

D-03 Training

React.js

Introduction



TABLE OF CONTENT

Day 3 - Sesi 1

01

What is React.js & JSX

02

Props, Children & Conditional Rendering

03

State & Render

04

Event & Navigation

1. Kenalan Dengan React.js

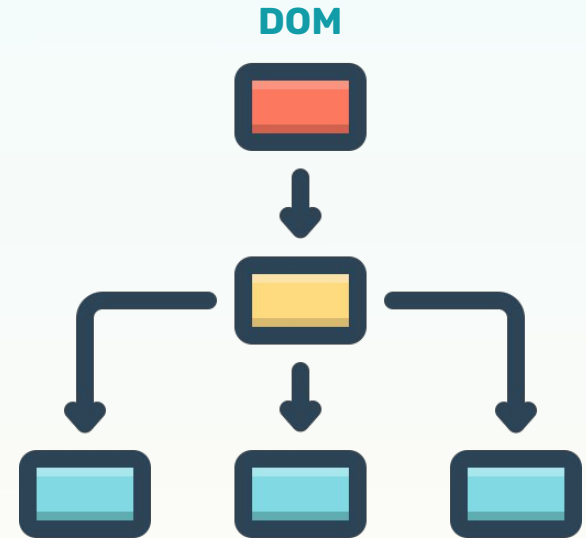
Tak kenal maka kenalan



Hal yang Harus Diketahui



Cara Website Memproses HTML

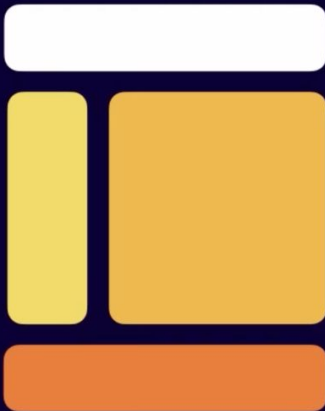


Vanilla JavaScript

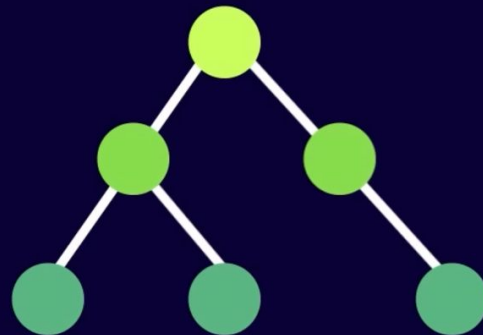
```
const btn = document.querySelector('#btn');  
btn.addEventListener('click', () => {  
  const div = document.querySelector('#div');  
  div.style.display = 'none';  
});
```

Cara Website Memproses HTML

COMPONENTS



DOM



Components everywhere~



Components everywhere~



Hubungan Investor

Informasi kinerja Perseroan terkini dan transparan bagi investor dan masyarakat



www.bankbsi.co.id
Bank Syariah Indonesia Call 14040

PT Bank Syariah Indonesia Tbk berizin & diawasi oleh Otoritas Jasa Keuangan serta merupakan Peserta Penjaminan

Harga Saham

2.530 ≈ -20 (-0,78%)

Tertunda 10 Menit
Transaksi terakhir: 14 Mar 2025 16:14

Volume

11.556.900

Rentang Hari Ini

Nama Emiten

Bank Syariah Indonesia

Kode Saham

BRIS

Komponent Pertama: Header



Komponent Kedua: Slider

Hubungan Investor

Informasi kinerja Perseroan terkini dan transparan bagi investor dan masyarakat



www.bankbsi.co.id
Bank Syariah Indonesia Call 14040

PT Bank Syariah Indonesia Tbk berizin & diawasi oleh Otoritas Jasa Keuangan serta merupakan Peserta Penjaminan



Komponent Ketiga: Saham

Harga Saham

2.530 ▼ -20 (-0,78 %)

Tertunda 10 Menit
Transaksi terakhir: 14 Mar 2025 16:14

Volume

11.556.900

Nama Emiten

Bank Syariah Indonesia

Rentang Hari Ini

2.520 - 2.560

Kode Saham

BRIS

Pertanyaan:
**Coba berikan daftar komponen pada contoh website
yang diberikan**

Sekarang Saatnya Kita Hands-on!

2. Props, Children & Conditional Rendering

Elemen tambahan pada pembuatan komponen



Bahas Analogi: Props

Props = Paket Pesanan dalam Pengiriman



Restoran = Komponen utama

Driver pengantar = Props

Makanan yang dipesan = Data yang dikirim
lewat props

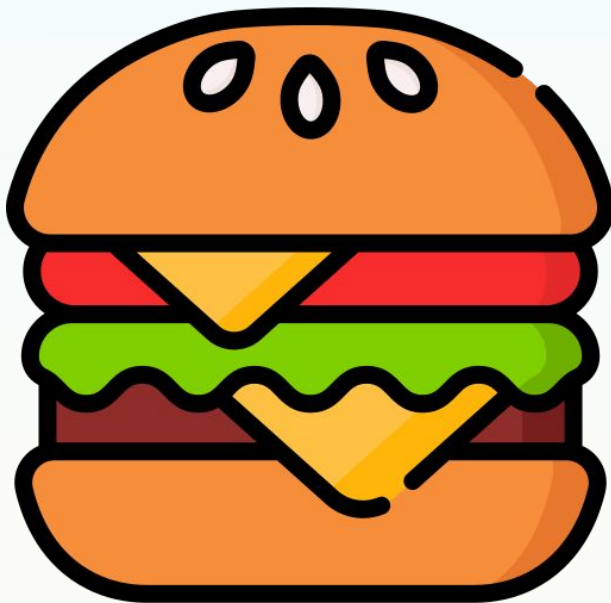
Contoh Penggunaan Props

```
function Welcome(props) {  
  return <h1>Hello, {props.name}!</h1>;  
}
```

// Digunakan seperti ini:

```
<Welcome name="Alice" /> // Output: Hello, Alice!
```

Bahas Analogi: Children



Roti = Komponen Utama

Isi Burger = Children:

- Jika kamu mau burger keju, kamu bisa isi dengan **keju** 🧀
- Jika kamu mau burger ayam, kamu bisa isi dengan **ayam** 🍗
- Jika kamu mau burger spesial, kamu bisa isi dengan **keju, ayam, dan saus ekstra!** 🌟

Contoh Penggunaan Children

```
function Burger(props) {  
  return (  
    <div className="burger">  
      🍔 {props.children} 🍔  
    </div>  
  );  
}  
  
// Digunakan seperti ini:  
<Burger>Keju</Burger>      // Output: 🍔 Keju 🍔  
<Burger>Ayam</Burger>      // Output: 🍔 Ayam 🍔  
<Burger>Keju & Ayam</Burger> // Output: 🍔 Keju & Ayam 🍔
```

Terus kapan saatnya menggunakan Props dan Children?

Pakai Props Saat:

Gunakan **props** ketika kita ingin **memberikan informasi spesifik ke dalam komponen**.



Bayangkan kamu memesan **pizza** melalui aplikasi.

- Kamu harus memilih ukuran (**besar/sedang/kecil**)
- Kamu harus memilih topping (**pepperoni, keju, jamur, dll.**)

Ketika kamu memilih topping dan ukuran, **kamu tidak bisa mengubahnya setelah pizza selesai dibuat.**

Pakai Children Saat:

Gunakan **children** ketika kita ingin memasukkan **konten fleksibel di dalam komponen**.



Bayangkan kamu punya **kotak hadiah** yang bisa diisi dengan apa saja.

- Kotaknya tetap sama (**komponen utama**)
- Isinya bisa berbeda (**children**), misalnya boneka, buku, atau coklat.

Dengan **children**, kita bisa membuat satu komponen yang isinya bisa berubah-ubah tanpa mengubah strukturnya.

Conditional Rendering

Bayangkan kamu punya **lampu ajaib** di kamar yang bisa menyala atau mati sendiri berdasarkan suatu kondisi!



- **Jika ada orang masuk ke kamar** → Lampu otomatis menyala 💡 ✨
- **Jika kamar kosong** → Lampu otomatis mati 🟡

Di React, **Conditional Rendering** bekerja seperti **lampu ajaib** ini. Kita bisa menampilkan atau menyembunyikan sesuatu berdasarkan kondisi tertentu.

Contoh Conditional Rendering

```
function Lampu(props) {  
  if (props.ada0rang) {  
    return <h2>💡 Lampu Menyala!</h2>;  
  } else {  
    return <h2>🚫 Lampu Mati!</h2>;  
  }  
}
```

// Cara menggunakannya:

```
<Lampu ada0rang={true} /> // Output: 💡 Lampu Menyala!  
<Lampu ada0rang={false} /> // Output: 🚫 Lampu Mati!
```

3. State & Render

Elemen tambahan pada pembuatan komponen



State



State Analogy: Super Mario



State Analogy: Super Mario [TLDR]



Breakdown!!

Di pojok layar ada **skor** yang menunjukkan berapa banyak koin yang sudah kamu kumpulkan.

- Awalnya skor = 0
- Setiap kali Mario mengambil koin, skor bertambah!
- Setiap kali Mario kena musuh, skor bisa berkurang!

Nah, skor ini adalah STATE!

📌 State adalah data yang bisa berubah dan disimpan dalam sebuah komponen.

State Analogy: Score Bola

UEFA Champions League · 20 Feb

Full-time



3

-

1



Real Madrid

Man City

Aggregate: 6 - 3

Kylian Mbappé 4', 33', 61'



Nico González 90+2'

Render Analogy: Super Mario



Render Analogy: Super Mario [TLDR]



Breakdown!!

Sekarang bayangkan **jika skor di layar tidak berubah saat Mario mengambil koin**. Itu akan membingungkan, kan? 🤔

Di React, saat state berubah, tampilan akan diperbarui secara otomatis!

Contoh Penggunaan State & Render

```
function Game() {  
  const [score, setScore] = React.useState(0); // State untuk skor  
  
  return (  
    <div>  
      <h2>🎮 Skor: {score}</h2>  
      <button onClick={() => setScore(score + 1)}>Ambil Koin! 🌐</button>  
    </div>  
  );  
}
```

4. Event & Navigation

Elemen tambahan pada pembuatan komponen



Event Analogy: Super Mario Kart



Event Analogy: Super Mario Kart [TLDR]



Breakdown!!

Bayangkan kamu sedang bermain **game balapan** 🏎️.

- Jika kamu **menekan tombol gas**, mobil akan melaju! 🏎️💨
- Jika kamu **menekan tombol rem**, mobil akan berhenti! 🛑
- Jika kamu **menekan tombol bel**, mobil akan berbunyi! 📢

Ini sama seperti Event di React!

Ketika kita **mengklik tombol** atau **melakukan sesuatu**, React bisa menjalankan aksi tertentu.

Contoh Event Dalam React

```
function Mobil() {  
  function jalan() {  
    alert("Mobil Melaju! 🏍️💨");  
  }  
  
  return <button onClick={jalan}>Tekan Gas! 🏍️</button>;  
}
```

Navigation Analogy: Super Mario Bus



Event Analogy: Super Mario Bus [TLDR]



Breakdown!!

Sekarang bayangkan kamu sedang naik **bus keliling kota!** 🚌

- Jika ingin pergi ke **Taman Bermain**, kamu pilih rute ke **Taman Bermain** 🎢
- Jika ingin pergi ke **Kebun Binatang**, kamu pilih rute ke **Kebun Binatang** 🦁
- Jika ingin pulang, kamu pilih rute ke **Rumah** 🏠

Ini sama seperti Navigasi di React menggunakan React Router!

Kita bisa pindah dari **halaman satu ke halaman lain** dengan menekan tombol, seperti memilih rute di peta.

Contoh Navigation Dalam React

```
import { BrowserRouter as Router, Route, Link, Routes } from "react-router-dom";

function App() {
  return (
    <Router>
      <nav>
        <Link to="/taman">Pergi ke Taman 🌳</Link> |
        <Link to="/kebun">Pergi ke Kebun Binatang 🐼</Link>
      </nav>

      <Routes>
        <Route path="/taman" element={<h2>🌳 Selamat datang di Taman Bermain!</h2>} />
        <Route path="/kebun" element={<h2>🐼 Selamat datang di Kebun Binatang!</h2>} />
      </Routes>
    </Router>
  );
}
```

Layout Analogy: Bus Kota



Layout Analogy: Super Mario House [TLDR]



Bayangkan kamu punya **rumah besar** dengan banyak kamar! 🏠

- Ruang tamu ada **sofa & TV** 📺
- Kamar tidur ada **kasur & lemari** 🛏️
- Dapur ada **kompur & kulkas** 🔪

Di React, layout digunakan untuk menyusun tampilan halaman supaya lebih rapi!

Contoh Layouting Dalam React

```
function Layout(props) {  
  return (  
    <div>  
      <header>📄 Ini Header</header>  
      <main>{props.children}</main>  
      <footer>📄 Ini Footer</footer>  
    </div>  
  );  
}  
  
function Home() {  
  return <h2>🏠 Selamat Datang di Rumah!</h2>;  
}  
  
function App() {  
  return (  
    <Layout>  
      <Home />  
    </Layout>  
  );  
}
```


Thank You

React.js

Introduction

