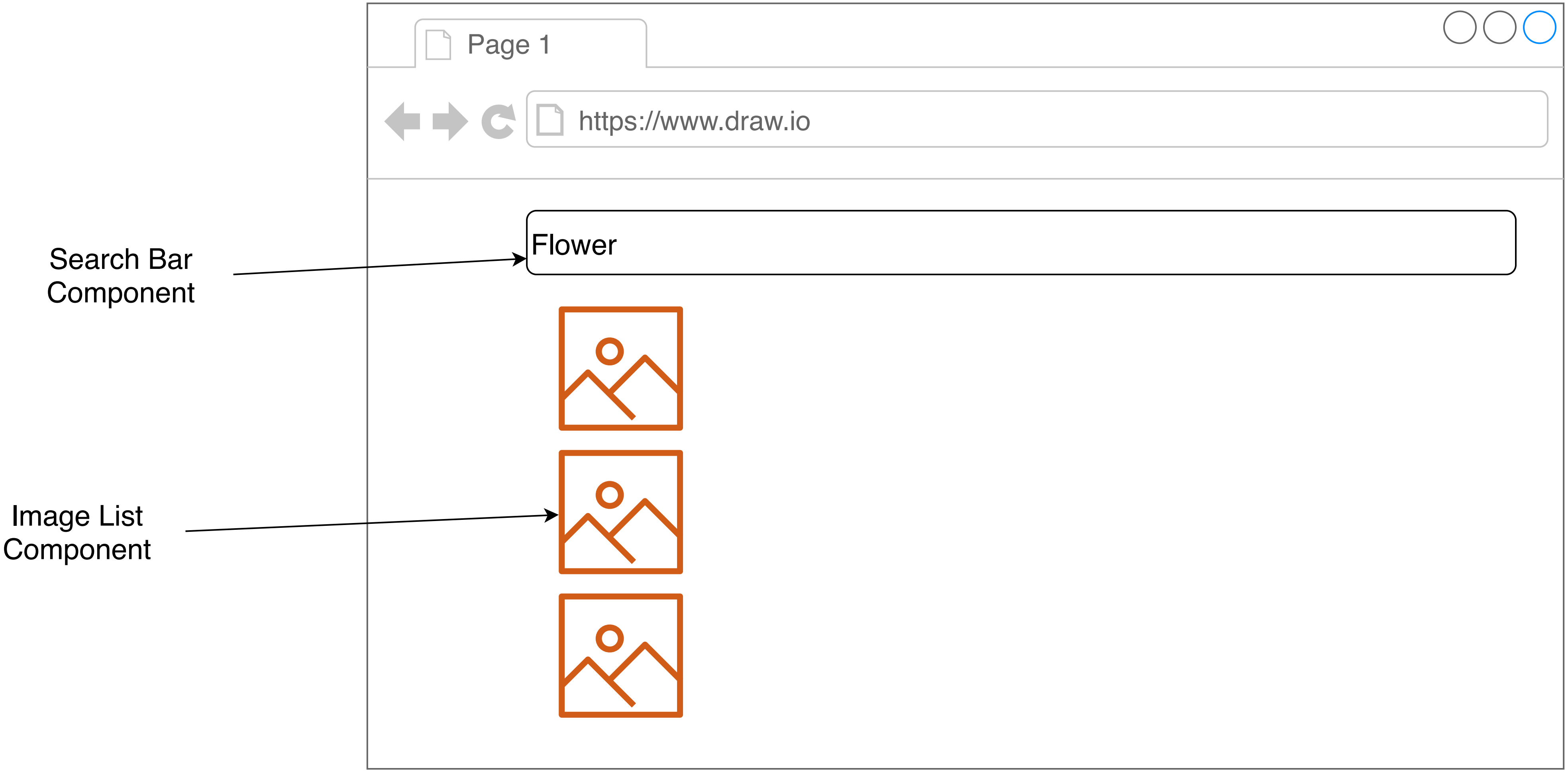
Hands On


Page 1
https://www.draw.io
Flower
Search Bar Component
Image List Component

Project Init Default

create-react-app search-image

kosongkan isi src

buat file src > App.js

buat function component App

buat file src > index.js

panggil ReactDOM, daftarkan
<App/> ke #root

ubah file public > index.html
daftar link stylesheet dengan css
framework

buat file src > SearchBar.js

```
import React from 'react';

class SearchBar extends React.Component {
    render() {
        return (
           <form>
                <div class="form-group">
                    <input type="text" class="form-control" id="searchBar"
                           placeholder="Search Image"/>
                </div>
            </form>
        );
    }
}

export default SearchBar;
```

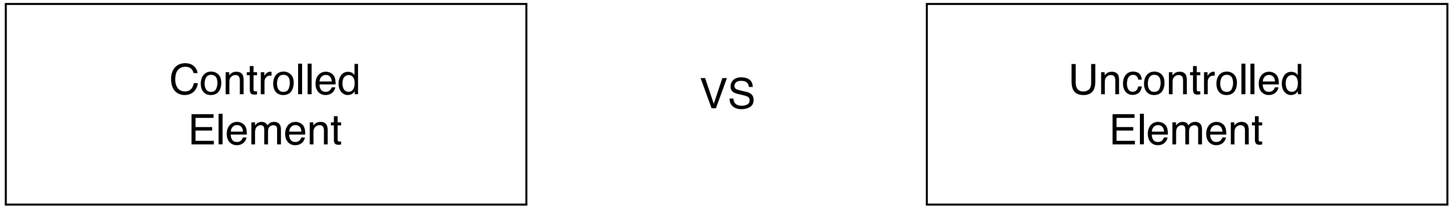
ubah file src > App.js

```
import React from 'react';

import SearchBar from './SearchBar';

class App extends React.Component {
    render() {
        return (
            <div className="container mt-3">
               <SearchBar/>
            </div>
        )
    };
}

export default App;
export default App;
```

buat file src > SearchBar.js

```
import React from 'react';

class SearchBar extends React.Component {
    onInputChange=(event)=>{
        console.log(event.target.value);
    }
    render() {
        return (
            <form>
                <div className="form-group">
                    <input type="text" className="form-control" id="searchBar"
                           placeholder="Search Image" onChange={this.onInputChange}/>
                </div>
            </form>

        );
    }
}

export default SearchBar;
```

Hati2... kita tidak menambahkan () di fungsi onInputChange,
karena kita ingin JSX mengeksekusi callback saat ada
perubahan text input, apabila () ditambahkan, maka akan
langsung dieksekusi pada saat render



Controlled
Element

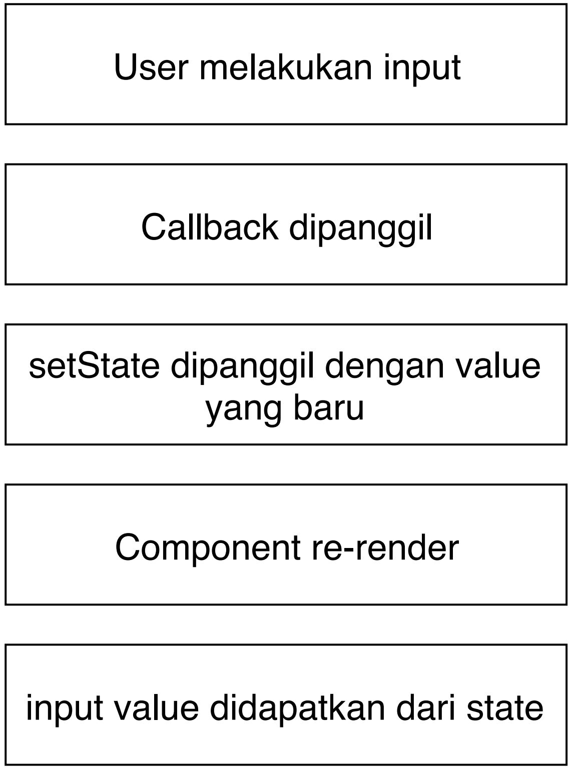
vs

Uncontrolled
Element

Ubah file src > SearchBar.js

```
import React from 'react';

class SearchBar extends React.Component {
   state = {searchKeyword : ''};

   onInputChange = (event)=>{
      this.setState({searchKeyword :event.target.value});
   }
   render() {
      return (
         <form>
            <div class="form-group">
               <input type="text" class="form-control" id="searchBar"
                    value={this.state.searchKeyword}
                    placeholder="Search Image" onChange={this.onInputChange}/>
            </div>
         </form>
      );
   }
}

export default SearchBar;
```

Dengan controlled element, kita bisa mengatur output tampilannya, seperti contoh, input hanya boleh angka/huruf, auto format angka, auto upper case



User melakukan input

Callback dipanggil

setState dipanggil dengan value yang baru

Component re-render

input value didapatkan dari state

Ubah file src > SearchBar.js

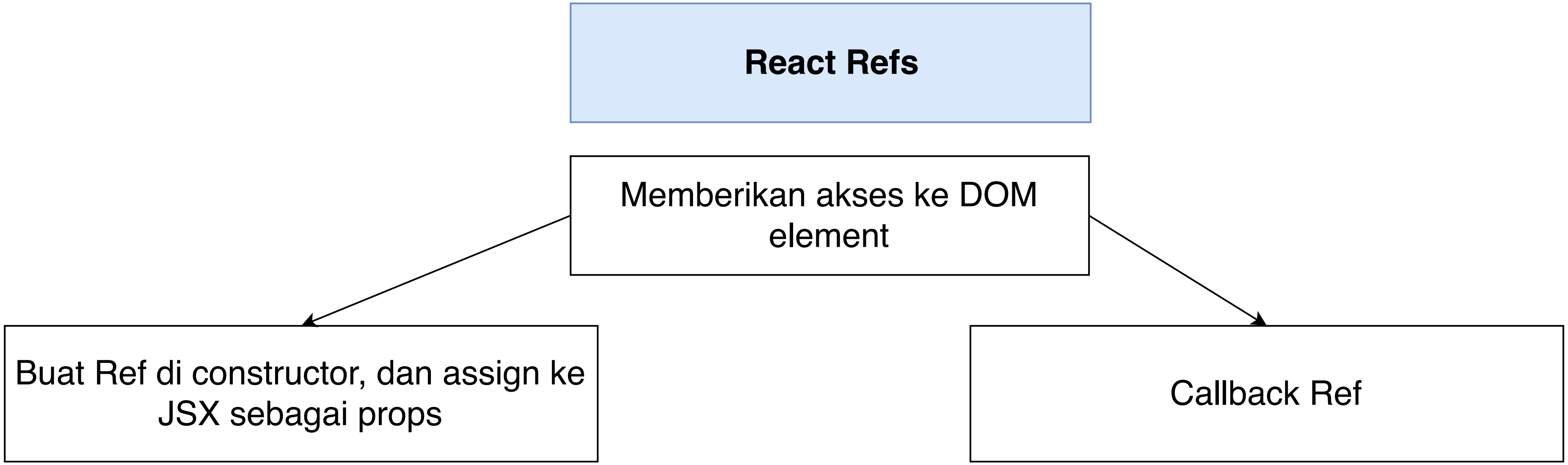
```
import React from 'react';

class SearchBar extends React.Component {
    state = {searchKeyword : ''};

    onFormSubmit = (event)=>{
        event.preventDefault();
        console.log(this.state.searchKeyword);
    }
    onInputChange=(event)=>{
        this.setState({searchKeyword :event.target.value});
    }
    render() {
        return (
           <form onSubmit={this.onFormSubmit}>
                <div class="form-group">
                    <input type="text" class="form-control" id="searchBar"
                           value={this.state.searchKeyword}
                           placeholder="Search Image" onChange={this.onInputChange}/>
                </div>
            </form>
        );
    }
}

export default SearchBar;
```

ubah file src > App.js

```
...
class App extends React.Component {

    onSearchSubmit=(term)=>{
        console.log(term);
    }

    render() {
        return (
            <div className="container mt-3">
               <SearchBar onSubmit={this.onSearchSubmit}/>
            </div>
        )
    };
}
...
```

Ubah file src > SearchBar.js

```
import React from 'react';

class SearchBar extends React.Component {
    state = {searchKeyword : ''};

    onFormSubmit = (event)=>{
        event.preventDefault();
        this.props.onSubmit(this.state.searchKeyword);
    }
    onInputChange(event){
        this.setState({searchKeyword :event.target.value});
    }
    render() {
        return (
            <form onSubmit={this.onFormSubmit}>
                <div class="form-group">
.
```



React Refs

Memberikan akses ke DOM element

Buat Ref di constructor, dan assign ke JSX sebagai props

Callback Ref

```
class UserFormRef extends React.Component {
   constructor() {
      super();
      this.myRef = React.createRef();
   }

focusTextInput() {
   this. myRef.current.focus();
 }

render() {
     return (
        <div>
           <label htmlFor='name'>Nama</label>
           <input ref={this.myRef} id='name'
type='text' />
        </div>
     )
  }
```

```
class UserFormRef extends React.Component {
  constructor(props) {
    this.textInput = null;

    this.setTextInputRef = element => {
      this.textInput = element;
    };
...

render() {
   return (
     <div>
       <input type="text" ref={this.setTextInputRef} />
...
```