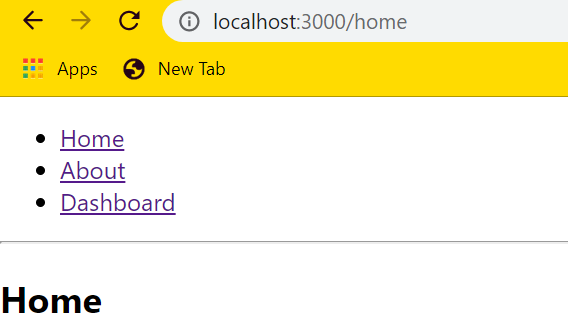
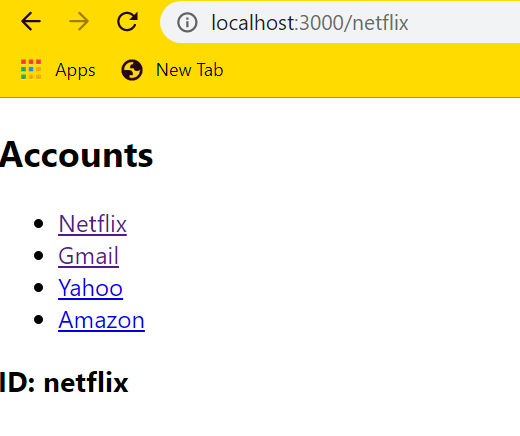
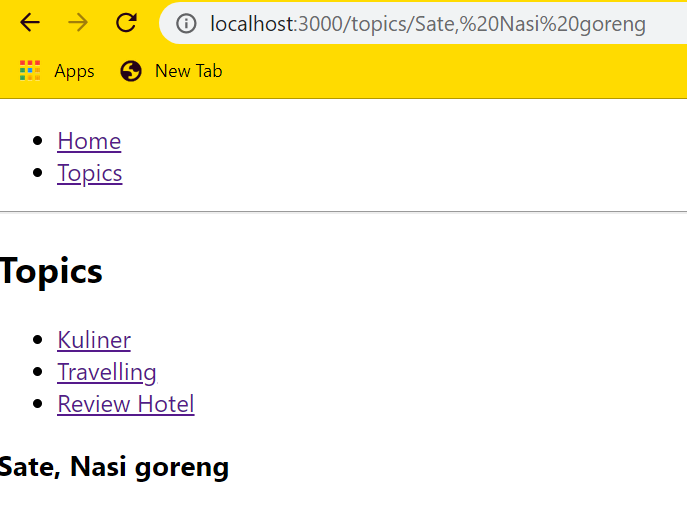
|  |
| --- |
| import React from 'react';  import {    BrowserRouter as Router,    Switch,    Route,    Link,    useParams  } from "react-router-dom";  export default function BasicExample(){    return (      <Router>        <div>          <ul>            <li>              <Link to="/home">Home</Link>            </li>            <li>              <Link to="/about">About</Link>            </li>            <li>              <Link to="/dashboard">Dashboard</Link>            </li>          </ul>          <hr/>          <Switch>            <Route exact path="/home">              <Home/>            </Route>            <Route path="/about">              <About/>            </Route>            <Route path="/dashboard">              <Dashboard/>            </Route>          </Switch>        </div>        </Router>    );  }  function Home(){    return (      <div>        <h2>Home</h2>      </div>    );  }  function About(){    return (      <div>        <h2>About</h2>      </div>    );  }  function Dashboard(){    return (      <div>        <h2>Dashboard</h2>      </div>    );    } |



|  |
| --- |
| import React from 'react';  import {    BrowserRouter as Router,    Switch,    Route,    Link,    useParams  } from "react-router-dom";  export default function ParamsExample(){    return (      <Router>        <div>          <h2>Accounts</h2>          <ul>            <li>              <Link to="/netflix">Netflix</Link>            </li>            <li>              <Link to="/gmail">Gmail</Link>            </li>            <li>              <Link to="/yahoo">Yahoo</Link>            </li>            <li>              <Link to="/amazon">Amazon</Link>            </li>            </ul>          <Switch>            <Route path="/:id" children={<Child />} />          </Switch>        </div>      </Router>    );  }  function Child(){    let {id} = useParams();    return(      <div>        <h3>ID: {id}</h3>      </div>    );  } |



|  |
| --- |
| import React from 'react';  import {    BrowserRouter as Router,    Switch,    Route,    Link,    useParams,    useRouteMatch  } from "react-router-dom";  export default function BasicExample(){    return (      <Router>        <div>          <ul>            <li>              <Link to="/">Home</Link>            </li>            <li>              <Link to="/topics">Topics</Link>            </li>          </ul>          <hr/>          <Switch>            <Route exact path="/">              <Home/>            </Route>            <Route path="/topics">              <Topics/>            </Route>            </Switch>        </div>      </Router>    );  }  function Home(){    return (      <div>        <h2>Home</h2>      </div>    );  }  function Topics(){    let { path, url } = useRouteMatch();    return (      <div>        <h2>Topics</h2>        <ul>        <li>          <Link to={`${url}/Sate, Nasi goreng`}>Kuliner</Link>        </li>        <li>          <Link to={`${url}/Wisata alam, Museum`}>Travelling</Link>        </li>        <li>          <Link to={`${url}/Ibis, JW Marriot`}>Review Hotel</Link>        </li>        </ul>        <Switch>          <Route exact path={path}>            <h3>Please select a topic.</h3>          </Route>          <Route path={`${path}/:topicId`}>            <Topic/>          </Route>        </Switch>      </div>    );  }  function Topic() {    let { topicId } = useParams();    return (      <div>        <h3>{topicId}</h3>      </div>    );  } |



|  |
| --- |
| import React from 'react';  import {    BrowserRouter as Router,    Switch,    Route,    Link,    useParams,    useRouteMatch,    Redirect,    useHistory,    useLocation  } from "react-router-dom";  export default function AuthExample(){    return(      <Router>        <div>          <AuthButton/>          <ul>            <li>              <Link to="/public">Public page</Link>            </li>            <li>              <Link to="/private">Private page</Link>            </li>          </ul>          <Switch>            <Route path="/public">              <PublicPage/>            </Route>            <Route path="/login">              <LoginPage/>            </Route>            <PrivateRoute path="/private">              <ProtectedPage/>            </PrivateRoute>          </Switch>        </div>      </Router>    );  }  const fakeAuth ={    isAuthenticated: false,    authenticate(cb) {      fakeAuth.isAuthenticated = true;      setTimeout(cb, 100);    },    signout(cb){      fakeAuth.isAuthenticated = false;      setTimeout(cb, 100);    }  };  function AuthButton() {    let history = useHistory();    return fakeAuth.isAuthenticated ? (      <p>        Welcome!{" "}        <button          onClick={() =>{            fakeAuth.signout(() => history.push("/"));          }}        >          Sign out        </button>      </p>    ) : (      <p> You are not logged in.</p>    );  }  function PrivateRoute({children, ...rest}) {    return (      <Route        {...rest}        render={({ location}) =>        fakeAuth.isAuthenticated ? (          children        ) : (          <Redirect           to= {{             pathname: "/login",             state: {from: location}           }}          />        )      }/>    );  }  function PublicPage(){    return <h3>Public</h3>;  }  function ProtectedPage(){    return <h3>Private</h3>;  }  function LoginPage() {    let history = useHistory();    let location = useLocation();    let {from} = location.state || { from: { pathname: "/"} };    let login = () => {      fakeAuth.authenticate(() => {        history.replace(from);      });    };    return (      <div>        <p>You Must log in to view the page at {from.pathname}</p>        <button onClick={login}>Log In</button>      </div>    );  } |

