

LAPORAN PEMROGRAMAN FRAMEWORK

“SESSION 3 React Component”



Oleh:

Nama : Bagus Satria Putra

Kelas : 3F

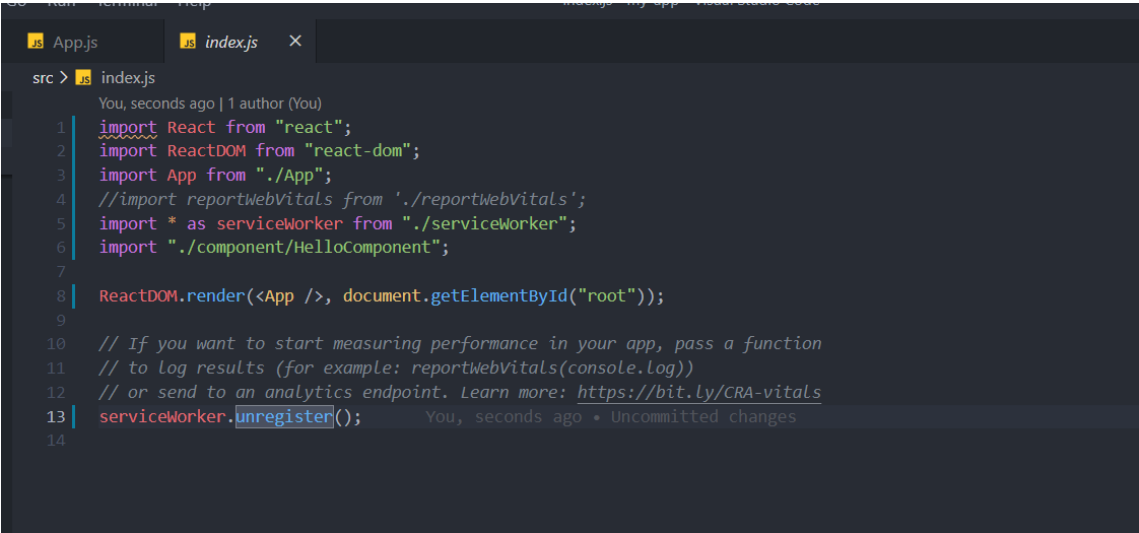
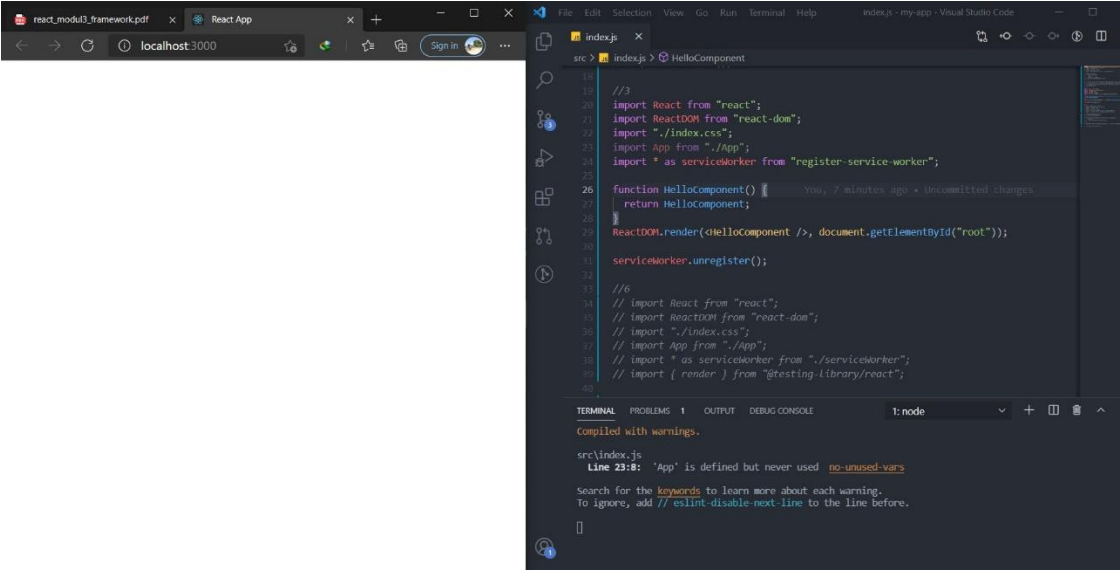
Absen/NIM : 8 / 1841720146

PROGRAM STUDI TEKNIK INFORMATIKA

JURUSAN TEKNOLOGI INFORMASI

POLITEKNIK NEGERI MALANG

23 FEBRUARI 2021

No	Keterangan
.	<p>Membuat Stateless component & statefull component</p> <p>Project sebelumnya create hello world</p> <p>Bagaimana cara membuat komponen dengan stateless component :</p> <ol style="list-style-type: none"> 1. Membuka halaman react 2. Buka index.js 3. Membuat function baru dengan hello component (stateless component) 4. Run di browser dan catat hasilnya  <p>Hasil :</p> 
	<p>Membuat Props</p> <p>Props atau properties digunakan untuk menampung nilai yang nantinya dikirimkan ke suatu Component.</p> <ol style="list-style-type: none"> 1. Buatlah Program berikut ini

```

...
4 import List from './List';class App extends Component {
5   render() {
6     return (
7       <div>
8         <Header/>
9         <h1>Component dari Class App</h1>
10        <List/>
11        <Footer judul='Halaman Footer' nama='Aufa' />
12      </div>
13    );
14  }
15 }export default App;
16

```

2. Simpan ke Component Footer atau File Footer.js.

Menampilkan props

1. Buatlah Program berikut ini

Hasil :

The screenshot shows a code editor with two tabs: 'App.js' and 'Footer.js'. The 'Footer.js' tab is active, displaying the following code:

```

1 import React from "react"; // Component menggunakan Functionconst
2
3 const Footer = (props) => {
4   return (
5     <div>
6       <h3>Halaman Footer</h3>
7       <h3>Component ini dibuat menggunakan Function bukan Class</h3>
8       <p>Nilai ini ditampilkan dari props: {props.judul} </p>
9       <p>Nama Saya: {props.nama} </p>
10    </div>
11  );
12 };
13
14 export default Footer;
15

```

Below the code editor, the 'TERMINAL' tab is active, showing the following output:

```

Line 8:8: 'Test' is defined but never used  no-unused-vars
Line 9:8: 'Hello' is defined but never used    no-unused-vars

```

Halaman Footer

Component ini dibuat menggunakan Function bukan Class

Nilai ini ditampilkan dari props: Halaman Footer

Nama Saya: Aufa

Pada kode di atas, kirim mengirimkan sebuah parameter props ke fungsi (anonymous function), lalu parameter tersebut akan kita gunakan untuk mengakses variabel yang dikirimkan dari file App.js.

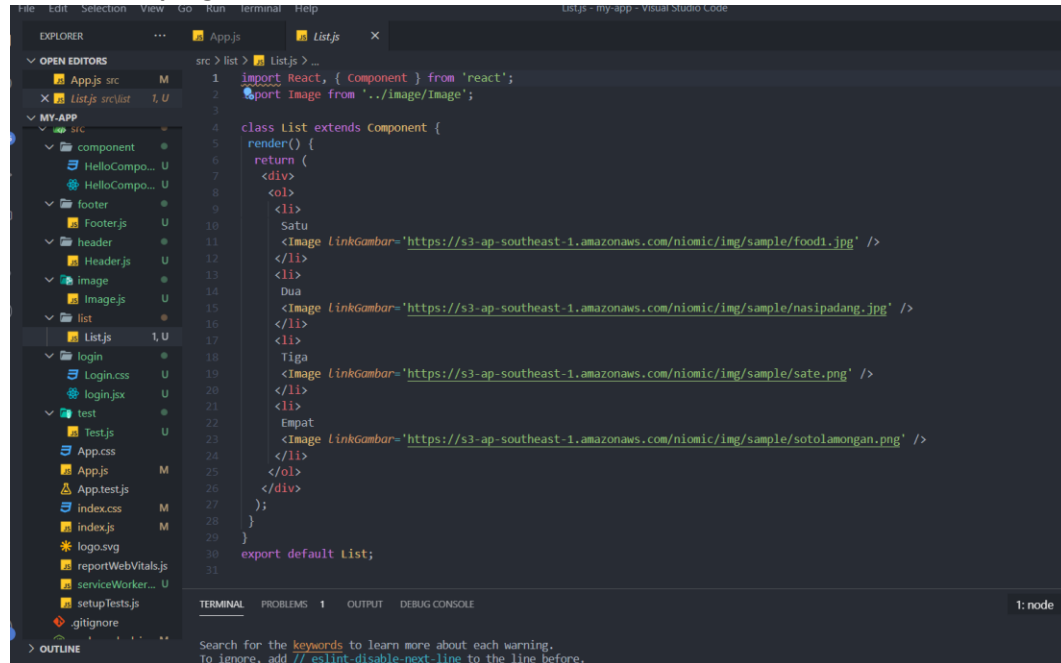
Props pada Class Component

Contoh Kasus:

Kita ingin menampilkan banyak gambar di dalam component list, di mana component list akan menggunakan component image.

Component List

1. Buatlah program berikut ini



```
1 import React, { Component } from 'react';
2 import Image from './image/Image';
3
4 class List extends Component {
5   render() {
6     return (
7       <div>
8         <ol>
9           <li>
10            Satu
11            <Image linkGambar='https://s3-ap-southeast-1.amazonaws.com/niomic/img/sample/food1.jpg' />
12           </li>
13           <li>
14            Dua
15            <Image linkGambar='https://s3-ap-southeast-1.amazonaws.com/niomic/img/sample/nasipadang.jpg' />
16           </li>
17           <li>
18            Tiga
19            <Image linkGambar='https://s3-ap-southeast-1.amazonaws.com/niomic/img/sample/sate.png' />
20           </li>
21           <li>
22            Empat
23            <Image linkGambar='https://s3-ap-southeast-1.amazonaws.com/niomic/img/sample/sotolamongan.png' />
24           </li>
25         </ol>
26       </div>
27     );
28   }
29   export default List;
```

2. Pada kode di atas, kita membuat empat buah list di mana masing masing list memanggil Component image. Kemudian Component Image di panggil dengan program berikut ini

Component Image

1. Buatlah Program berikut ini



```
1 import React, { Component } from 'react';
2
3 class Image extends Component {
4   render() {
5     return (
6       <img src={ this.props.linkGambar } alt="Food" width='500' />
7     );
8   }
9 }
10 export default Image;
```

Component dari Class App



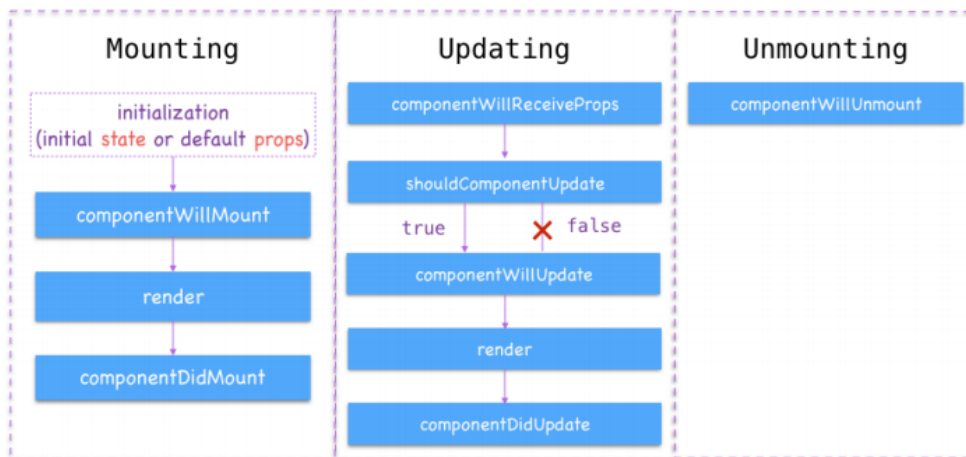
Lifecycle Component

Component — component di react setidaknya akan melewati tiga fase, yaitu :

1. **Mounting**
2. **Updating**
3. **Unmounting**

Pada setiap fase, components akan mengeksekusi fungsi atau method yang berbeda-beda yang di kenal sebagai lifecycle methods.

Lebih jelasnya perhatikan gambar di bawah ini !

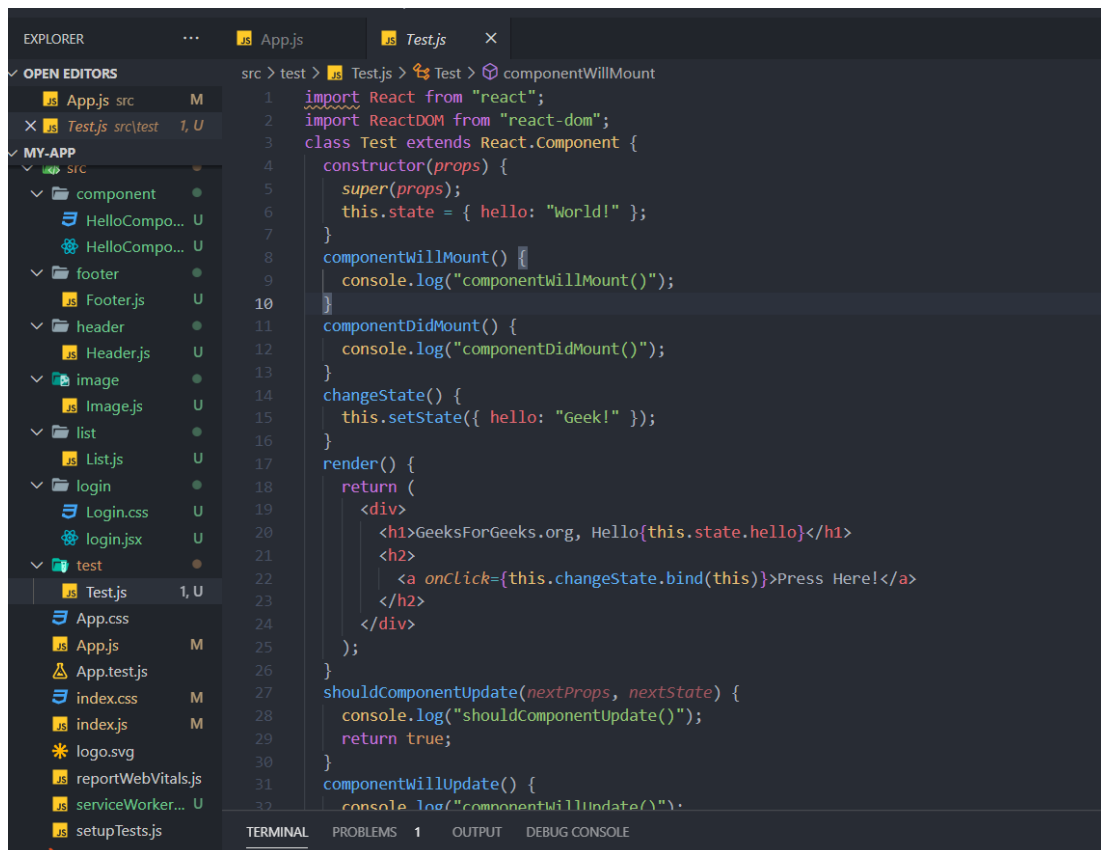


<https://rangle.github.io/react-training/react-lifecycles/>

Buatlah Program dibawah ini dan deteksilah bagian dari lifecycle Component

```
import React from 'react';
import ReactDOM from 'react-dom';
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, 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()



The screenshot shows a code editor with a file named `Test.js` open. The file contains the following code:

```
1 import React from "react";
2 import ReactDOM from "react-dom";
3 class Test extends React.Component {
4   constructor(props) {
5     super(props);
6     this.state = { hello: "World!" };
7   }
8   componentWillMount() {
9     console.log("componentWillMount()");
10  }
11  componentDidMount() {
12    console.log("componentDidMount()");
13  }
14  changeState() {
15    this.setState({ hello: "Geek!" });
16  }
17  render() {
18    return (
19      <div>
20        <h1>GeeksForGeeks.org, Hello{this.state.hello}</h1>
21        <h2>
22          <a onClick={this.changeState.bind(this)}>Press Here!</a>
23        </h2>
24      </div>
25    );
26  }
27  shouldComponentUpdate(nextProps, nextState) {
28    console.log("shouldComponentUpdate()");
29    return true;
30  }
31  componentWillUpdate() {
32    console.log("componentWillUpdate()");
33  }
34}
```

Below the code editor, there is a terminal window showing the following output:

```
componentDidMount() index.js:21
The development server has disconnected. webpackHotDevClient.js:76
Refresh the page if necessary.
shouldComponentUpdate() index.js:42
componentWillUpdate() index.js:48
componentDidUpdate() index.js:53
shouldComponentUpdate() index.js:42
componentWillUpdate() index.js:48
componentDidUpdate() index.js:53
```

2. Sehingga hasilnya berikut ini

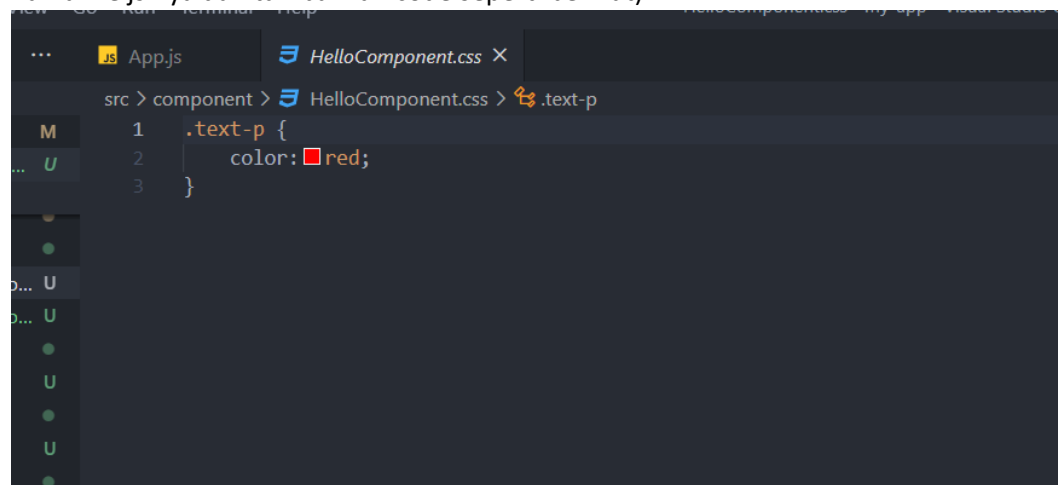
Hasil :

localhost:3000

GeeksForGeeks.org, HelloWorld!

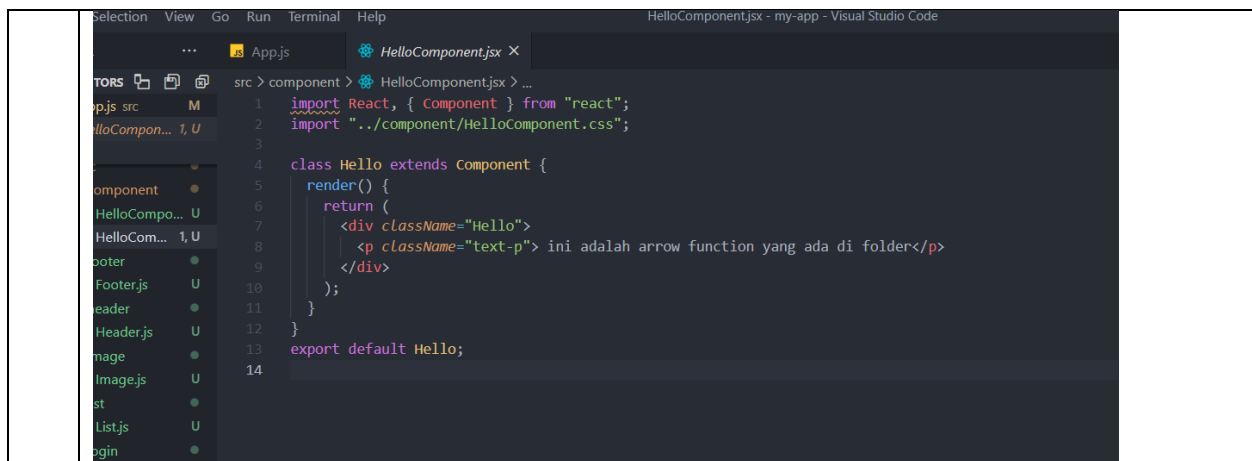
Press Here!

Selain kita dapat membuat file stateless component pada folder component, kita juga dapat menambahkan file css seperti pada gambar berikut (sebaiknya nama file css disamakan dengan nama file js nya dan tambahkan code seperti berikut)



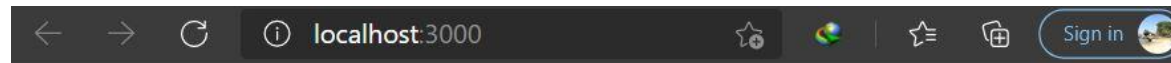
```
src > component > HelloComponent.css > .text-p
1  .text-p {
2    color: red;
3  }
```

▪ Setelah itu import HelloComponent.css pada HelloComponent.js seperti pada berikut



```
1 import React, { Component } from "react";
2 import "../component/HelloComponent.css";
3
4 class Hello extends Component {
5   render() {
6     return (
7       <div className="Hello">
8         <p className="text-p"> ini adalah arrow function yang ada di folder</p>
9       </div>
10     );
11   }
12 }
13 export default Hello;
```

Hasil :



ini adalah arrow function yang ada di folder

Tugas

Buatlah menu login seperti pada gambar menggunakan CSS

Login.css

```
form {
  border: 3px solid #f1f1f1;
  width: 350px;
}
input[type=text], input[type=password] {
  width: 100%;
  padding: 12px 20px;
  margin: 8px 0;
  display: inline-block;
  border: 1px solid #ccc;
  box-sizing: border-box;
}
```

```
button {
  background-color: #4CAF50;
  color: white;
  padding: 14px 20px;
  margin: 8px 0;
  border: none;
  cursor: pointer;
  width: 100%;
}

button:hover {
  opacity: 0.8;
}

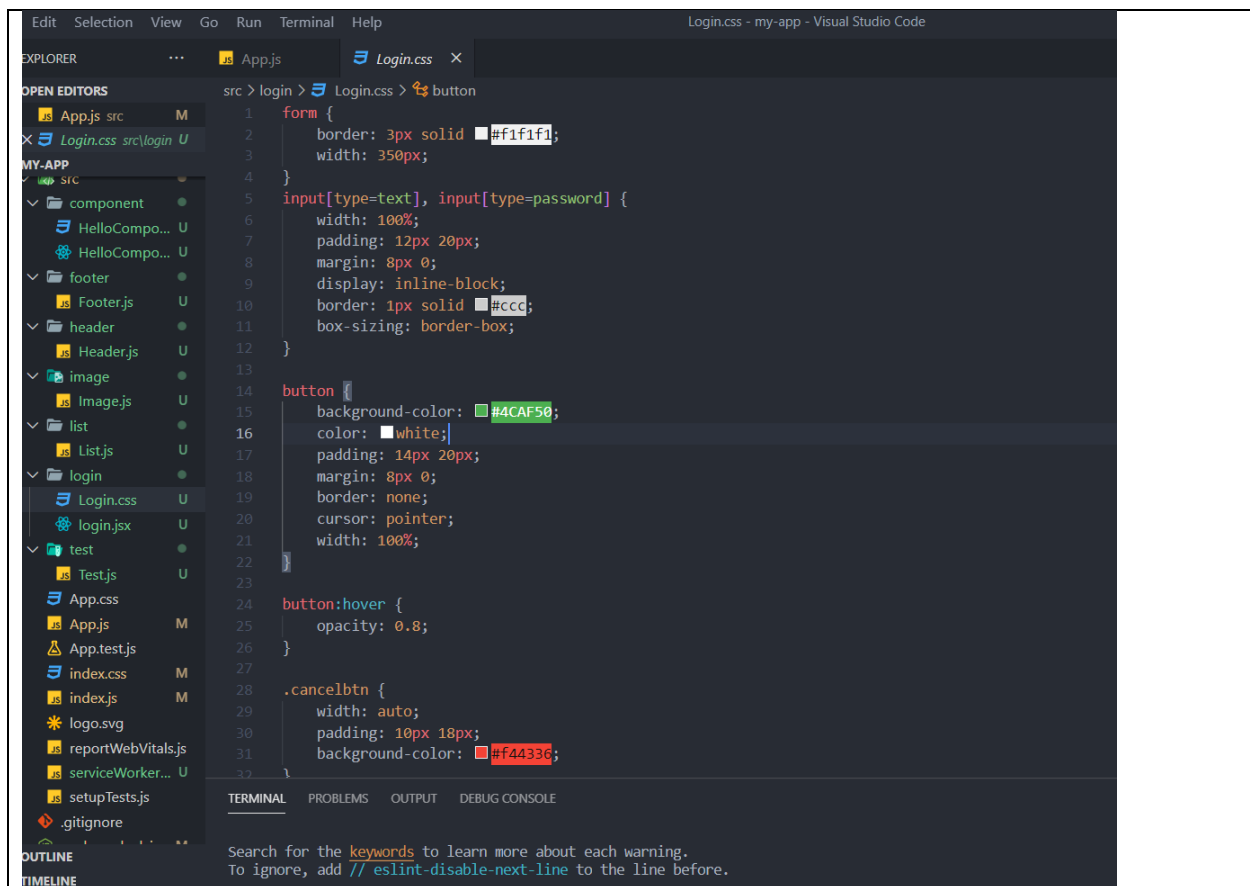
.cancelbtn {
  width: auto;
  padding: 10px 18px;
  background-color: #f44336;
}

.imgcontainer {
  text-align: center;
  margin: 24px 0 12px 0;
}

img.avatar {
  width: 40%;
  border-radius: 50%;
}

.container {
  padding: 16px;
}

span.psw {
  float: right;
  padding-top: 16px;
}
```



Login.jsx

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

class Login extends Component {

  render() {
    return (
      <div className="Login">
        <h2>Form Login</h2>
        <form>
          <div className="imgcontainer">
            <label><b>Tugas Pertemuan Ke Tiga</b></label>
          </div>

          <div className="container">
            <label><b>Username</b></label>
            <input type="text" placeholder="Enter Username" name="u
name" required/>

            <label><b>Password</b></label>
```

```

        <input type="password" placeholder="Enter Password" name="psw" required/>

        <button type="submit">Login</button>
        <input type="checkbox" defaultChecked/> Remember me
    </div>
    <div className="container" style={{backgroundColor: '#f1f1f1'}}>

        <button type="button" className="cancelbtn">Cancel</button>

        <span className="psw">Forgot password?</span>
    </div>
</form>
</div>
);
}
}

export default Login ;

```

The screenshot shows the Visual Studio Code editor with a project named 'my-app'. The file explorer on the left shows the project structure, including a 'login' directory with 'Login.css' and 'login.jsx'. The main editor displays the 'login.jsx' file, which defines a 'Login' class component. The component's render method returns a JSX element that includes a form with a title, username and password input fields, a login button, a 'Remember me' checkbox, a 'Forgot password?' link, and a 'Cancel' button. The form is styled with a light gray background. The terminal at the bottom shows a message about ESLint warnings.

```

class Login extends Component {
  render() {
    return (
      <div className="Login">
        <h2>Form Login</h2>
        <form>
          <div className="imgcontainer">
            <label><b>Tugas Pertemuan Ke Tiga</b></label>
          </div>
          <div className="container">
            <label><b>Username</b></label>
            <input type="text" placeholder="Enter Username" name="uname" required/>
            <label><b>Password</b></label>
            <input type="password" placeholder="Enter Password" name="psw" required/>
            <button type="submit">Login</button>
            <input type="checkbox" defaultChecked/> Remember me
          </div>
          <div className="container" style={{backgroundColor: '#f1f1f1'}}>
            <button type="button" className="cancelbtn">Cancel</button>
            <span className="psw">Forgot password?</span>
          </div>
        </form>
      </div>
    );
  }
}

export default Login ;

```

Search for the keywords to learn more about each warning.
To ignore, add // eslint-disable-next-line to the line before.

Hasil

react_modul3_framework.pdf

React App

localhost:3000

Sign in

Form Login

Tugas Pertemuan Ke Tiga

Username

Password

☒ Remember me

[Forgot password?](#)

--Alhamdulillah--