

Pengenalan Count menggunakan button dan form pada React

Pengertian:

React.js adalah pustaka JavaScript yang digunakan untuk membangun antarmuka pengguna yang interaktif dan dinamis. Salah satu fitur utama dalam React adalah kemampuan untuk mengelola state komponen menggunakan `useState`. State memungkinkan komponen untuk melacak dan merespon perubahan data.

Pembelajaran pada modul:

Dalam modul ini, mahasiswa diajarkan membuat sebuah komponen count sederhana menggunakan React.js.

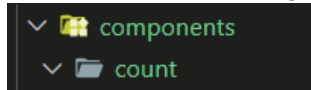
- Langkah pertama adalah membuat folder baru bernama `count` di dalam folder `components`, lalu membuat file `Count.js` di dalam folder tersebut.
- Selanjutnya, mahasiswa diminta untuk mengimport `useState` dari React ke dalam file `Count.js`.
- Komponen `Count` dibuat dengan menggunakan `export const Count` yang mendefinisikan operasi counter, dimulai dengan inisialisasi nilai awal 0 menggunakan `useState(0)`.
- Dua fungsi, `handleDecrement` untuk mengurangi angka dan `handleIncrement` untuk menambah angka, juga dideklarasikan.
- Pada bagian `return`, tampilan nilai awal count serta button `increment` dan `decrement` ditampilkan, dimana saat button diklik, nilai count akan berubah sesuai fungsi yang dipanggil.
- Modul ini juga mengilustrasikan tampilan awal website dan perubahan nilai count saat button diklik.
- Selanjutnya, mahasiswa diminta mengubah operasi button dari `-1` dan `+1` menjadi `-3` dan `+3` dengan menyesuaikan fungsi `handlernya`, serta memodifikasi tampilan button untuk mencerminkan perubahan fungsionalitas.
- Nama button `-1` diganti menjadi `Increment` dan `+1` menjadi `Decrement`.
- Tambahan button `reset` juga dibuat untuk mengembalikan nilai count ke 0 menggunakan `useState(0)`.
- Test case terakhir meminta mahasiswa membuat form dan button `Set` untuk menampilkan angka pada `<h1>{count}</h1>`, kemudian menguji fungsionalitas dengan berbagai input.

Kesimpulan:

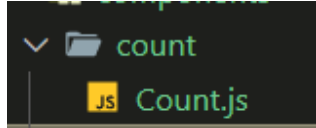
Dalam modul ini, mahasiswa belajar bagaimana membuat komponen counter sederhana menggunakan React.js. Dengan menggunakan `useState`, mahasiswa dapat mengelola state untuk operasi penambahan, pengurangan, dan reset pada counter. Proses pembelajaran ini mencakup pembuatan komponen, pengelolaan state, dan manipulasi DOM melalui React, memberikan pemahaman yang kuat tentang interaktivitas dan responsivitas aplikasi web berbasis React. Praktikum ini juga menekankan pentingnya pengujian fungsionalitas komponen melalui implementasi test case.

Langkah-langkah membuat sebuah count sederhana:

1. Buatlah folder baru dengan nama count pada folder components.



2. Kemudian buat file Count.js didalam folder count.



3. Selanjutnya di dalam file Count.js lakukan import useState dari package React.js.

```
import { useState } from "react";
```

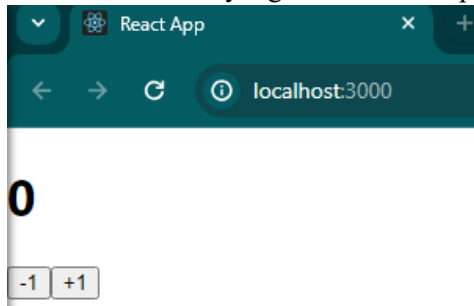
4. Kemudian membuat export const Count yang berfungsi untuk membuat sebuah operasi/proses dari count ini. Kemudian agar tampilan awal bilang 0 maka buatlah fungsi useState(0). Pada studi kasus ini anda akan membuat handleDecrement yang berfungsi untuk menambahkan bilangan dan handleIncrement yang berfungsi untuk mengurangi bilangan.

```
export const Count = () => {  
  const [count, setCount] = useState(0);  
  const handleDecrement = () => { setCount(count + 1);};  
  const handleIncrement = () => { setCount(count - 1);};
```

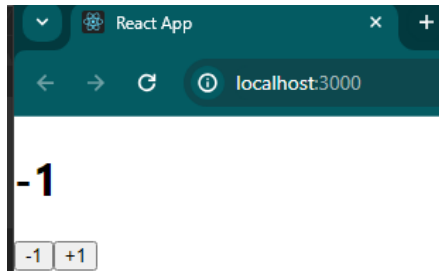
5. Kemudian isi dari pengembalian nilai/return adalah menampilkan nilai awal count, button increment yang ketika diklik maka akan mengurangi 1 angka dari angka awal, dan decrement yang ketika diklik maka akan menambahkan 1 angka dari angka awal.

```
  return (  
    <div>  
      <h1>{count}</h1>  
      <button onClick={handleIncrement}>-1</button>  
      <button onClick={handleDecrement}>+1</button>  
    </div>  
  );  
};
```

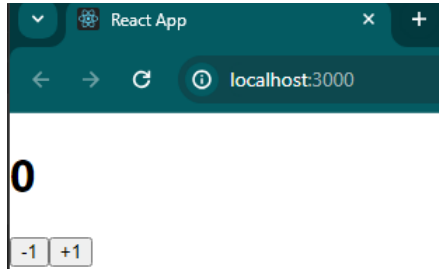
6. Hasil dari website yang telah dibuat seperti berikut ini.



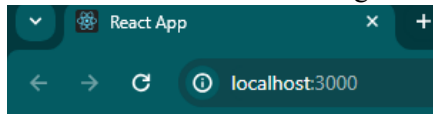
Jika button -1 diklik



Jika button +1 diklik dari nilai awal -1



Jika button +1 diklik sekali lagi.



1



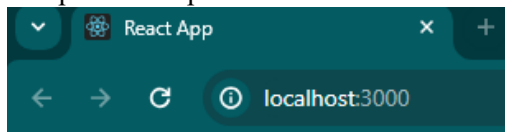
7. Selanjutnya mengubah button yang awalnya -1 dan +1 menjadi -3 dan +3 dengan mengubah operasi pada handlenya juga.

```
const handleDecrement = () => { setCount(count + 3);};  
const handleIncrement = () => { setCount(count - 3);};
```

8. Merubah tampilan button.

```
return (  
  <div>  
    <h1>{count}</h1>  
    <button onClick={handleIncrement}>-3</button>  
    <button onClick={handleDecrement}>+3</button>  
  </div>  
)
```

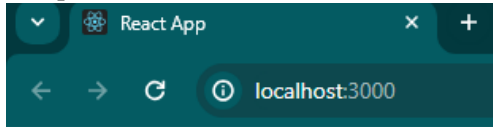
9. Tampilan awal pada website.



0



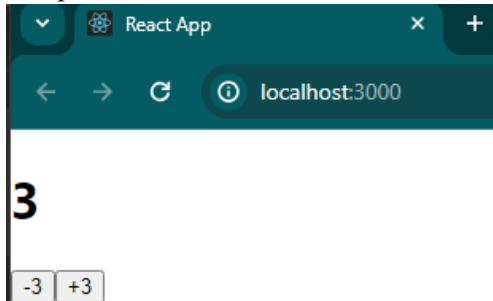
Tampilan setelah diklik button -3



-3



Tampilan setelah diklik +3



10. Selanjutnya ubah nama button -1 menjadi Increment, kemudian button +1 menjadi Decrement.

```
<button onClick={handleIncrement}>Increment</button>  
<button onClick={handleDecrement}>Decrement</button>
```

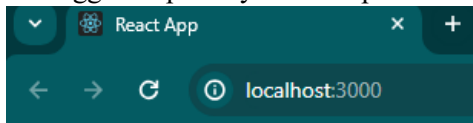
11. Kemudian membuat button reset agar angka yang sudah melakukan increment, decrement atau melewati angka default bisa Kembali lagi ke angka default yaitu 0.

```
<button onClick={() => setCount(0)}>Reset</button>
```

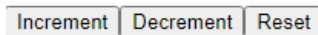
Button reset ini memanggil fungsi dari useState(0).

```
const [count, setCount] = useState(0)
```

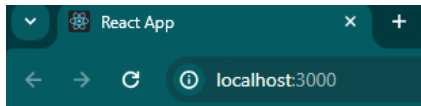
Sehingga tampilannya akan seperti berikut ini ketika sudah mengeklik button -1.



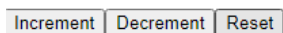
-1



Setelah di klik button reset.

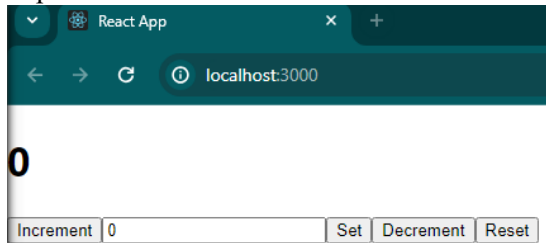


0

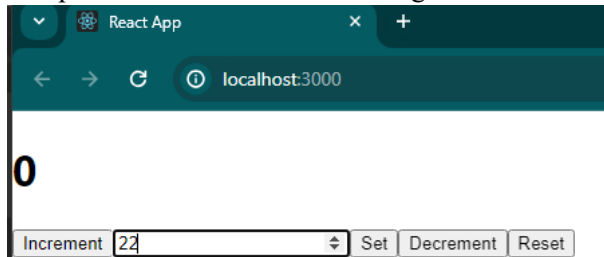


Test Case

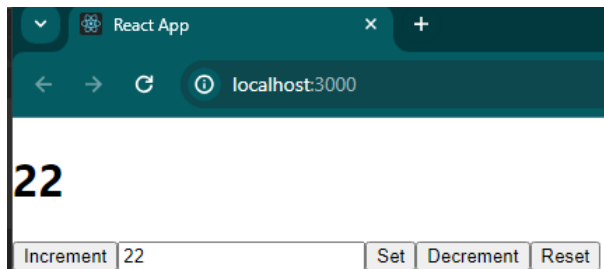
Buatlah sebuah form dan button Set untung menampilkan bilangan pada `<h1> {count} </h1>` Seperti berikut ini.



Tampilan form ketika diisi oleh angka.



Setelah di klik Set



Clue

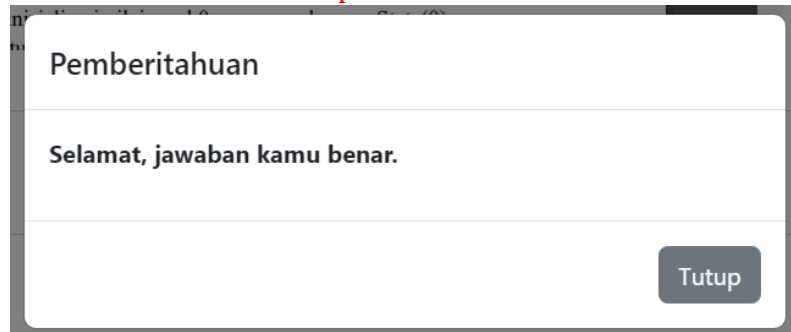
```
import { useState } from 'react'

export const Counter = () => {
  const [count, setCount] = useState(0)
  const [amount, setAmount] = useState(0)
  return (
    <div>
      <h1>{count}</h1>
      <button onClick={() => setCount(count - 1)}>Decrement</button>
      <input
        type="text"
        value={amount}
        onChange={(e) => setAmount(parseInt(e.target.value))}
      />
      <button onClick={() => setCount(amount)}>Set</button>
      <button onClick={() => setCount(count + 1)}>Increment</button>
      <button type="button" value="Reset" onClick={() => setCount(0)}>Reset</button>
    </div>
  )
}
```

Jika sudah upload file Counter.js ke dalam iCLOP:

Warning!

Jika file yang di Upload sesuai dengan perintah praktikum diatas maka akan muncul seperti berikut:



Jika file tidak sesuai dengan perintah diatas praktikum maka akan muncul seperti berikut:

