strategy | consulting | digital | technology | operations

REACTJS – MODULE 6

ROUTING IN REACT

accenture

AGENDA

- Understand what is router, why router is required
- How to achieve router
- Demo
- Activity

accenture

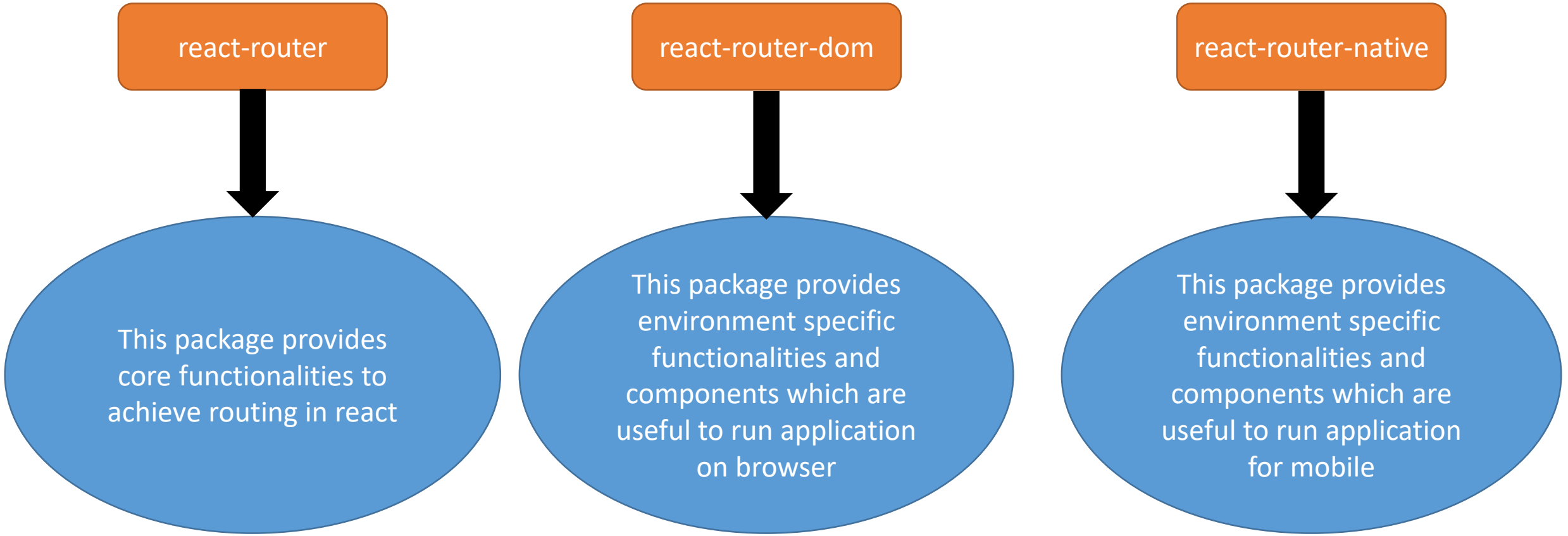
Introduction to Routing

Single page applications are todays most popular requirement in any application design which makes navigation easier

React provides the routing concept using which we can create single page applications

React provides mainly three packages to use routing concept.

Introduction to Routing (Cont...)



Introduction to Routing (Cont...)

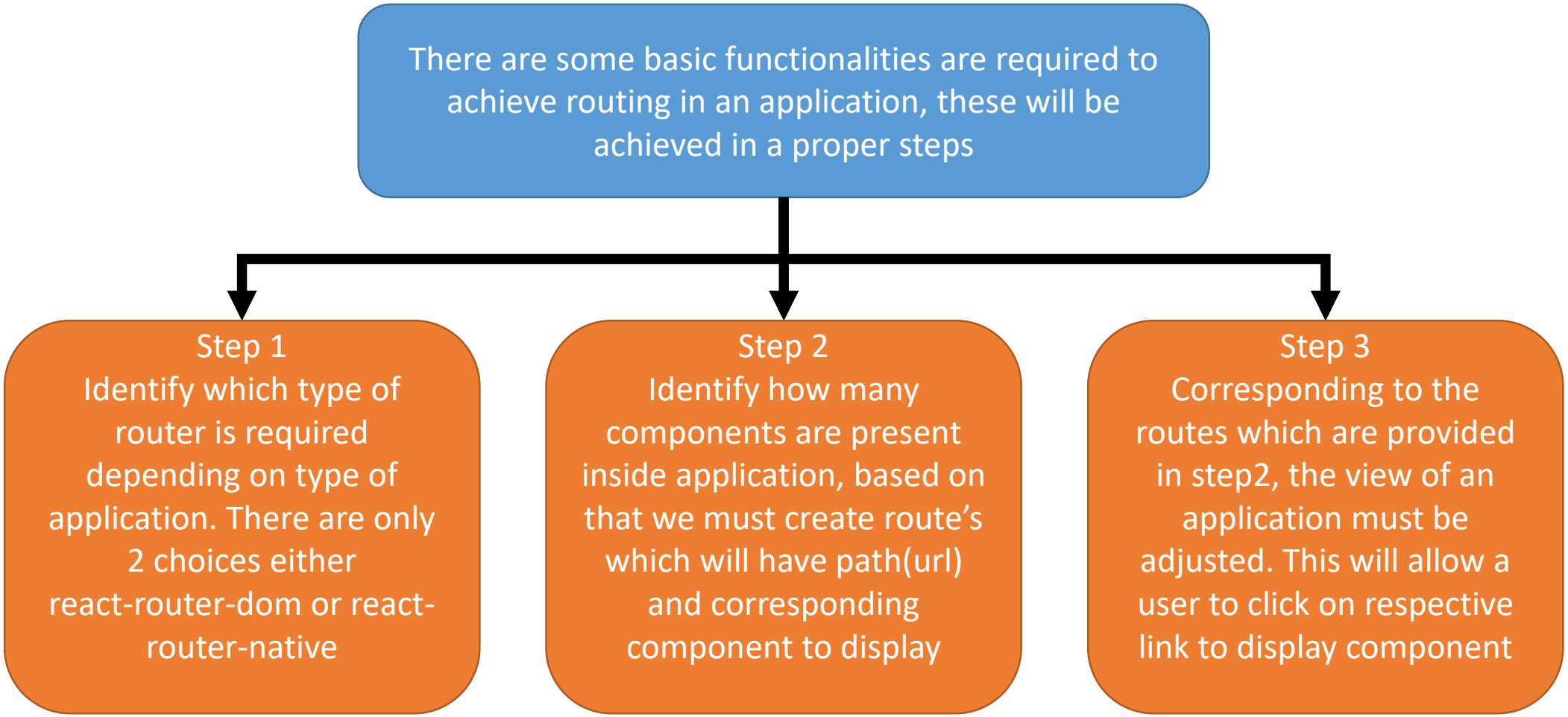
In this section we will focus on installing and using react-router-dom library to design web application

The following command is used to install router library
“npm install react-router-dom”

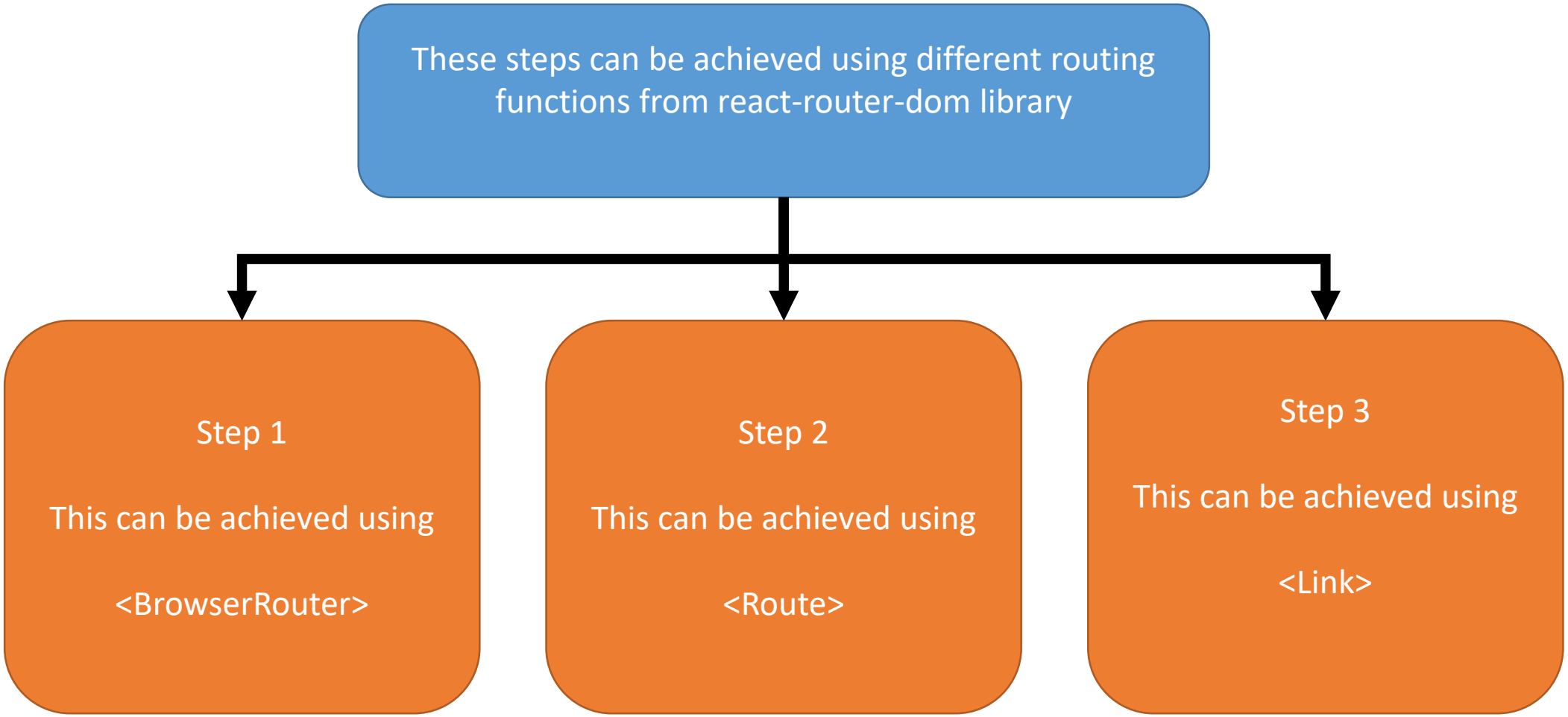
Once the above library is installed we can use the functionalities related to router and achieve the routing in our application

accenture

Introduction to Routing (Cont...)



Introduction to Routing (Cont...)



Introduction to Routing (Cont...) - <BrowserRouter>

This is very useful in understanding url and handle dynamic requests. This will contain all route related information within it

This will be imported from react-router-dom library either using the same name or by providing alias name during the import

Import {BrowserRouter} from 'react-router-dom'
Or
Import {BrowserRouter as Router} from 'react-router-dom'

Introduction to Routing (Cont...) - <Route>

Using this one can provide all route information. This will be imported using
Import {Route} from 'react-router-dom'

The Route element will have "path" and "component". "path" will have url or location and "component" will have name of the component

When the path is matched with appropriate url or location then the respective component is rendered

Introduction to Routing (Cont...) - <Link>

This element is mainly used to provide user a way to click on the link so that appropriate component can displayed on the view

This will be imported using
Import {Link} from 'react-router-dom'

Link element will have one of the important attribute known as "to". Here one must mention the path of a component

accenture

Demo – Routing

Create new react project
create-react-app reactexample

Install react library
npm install react-router-dom

```
C:\reactdemos\routerexample>npm install react-router-dom
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.1.2 (node_modules\react-scripts\node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.1.2: wanted {"os":"darwin","arch":"any
"} (current: {"os":"win32","arch":"x64"})
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.1.3 (node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.1.3: wanted {"os":"darwin","arch":"any
"} (current: {"os":"win32","arch":"x64"})

+ react-router-dom@4.2.2
added 231 packages in 49.515s
```

accenture

Demo – Routing (Cont…)

Open the project using webstorm and create new Home Component(Home.js)

```
import React,{Component} from 'react';

class Home extends Component{
    render(){
        return(
            <h2>This is Home Component</h2>
        )
    }
}

export default Home;
```

Demo – Routing (Cont…)

Create another component Customer Component(Customer.js)

```
import React,{Component} from 'react';

class Customer extends Component{
    render(){
        return(
            <h2>This is Customer component</h2>
        )
    }
}

export default Customer;
```

Demo – Routing (Cont…)

Inside index.js

```
import React from 'react';
import ReactDOM from 'react-dom';
import './index.css';
import App from './App';
import registerServiceWorker from './registerServiceWorker';
import { BrowserRouter as Router,Route} from 'react-router-dom';
import Home from './Home';
import Customer from "./Customer";

ReactDOM.render(
    <Router>
        <div>
            <Route path="/" component={App}/>
            <Route path="/Home" component={Home}/>
            <Route path="/Customer" component={Customer}/>
        </div>
    </Router>
    ,
    document.getElementById('root')
);
registerServiceWorker();
```

accenture

Demo – Routing (Cont…)

Create new component NavigationBar.js

```
import React,{Component} from 'react';
import {Link} from 'react-router-dom';
class NavigationBar extends Component{
    render(){
        return(
            <div>
                <div style={{'textAlign':'center','width':'100%',
                    'color':'green','backgroundColor':'pink','height':'60%'}}>
                    <ul>
                        <Link to="/Home">Home</Link> |
                        <Link to="/Customer">Customer</Link>
                    </ul>
                </div>
            </div>
        )
    }
}
export default NavigationBar;
```

Demo – Routing (Cont…)

Inside App.js

```
import React, { Component } from 'react';
import NavigationBar from './NavigationBar';
import './App.css';
class App extends Component {
  render() {
    return (
        <div>
      <div style={{'textAlign':'center','width':'100%','color':'blue',
          'backgroundColor':'lightgreen'}}>
          <h1>Welcome to Routing Example</h1>
      </div>
        <NavigationBar/>
         </div>
    );
  }
}
export default App;
```

Demo – Routing (Cont…) - Explanation

The Example contains two custom components
"Home" and "Customer"

```
import React,{Component} from 'react';

class Home extends Component{
    render(){
        return(
            <h2>This is Home Component</h2>
        )
    }
}

export default Home;
```

```
import React,{Component} from 'react';

class Customer extends Component{
    render(){
        return(
            <h2>This is Customer component</h2>
        )
    }
}

export default Customer;
```

Demo – Routing (Cont…) - Explanation

Inside index.js

```
import React from 'react';
import ReactDOM from 'react-dom';
import './index.css';
import App from './App';
import registerServiceWorker from './registerServiceWorker';
import { BrowserRouter as Router,Route} from 'react-router-dom';
import Home from './Home';
import Customer from "./Customer";

ReactDOM.render(
    <Router>
        <div>
            <Route path="/" component={App}/>
            <Route path="/Home" component={Home}/>
            <Route path="/Customer" component={Customer}/>
        </div>
    </Router>
    ,
    document.getElementById('root')
);
registerServiceWorker();
```

BrowserRouter is used here as the application is created to execute on the browser. Also the alias name is provided as Router

accenture

Demo – Routing (Cont…) - Explanation

Inside index.js

```
import React from 'react';
import ReactDOM from 'react-dom';
import './index.css';
import App from './App';
import registerServiceWorker from './registerServiceWorker';
import { BrowserRouter as Router,Route} from 'react-router-dom';
import Home from './Home';
import Customer from "./Customer";

ReactDOM.render(
    <Router>
        <div>
            <Route path="/" component={App}/>
            <Route path="/Home" component={Home}/>
            <Route path="/Customer" component={Customer}/>
        </div>
    </Router>
    ,
    document.getElementById('root')
);
registerServiceWorker();
```

This is required to provide route details

Route element will have path which provides url/ location details. The Route element will also have name of a component, which will be rendered when the corresponding path is matched

Demo – Routing (Cont…) - Explanation

Inside index.js

```
import React from 'react';
import ReactDOM from 'react-dom';
import './index.css';
import App from './App';
import registerServiceWorker from './registerServiceWorker';
import { BrowserRouter as Router,Route} from 'react-router-dom';
import Home from './Home';
import Customer from "./Customer";

ReactDOM.render(
    <Router>
        <div>
            <Route path="/" component={App}/>
            <Route path="/Home" component={Home}/>
            <Route path="/Customer" component={Customer}/>
        </div>
    </Router>
    ,
    document.getElementById('root')
);
registerServiceWorker();
```

Router element will contains all route details required for the application. Router element can have only one child element, here it is div element

Demo – Routing (Cont…) - Explanation

Create new component NavigationBar.js

```
import React,{Component} from 'react';
import {Link} from 'react-router-dom';
class NavigationBar extends Component{
    render(){
        return(
            <div>
                <div style={{'textAlign':'center','width':'100%',
                    'color':'green','backgroundColor':'pink','height':'60%'}}>
                    <ul>
                        <Link to="/Home">Home</Link> |
                        <Link to="/Customer">Customer</Link>
                    </ul>
                </div>
            </div>
        )
    }
}
export default NavigationBar;
```

This is required to provide view section to user, so that a user can click on whichever component he/she wants to navigate

Link element have “to” which matches the path provided in Route element. Link element also have text to display on the view which can be seen by the user and click on the link to navigate

Demo – Routing (Cont...)

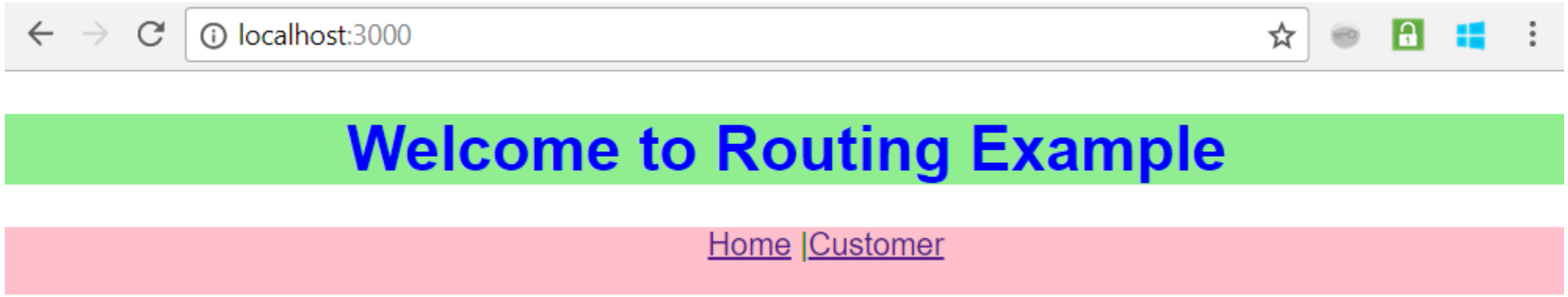
Inside App.js

```
import React, { Component } from 'react';
import NavigationBar from './NavigationBar';
import './App.css';
class App extends Component {
  render() {
    return (
        <div>
      <div style={{'textAlign':'center','width':'100%','color':'blue',
          'backgroundColor':'lightgreen'}}>
          <h1>Welcome to Routing Example</h1>
      </div>
        <NavigationBar/>
          </div>
    );
  }
}
export default App;
```

The NavigationBar component is rendered here

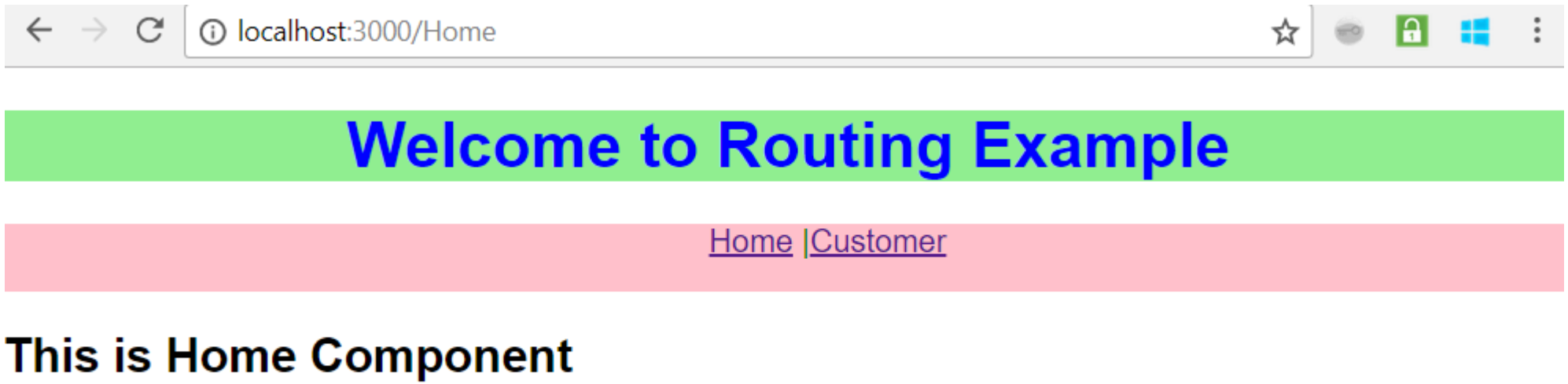
Demo – Routing (Cont...)

Output



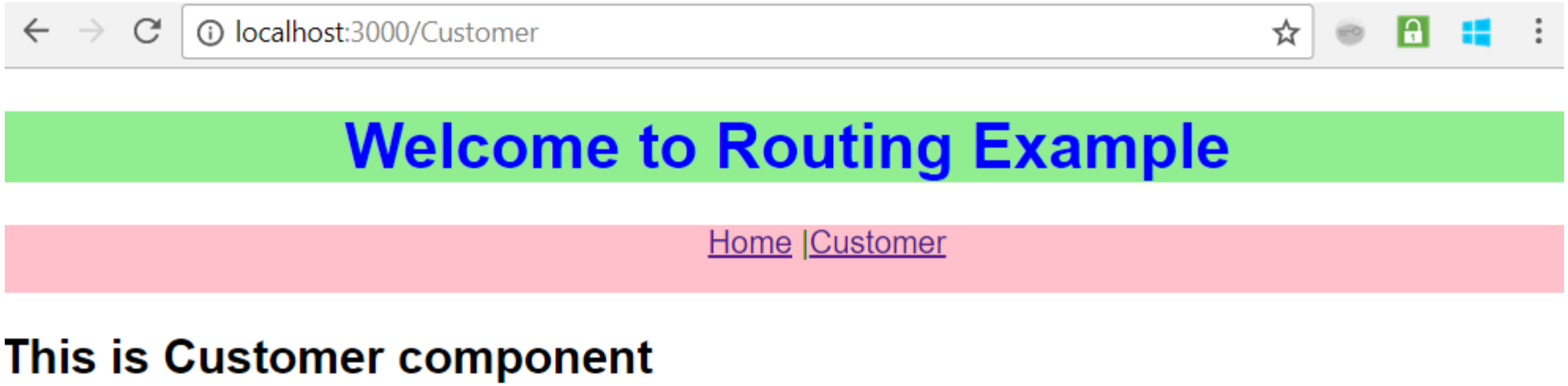
Demo – Routing (Cont...)

Output when clicked on Home Link



Demo – Routing (Cont...)

Output when clicked on Customer Link



Activity

1. Create a Product component which will have product details with Pid, Pname and Price and data as follows

Pid	Pname	Price
101	Mobile	12000
102	Laptop	25000
103	Fridge	9000

2. Create new route to this product component

3. Provide a link to product component

4. When the user clicks on the Product link the product component along with the product details should be displayed as follows
   - 101 →

Pid	Pname	Price
101	Mobile	12000

   - 102
   - 103

MODULE SUMMARY

- What is router
- Importance of router
- How to configure router

accenture

THANK YOU

accenture