement("li", {key: 'c'},
                    React.createElement("a", {href:
"https://www.google.com"}, "Moccacino"))
            ]);
    
```



```
ReactDOM.createRoot(document.getElementById("root")).ren
der(myMenu);
    </script>
</body>

</html>
```

ii. Hasil

Belajar React

- Espresso
- Cappucino
- Moccacino

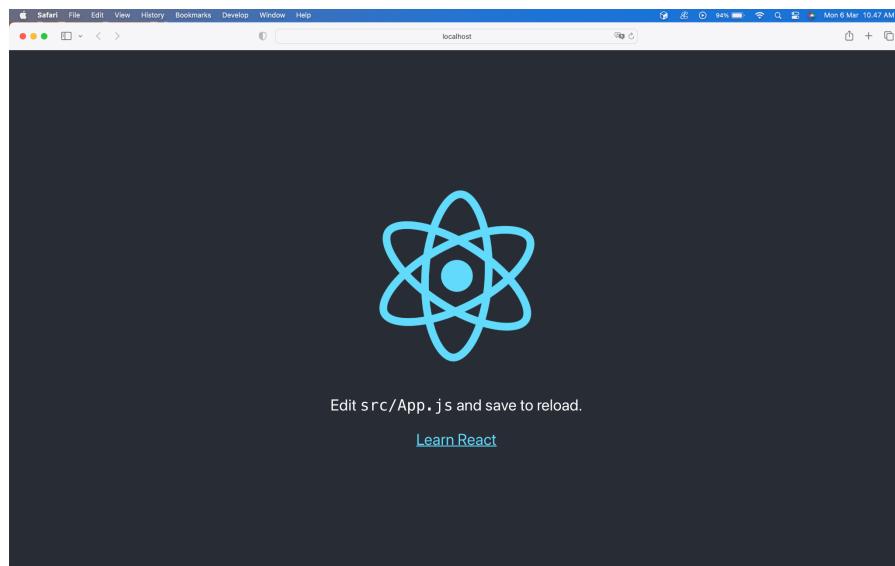
iii. Penjelasan

1. Menambahkan nested lagi pada tiap element dengan element “a” untuk membuat hyperlink.
2. Kemudian, pada tag “a” memiliki nilai object yang di dalamnya terdapat href.
3. Kemudian, tag “a” juga memiliki nilai dari dirinya sendiri seperti “Moccacino”.

3. Praktikum

a. Membuat Stateless component & stateful component

i. Halaman react pada awal pembuatan



ii. Kode Program

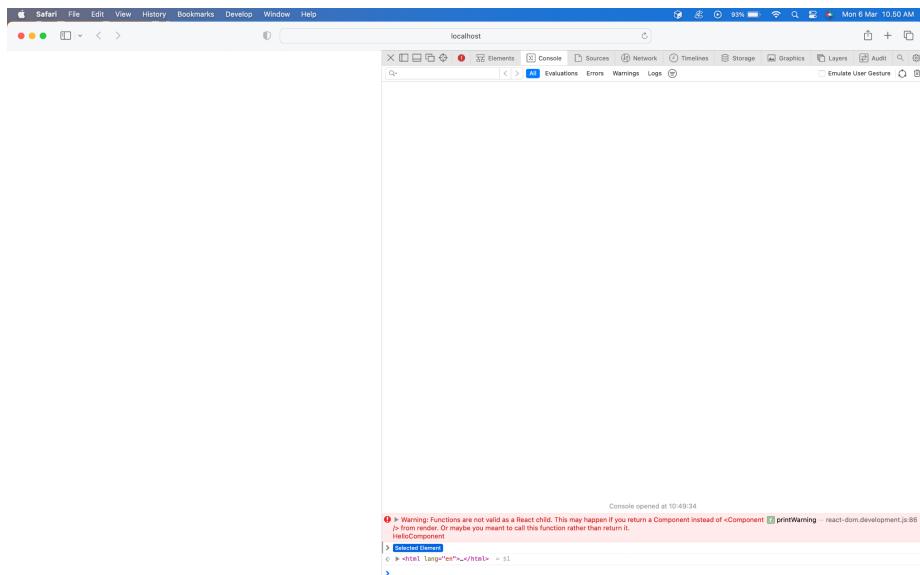
index.js

```
import React from 'react';
import ReactDOM from 'react-dom/client';
import './index.css';
import App from './App';
import reportWebVitals from './reportWebVitals';

function HelloComponent() {
  return HelloComponent;
}

const root =
ReactDOM.createRoot(document.getElementById('root'));
root.render(
  <React.StrictMode>
    <HelloComponent />
  </React.StrictMode>
);
reportWebVitals();
```

iii. Hasil



iv. Penjelasan

1. Fungsi HelloComponent merupakan sebuah komponen React yang sederhana, namun pengembangannya tidak dilakukan dengan benar, karena mengembalikan komponen itu sendiri. Sehingga nantinya akan terdapat error jika kita lihat pada console.log

v. Kode Program

```
index.js

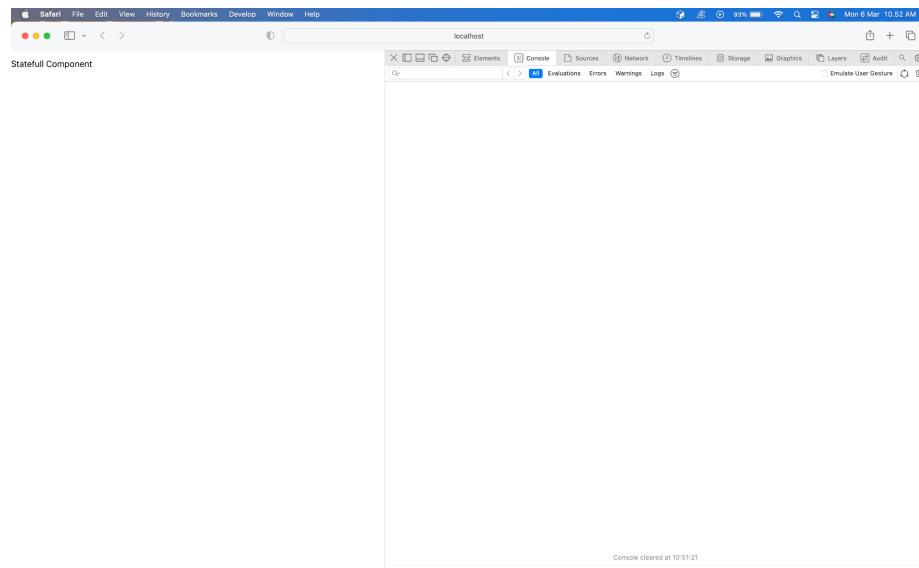
import React from 'react';
import ReactDOM from 'react-dom/client';
import './index.css';
import App from './App';
import reportWebVitals from './reportWebVitals';

function HelloComponent() {
    return HelloComponent;
}

class StateFullComponent extends React.Component {
    render() {
        return <p>Statefull Component</p>;
    }
}

const root =
ReactDOM.createRoot(document.getElementById('root'));
root.render(
    <React.StrictMode>
        <StateFullComponent />
    </React.StrictMode>
);
reportWebVitals();
```

vi. Hasil



vii. Penjelasan

1. Mendefinisikan sebuah komponen kelas StateFullComponent yang akan merender sebuah paragraf dengan teks Statefull Component.
2. Kemudian, kita menggunakan ReactDOM.createRoot untuk membuat root yang akan digunakan untuk merender aplikasi React, dan kita menentukan elemen HTML yang akan digunakan sebagai tempat rendering, yaitu dengan menggunakan document.getElementById('root').
3. Setelah itu, kita menggunakan root.render untuk merender komponen kita, yang di dalamnya kita menempatkan komponen StateFullComponent yang dikelilingi oleh <React.StrictMode>.
4. Tujuan dari <React.StrictMode> adalah untuk memberikan peringatan ketika ada kode yang memicu peringatan dan juga untuk memicu beberapa perilaku pembatasan pada mode pengembangan.
5. Sehingga, pada saat halaman dirender, akan terlihat tampilan teks "Statefull Component" yang dihasilkan oleh komponen StateFullComponent.

b. Membuat Props

i. Kode Program

index.js

```
import React from 'react';
import ReactDOM from 'react-dom/client';
import './index.css';
import App from './App';
import reportWebVitals from './reportWebVitals';

const root =
ReactDOM.createRoot(document.getElementById('root'));
root.render(
  <React.StrictMode>
    <App />
  </React.StrictMode>
);

reportWebVitals();
```

App.js

```
// import logo from './logo.svg';
import './App.css';
import React, {Component} from 'react';
import Header from './components/Header';
import Footer from './components/Footer';
import List from './components/List';

class App extends Component {
  render() {
    return (
      <div>
        <Header />
        <h1>Component dari Class App</h1>
        <List />
        <Footer judul="Halaman Footer" nama="Akhmad
Ramadani"/>
      </div>
    );
  }
}

export default App;
```

./components/Header.js

```
import React from 'react'; // Component menggunakan
Functions
const Header = (props) => {
  return (
    <div>
      <h3>Halaman Header</h3>
      <h3>Component ini dibuat menggunakan
      Function bukan Class</h3>
    </div>
  );
}
```

```
}
```

```
export default Header;
```

./components/Footer.js

```
import React from 'react';// Component menggunakan
Functions
const Footer = (props) => {
    return (
        <div>
            <h3>Halaman Footer</h3>
            <h3>Component ini dibuat menggunakan
Function bukan Class</h3>
            <p>Nilai ini ditampilkan dari props:
{props.judul} </p>
            <p>Nama Saya: {props.nama} </p>
        </div>
    );
}
export default Footer;
```

./components/Image.js

```
import React, { Component } from 'react';
class Image extends Component {
    render() {
        return (
            <img src={ this.props.linkGambar } alt="Food"
width='500' />
        );
    }
}
export default Image;
```

./components/List.js

```
import React, { Component } from 'react';
import Image from './Image';
class List extends Component {
    render() {
        return (
            <div>
                <ol>
                    <li>
                        Satu
                        <Image
                            linkGambar='https://s3-ap-southeast-1.amazonaws.com/niom
ic/img/sample/food1.jpg' />
                    </li>
                    <li>
                        Dua
                        <Image
                            linkGambar='https://s3-ap-southeast-1.amazonaws.com/niom
ic/img/sample/nasipadang.jpg' />
                    </li>
                    <li>
                        Tiga
                        <Image
                            linkGambar='https://s3-ap-southeast-1.amazonaws.com/niom
ic/img/sample/sate.png' />
                    </li>
                </ol>
            </div>
        );
    }
}
export default List;
```

```
        </li>
        <li>
            Empat
            <Image
                linkGambar='https://s3-ap-southeast-1.amazonaws.com/niom
ic/img/sample/sotolamongan.png' />
        </li>
    </ol>
    </div>
);
}
}export default List;
```

ii. Hasil

Halaman Header

Component ini dibuat menggunakan Function bukan Class

Component dari Class App

1. Satu

2. Dua

3. Tiga

4. Empat

Halaman Footer

Component ini dibuat menggunakan Function bukan Class

Nilai ini ditampilkan dari props: Halaman Footer

Nama Saya: Akhmad Ramadani

iii. Penjelasan

1. App.js

- a. File tersebut merupakan file utama dari sebuah aplikasi React yang terdiri dari beberapa komponen, yaitu Header, Footer, dan List. Komponen App sendiri merupakan sebuah class yang extends dari Component di dalam library React, dan memiliki

sebuah method render() yang mengembalikan tampilan dari komponen-komponen tersebut.

- b. Pada code tersebut, komponen App memuat Header, List, dan Footer. Header dan Footer merupakan komponen yang sudah dibuat di dalam folder components, sementara List merupakan sebuah komponen yang masih harus dibuat.
- c. Komponen Header menampilkan tampilan header pada aplikasi, List menampilkan daftar, dan Footer menampilkan footer pada aplikasi. Komponen App menggabungkan ketiga komponen tersebut menjadi satu tampilan yang lengkap.

2. Header.js

- a. File tersebut adalah komponen React yang dibuat menggunakan Function Component, yaitu Header. Komponen ini memiliki parameter props yang berisi data yang dikirim dari komponen yang memanggilnya.
- b. Dalam file tersebut juga terdapat kode export default Header yang berfungsi untuk mengekspor komponen Header agar dapat digunakan pada file JavaScript lainnya.

3. Footer.js

- a. File tersebut adalah komponen React yang dibuat menggunakan Function Component, yaitu Footer. Komponen ini memiliki parameter props yang berisi data yang dikirim dari komponen yang memanggilnya. Komponen ini akan menampilkan elemen HTML berupa sebuah div dengan beberapa elemen anak di dalamnya, seperti h3 untuk judul halaman, p untuk menampilkan nilai dari props.judul dan props.nama.
- b. Dalam file tersebut juga terdapat kode export default Footer yang berfungsi untuk mengekspor komponen Footer agar dapat digunakan pada file JavaScript lainnya.

4. Image.js

- a. Mendefinisikan sebuah komponen bernama Image. Komponen ini merupakan sebuah class component, sehingga perlu mewarisi class Component yang didefinisikan di dalam pustaka react.
- b. Komponen Image memiliki satu buah method, yaitu method render(). Method ini akan dipanggil oleh React ketika komponen ini di-render ke dalam halaman web.

- c. Di dalam method render(), terdapat sebuah tag yang digunakan untuk menampilkan sebuah gambar di halaman web. Properti yang digunakan oleh tag tersebut diambil dari properti linkGambar yang diterima oleh komponen ini dari parent component. Properti tersebut diberikan nilai default berupa teks "Food" sebagai alternatif deskripsi gambar yang akan ditampilkan, dan juga memiliki lebar gambar sebesar 500 pixel. Kemudian, komponen ini diekspor dengan menggunakan perintah export default Image; sehingga bisa digunakan di file lain.
5. List.js
- a. Mendefinisikan komponen List yang merender daftar terurut dari item, masing-masing item berisi komponen Image. Komponen Image menampilkan gambar menggunakan atribut src, yang diberikan sebagai prop bernama linkGambar.
 - b. Pada komponen List, setiap elemen li berisi nomor dan komponen Image. Prop linkGambar dari komponen Image diberikan URL gambar, yang ditampilkan menggunakan tag img. URL-nya di-set secara hardcode untuk setiap item di daftar.

c. LifeCycle

i. Kode Program

```
index.js

import React from 'react';
import ReactDOM from 'react-dom/client';

class Test extends React.Component {
    constructor(props) {
        super(props);
        this.state = { hello: "World!" };
    }

    componentWillMount() {
        console.log("componentWillMount()");
    }

    componentDidMount() {
        console.log("componentDidMount()");
    }

    changeState() {
        this.setState({ hello: "Geek!" });
    }

    render() {
        return (
            <div>
                <h1>GeeksForGeeks.org,</h1>
                Hello{this.state.hello}</h1>
                <h2>
                    <a
                        onClick={this.changeState.bind(this)}>Press Here!</a>
                    </h2>
                </div>;
        )
    }

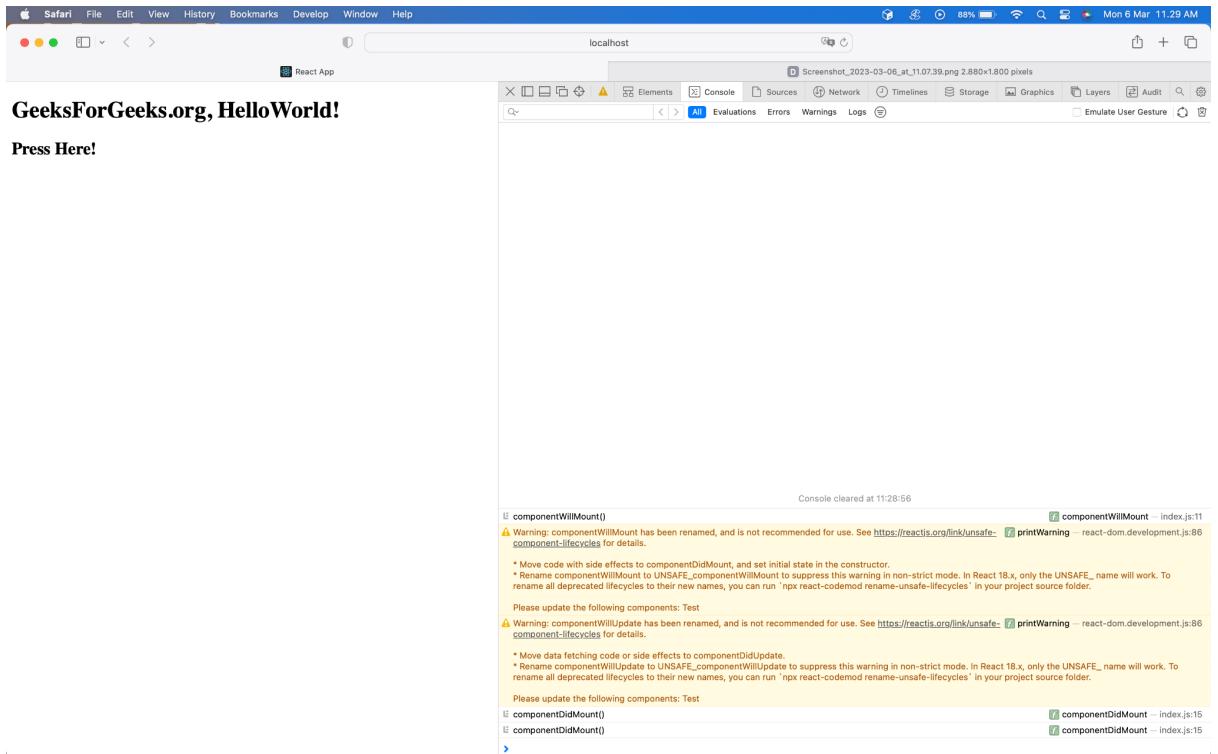
    shouldComponentUpdate(nextProps, nextState) {
        console.log("shouldComponentUpdate()");
        return true;
    }

    componentWillUpdate() {
        console.log("componentWillUpdate()");
    }

    componentDidUpdate() {
        console.log("componentDidUpdate()");
    }
}

const root =
ReactDOM.createRoot(document.getElementById('root'));
root.render(
    <React.StrictMode>
        <Test />
    </React.StrictMode>
);
```

ii. Hasil



iii. Penjelasan

1. `componentWillMount()`: fungsi ini akan dieksekusi tepat sebelum sebuah komponen di-mount atau ditampilkan di layar. Fungsi ini sering digunakan untuk melakukan persiapan awal sebelum tampilan komponen diperbarui.
2. `componentDidMount()`: fungsi ini akan dieksekusi setelah sebuah komponen berhasil di-mount atau ditampilkan di layar. Fungsi ini sering digunakan untuk memanggil data dari API atau melakukan manipulasi pada DOM setelah tampilan komponen ditampilkan.
3. `shouldComponentUpdate()`: fungsi ini akan dieksekusi ketika sebuah komponen sedang mempertimbangkan apakah perlu melakukan pembaharuan tampilan atau tidak. Fungsi ini sering digunakan untuk melakukan optimasi performa, dengan memastikan bahwa pembaharuan tampilan hanya dilakukan ketika benar-benar diperlukan.
4. `componentWillUpdate()`: fungsi ini akan dieksekusi sebelum sebuah komponen diperbarui atau dirender ulang. Fungsi ini sering digunakan untuk melakukan persiapan sebelum tampilan komponen diperbarui.

5. componentDidUpdate(): fungsi ini akan dieksekusi setelah sebuah komponen berhasil diperbarui atau dirender ulang. Fungsi ini sering digunakan untuk melakukan manipulasi pada DOM atau memperbarui data pada komponen setelah tampilan komponen diperbarui.

d. Styling

i. Kode Program

```
index.js
<import React from 'react';
import ReactDOM from 'react-dom/client';
import './index.css';
import App from './App';
import reportWebVitals from './reportWebVitals';
import HelloComponent from
'./components/HelloComponent';

const root =
ReactDOM.createRoot(document.getElementById('root'));
root.render(
  <React.StrictMode>
    <HelloComponent />
  </React.StrictMode>
);

reportWebVitals();

HelloComponent.css
.text-p{
  color: red;
  font-size: 20px;
}

HelloComponent.js
import React from "react";
import '../styles/HelloComponent.css';

const HelloComponent = () => {
  return <p className="text-p">Ini adalah fungsi arrow
function dari HelloComponent</p>;
}

export default HelloComponent;
```

ii. Hasil



Ini adalah fungsi arrow function dari HelloComponent

iii. Penjelasan

1. Kode di atas merupakan sebuah komponen React bernama HelloComponent. Komponen ini terdiri dari sebuah fungsi arrow yang mengembalikan elemen p yang berisi teks. Elemen

p tersebut memiliki kelas CSS text-p, yang didefinisikan di dalam file CSS yang berbeda (./styles/HelloComponent.css).

2. Komponen HelloComponent ini diekspor agar dapat digunakan di tempat lain dengan menambahkan export default HelloComponent; di akhir kode.

4. Tugas Praktikum

a. Soal 1

i. Kode Program

Login.js

```
import React, { Component } from 'react';
import '../styles/LoginStyle.css';

class Login extends Component {

    render() {
        return (
            <div className='main-body'>
                <h1>Form Login</h1>
                <div className='form-login'>
                    <h1 className='title'>Tugas
Pertemuan Ketiga</h1>
                    <div className='form-line'>
                        <label
                            className='label-style'>Username</label>
                        <input type='text'
                            placeholder='Username' className='input-style' />
                    </div>
                    <div className='form-line'>
                        <label
                            className='label-style'>Password</label>
                        <input type='password'
                            placeholder='Password' className='input-style' />
                    </div>
                    <div>
                        <button
                            className='btn-login'>Login</button>
                    </div>
                    <div>
                        <input type='checkbox' />
                        <label>Remember Me</label>
                    </div>
                    <div>
                        <button
                            className='btn-cancel'>Cancel</button>
                    </div>
                </div>
            );
        }
    }

    export default Login;
```

LoginStyle.css

```
.main-body{
    width: 100%;
    height: 100vh;
    justify-content: center;
    align-items: center;
    display: flex;
    flex-direction: column;
```

```
}

.form-login{
    width: 400px;
    height: 450px;
    background-color: #fff;
    border-radius: 10px;
    box-shadow: 0 0 10px 0 rgba(0,0,0,0.2);
    padding: 20px;
    justify-content: center;
    align-items: center;
    display: flex;
    flex-direction: column;
}

.btn-login{
    width: 350px;
    height: 50px;
    background-color: #1dca7c;
    border-radius: 10px;
    color: #fff;
    font-size: 20px;
    font-weight: bold;
    border: none;
    cursor: pointer;
    margin-top: 20px;
    margin-bottom: 20px;
}

.btn-cancel{
    width: 100px;
    height: 50px;
    background-color: #ff0000;
    border-radius: 10px;
    color: #fff;
    font-size: 20px;
    font-weight: bold;
    border: none;
    cursor: pointer;
    margin-top: 20px;
    margin-bottom: 20px;
}

.input-style{
    width: 65%;
    height: 30px;
    border-radius: 10px;
    border: 1px solid #ccc;
    padding: 10px;
    font-size: 20px;
    margin-bottom: 20px;
    margin-left: 20px;
}

.label-style{
    font-size: 20px;
    font-weight: bold;
    margin-bottom: 10px;
}
```

```
.form-line{  
    width: 100%;  
    display: flex;  
    flex-direction: row;  
    justify-content: center;  
    align-items: center;  
}
```

ii. Hasil

