MODUL 2: REACTJS

1. Tujuan

1. Student understand reactJS
2. Student understand to instalation ReactJS
3. Student understand to isntalation NodeJS

2. MATERIALS

Reactjs is a library created by Facebook to create a UI (user interface) on the Web (and also mobile) or often called FrontEnd

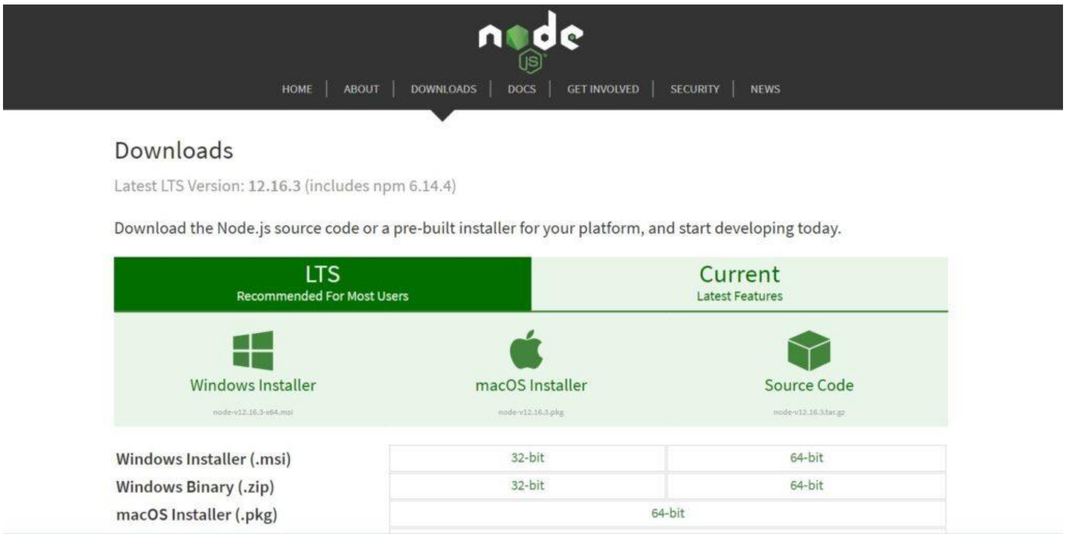
Many people call it a framework, but technically Reactjs is not a framework but a library for creating UI / web page views. Reactjs was originally created by Jordan Walke who is a Facebook employee. He released the first prototype Reactjs under the name "FaxJS" and was inspired by XHP. Facebook realizes, their (web) application is increasingly complex. On Facebook, there are many parts that need to be updated, such as news feeds, chat lists, chat boxes, etc. which are done in real time. If you use traditional methods like JQuery, this might cost a lot. Because manipulating the DOM requires large resources and Reactjs is here to provide a solution with VirutalDOM which they think is faster.

3. PRACTICE

A. Instalation NodeJS

Step 1 : Download Node.JS

The first step is that you open the browser, then enter this url https://nodejs.org/en/download/. You click the windows installer button to download the default version of node.js. As of this writing, version 12.16.3-x64 is the latest version. The Node.js installer includes NPM version 6.14.4.



Downloads

Latest LTS Version: 12.16.3 (includes npm 6.14.4)

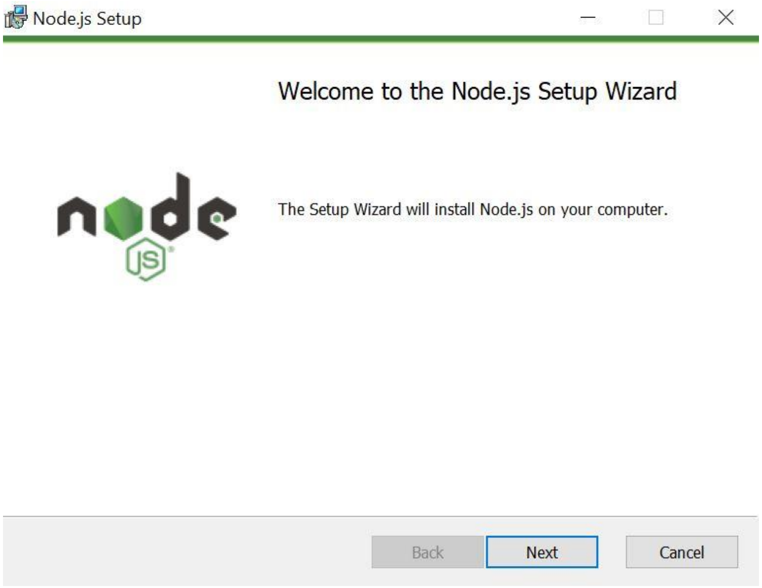
Download the Node.js source code or a pre-built installer for your platform, and start developing today.

Windows Installer (.msi)	32-bit	64-bit
Windows Binary (.zip)	32-bit	64-bit
macOS Installer (.pkg)	64-bit	

Note: On the website there is another version available which is newer. If you have an older system, you can probably go for the 32-bit version. You can also use switching from a stable LTS version to the current version. However, if you are new to Node.js or don't need a specific version, choose LTS

Step 2 : Install Node.JS and NPM

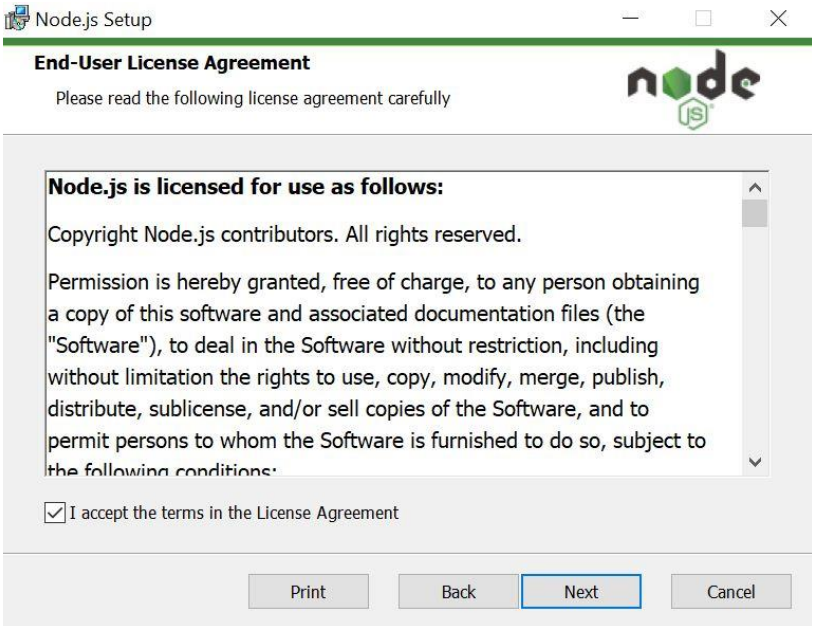
When finished downloading. Open the download link in your browser and click on the file. Or, browse to the location where you saved the file and double-click to start. The system will ask you if you want to run the software - click Run. You will be welcomed to the Node.js Setup Wizard - click Next.



Welcome to the Node.js Setup Wizard

The Setup Wizard will install Node.js on your computer.

On the next screen, review the license agreement. Click Next if you agree to the terms and install the software.



End-User License Agreement

Please read the following license agreement carefully

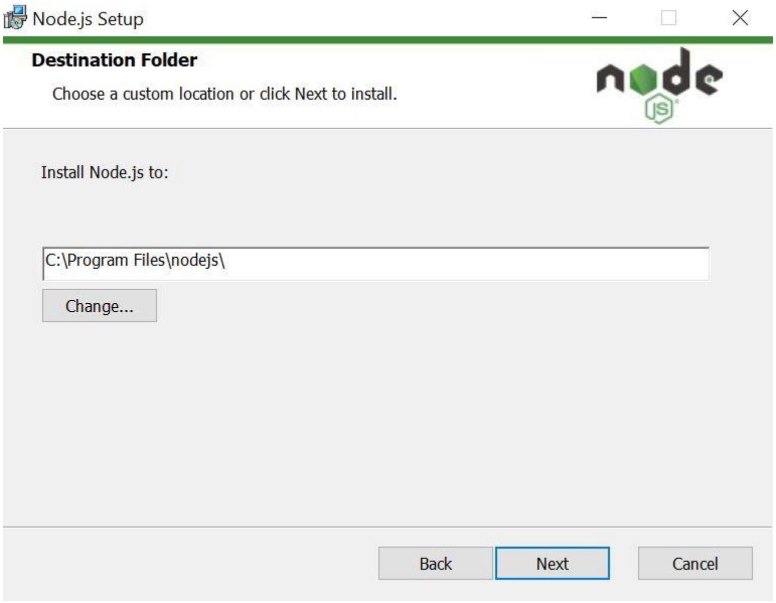
Node.js is licensed for use as follows:

Copyright Node.js contributors. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

☑ I accept the terms in the License Agreement

The installer will ask you for the installation location. Select the default location, unless you have a specific need to install it elsewhere - and then click Next.



Node.js Setup

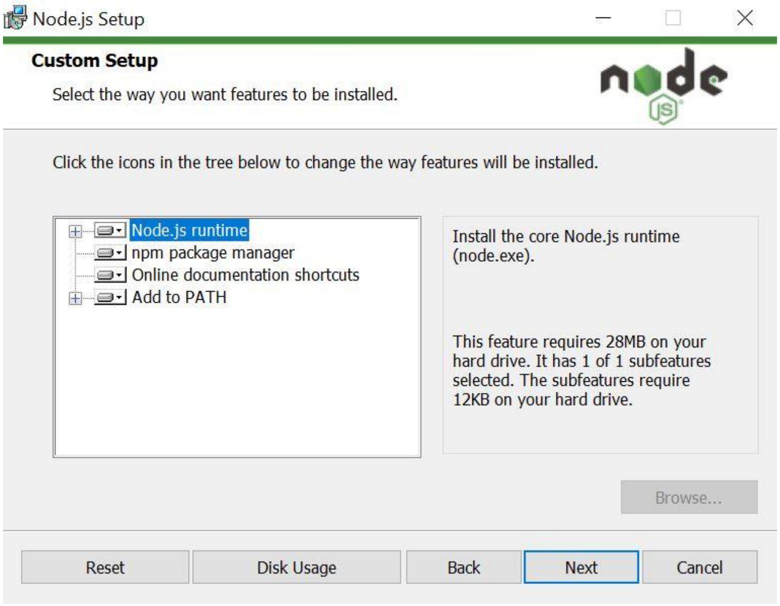
Destination Folder

Choose a custom location or click Next to install.

Install Node.js to:

C:\Program Files\nodejs\

The wizard will let you select components to include or remove from the installation. Select the default, unless you have special needs. Click Next.



Node.js Setup

Custom Setup

Select the way you want features to be installed.

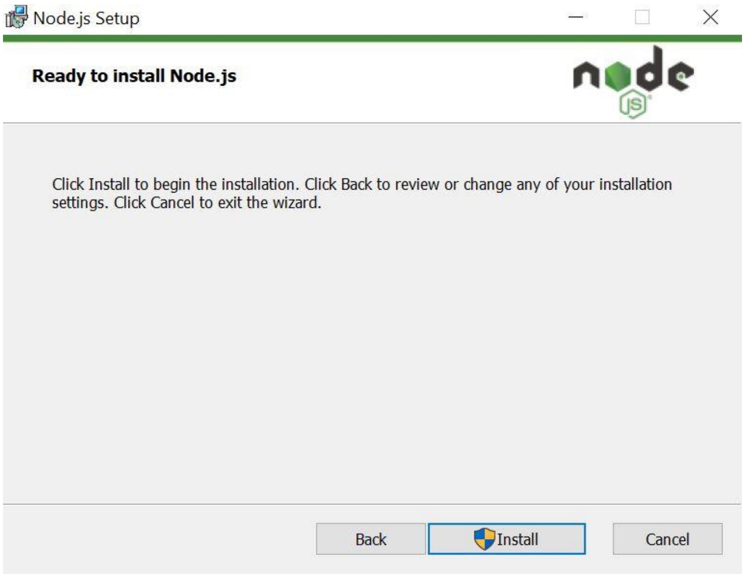
Click the icons in the tree below to change the way features will be installed.

- Node.js runtime
- npm package manager
- Online documentation shortcuts
- Add to PATH

Install the core Node.js runtime (node.exe).

This feature requires 28MB on your hard drive. It has 1 of 1 subfeatures selected. The subfeatures require 12KB on your hard drive.

Finally, click the Install button to run the installer. When done, click Finish



Node.js Setup

Ready to install Node.js

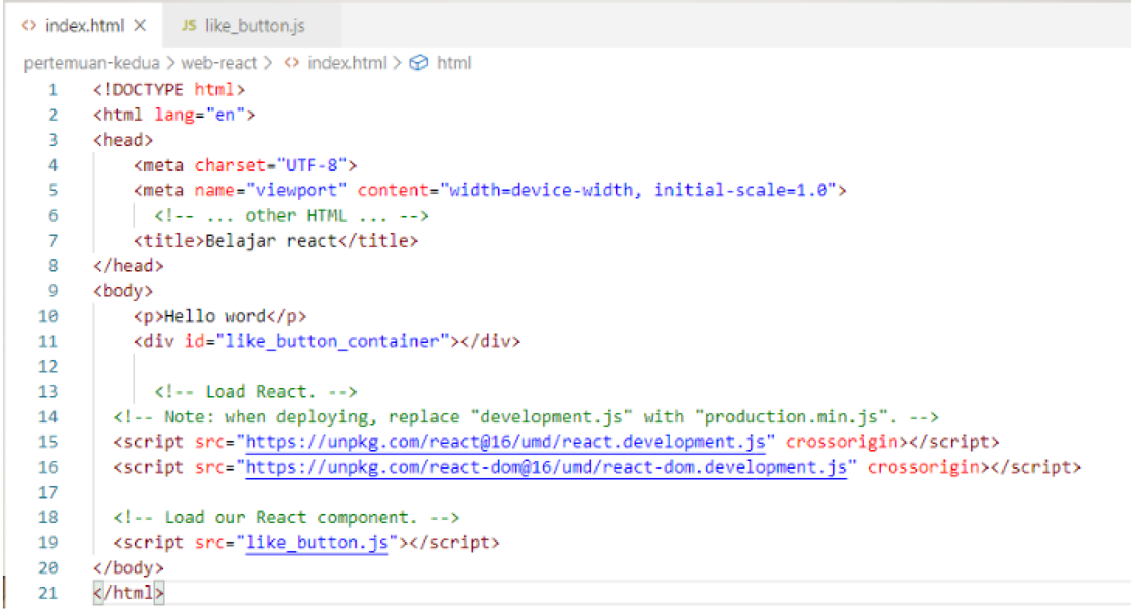
Click Install to begin the installation. Click Back to review or change any of your installation settings. Click Cancel to exit the wizard.

Step 3: Verify Installation
Open a command prompt or PowerShell, and enter the following command:
node –v
The system will display Node.js version installed on your system. You can do the same for NPM:
npm –v

B. Adding React to the website

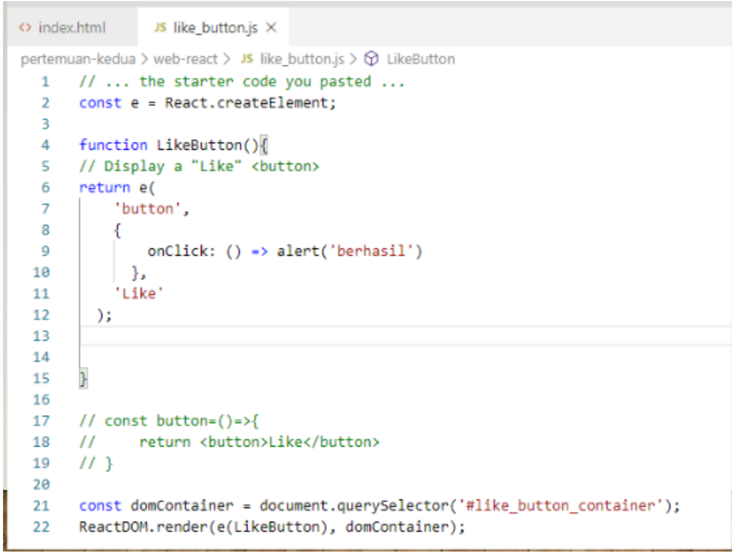
In addition to using the CLI we can also add react on the website
The steps are as follows

- Create a learn-react / second-meeting / web-react folder
- Create an index.html file



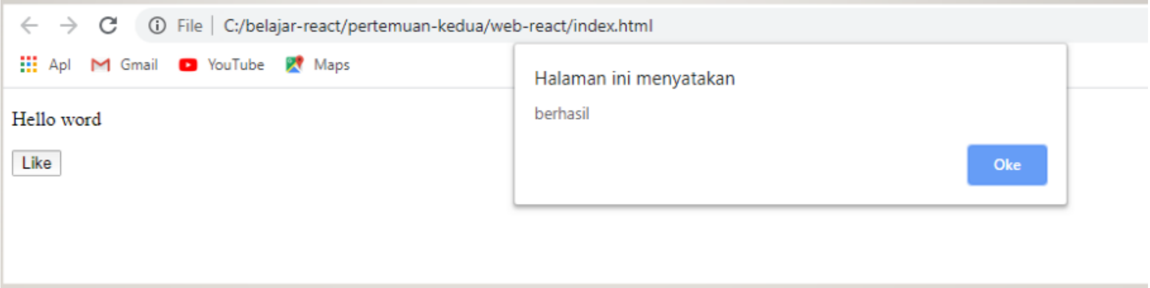
```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
      <!-- ... other HTML ... -->
    <title>Belajar react</title>
</head>
<body>
    <p>Hello word</p>
    <div id="like_button_container"></div>

      <!-- Load React. -->
  <!-- Note: when deploying, replace "development.js" with "production.min.js". -->
  <script src="https://unpkg.com/react@16/umd/react.development.js" crossorigin></script>
  <script src="https://unpkg.com/react-dom@16/umd/react-dom.development.js" crossorigin></script>

  <!-- Load our React component. -->
  <script src="like_button.js"></script>
</body>
</html>
```

After that create js file with the name like_button.js



```
// ... the starter code you pasted ...
const e = React.createElement;

function LikeButton(){
// Display a "Like" <button>
return e(
    'button',
    {
        onClick: () => alert('berhasil')
    },
    'Like'
  );


}

// const button=()=>{
//     return <button>Like</button>
// }

const domContainer = document.querySelector('#like_button_container');
ReactDOM.render(e(LikeButton), domContainer);
```

Result


File | C:/belajar-react/pertemuan-kedua/web-react/index.html
Apl
Gmail
YouTube
Maps

Hello word

Like

Halaman ini menyatakan

berhasil

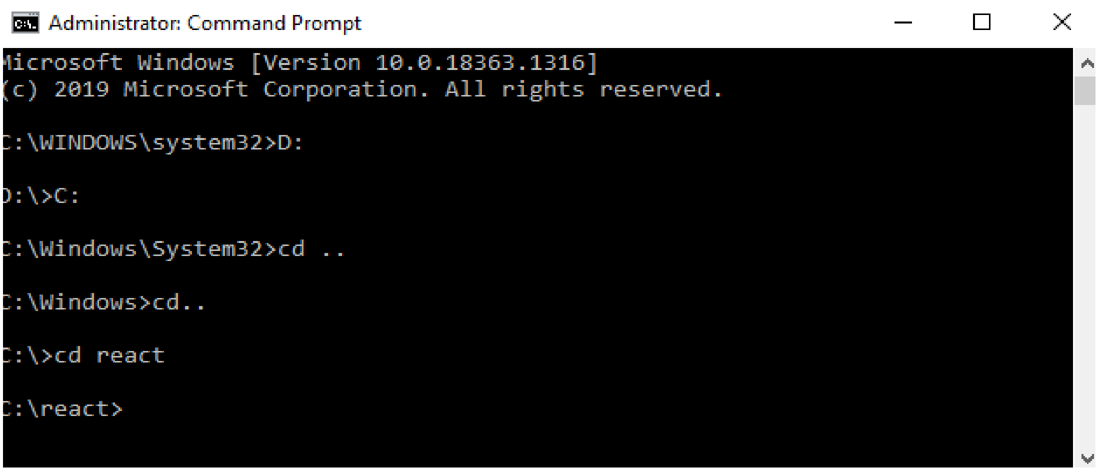
Oke

C. Instalasi react-app

create-react-app is the program used to create React projects. This program will generate all the things we need for the initial project. Starting from package.json, index.html file index.js and so on.

There are two ways to use it:
1. Installed first, then used it
2. Used directly with NPX

• specify the location for the react install, for example if in Windows C: react (create this folder if it doesn't exist yet) type in CMD / DOS



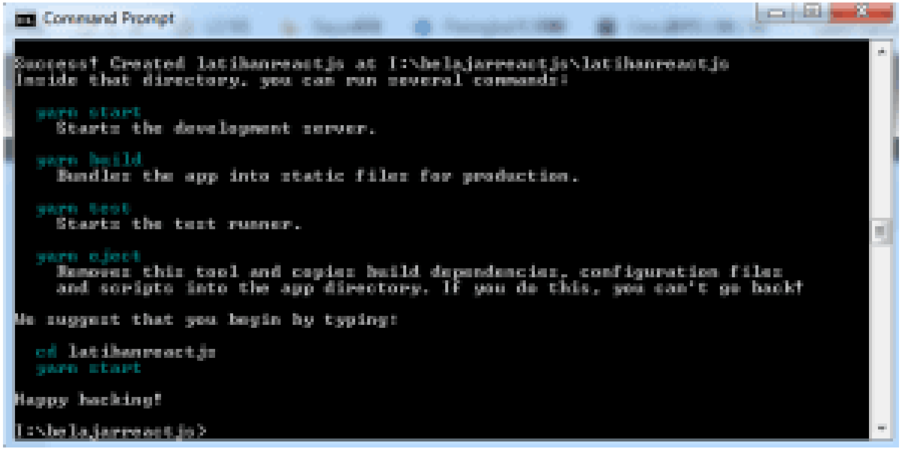
```
Microsoft Windows [Version 10.0.18363.1316]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\WINDOWS\system32>D:

D:\>C:

C:\Windows\System32>cd ..

C:\Windows>cd..

C:\>cd react

C:\react>
```

If the position is already in the folder, type the command

```
npm install -g create-react-app
```

- to check the success / no, typing the command:

```
create-react-app --version
```

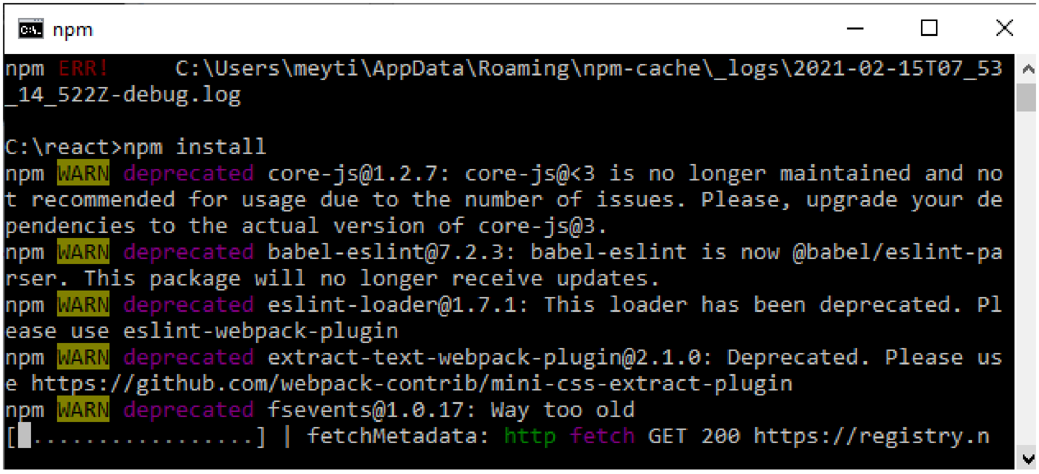


```
Success! Created latihanreactjs at I:\belajarreactjs\latihanreactjs
Inside that directory, you can run several commands:

  yarn start
    Starts the development server.

  yarn build
    Bundles the app into static files for production.

  yarn test
    Starts the test runner.

  yarn eject
    Removes this tool and copies build dependencies, configuration files
    and scripts into the app directory. If you do this, you can't go back!

We suggest that you begin by typing:

  cd latihanreactjs
  yarn start

Happy hacking!

I:\belajarreactjs>
```

```
create-react-app web-reactku
cd web-reactku
npm start
```

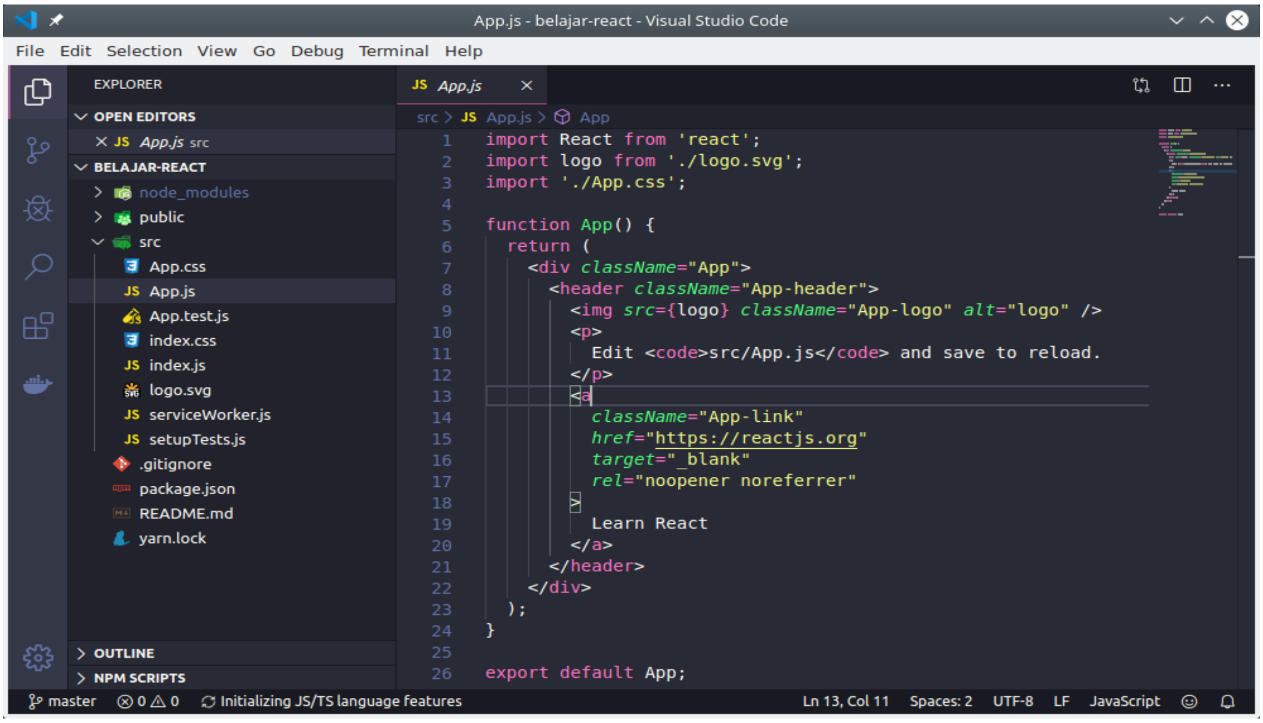
wait a minute, will automatically open the web page with the address localhost: 3000 if there is an error type the command

```
npm install
```



```
npm ERR!     C:\Users\meyti\AppData\Roaming\npm-cache\_logs\2021-02-15T07_53
_14_522Z-debug.log

C:\react>npm install
npm WARN deprecated core-js@1.2.7: core-js@<3 is no longer maintained and no
t recommended for usage due to the number of issues. Please, upgrade your de
pendencies to the actual version of core-js@3.
npm WARN deprecated babel-eslint@7.2.3: babel-eslint is now @babel/eslint-pa
rser. This package will no longer receive updates.
npm WARN deprecated eslint-loader@1.7.1: This loader has been deprecated. Pl
ease use eslint-webpack-plugin
npm WARN deprecated extract-text-webpack-plugin@2.1.0: Deprecated. Please us
e https://github.com/webpack-contrib/mini-css-extract-plugin
npm WARN deprecated fsevents@1.0.17: Way too old
[█.................] | fetchMetadata: http fetch GET 200 https://registry.n
```

- If the yarn is already installed on your computer, create-react-app defaults to using it. Nodejs package manager yarn is made in Facebook the same function as NPM. But if there is no yarn, it will use npm.
- If the project has been completed, there will be a new folder called learning-react to the folder where the command was executed.
- Now try opening the folder with VS Code. Click the File menu-> Open Folder, then find the folder.



```
import React from 'react';
import logo from './logo.svg';
import './App.css';

function App() {
  return (
    <div className="App">
      <header className="App-header">
        <img src={logo} className="App-logo" alt="logo" />
        <p>
          Edit <code>src/App.js</code> and save to reload.
        </p>
        <a
          className="App-link"
          href="https://reactjs.org"
          target="_blank"
          rel="noopener noreferrer"
        >
          Learn React
        </a>
      </header>
    </div>
  );
}

export default App;
```

This is the directory structure of the project React.

📁 node_modules contains Nodejs module packages;

📁 public contains files for the public such as HTML, CSS, icons, and images;

📄 index.html is the HTML file that the React application will use to render components

📁 src contains the code from the Reactjs application, this is where we will build the component;

📄 App.js contains code for the App component or the core component of the application;

📄 App.test.js contains code for testing App components;

📄 index.js contains code for rendering the App component to the Real DOM;

📄 serviceWorker.js contains code for service worker, this we need later when we build PWA (Progressive Web Apps) application;

📄 setTests.js contains code for application testing.

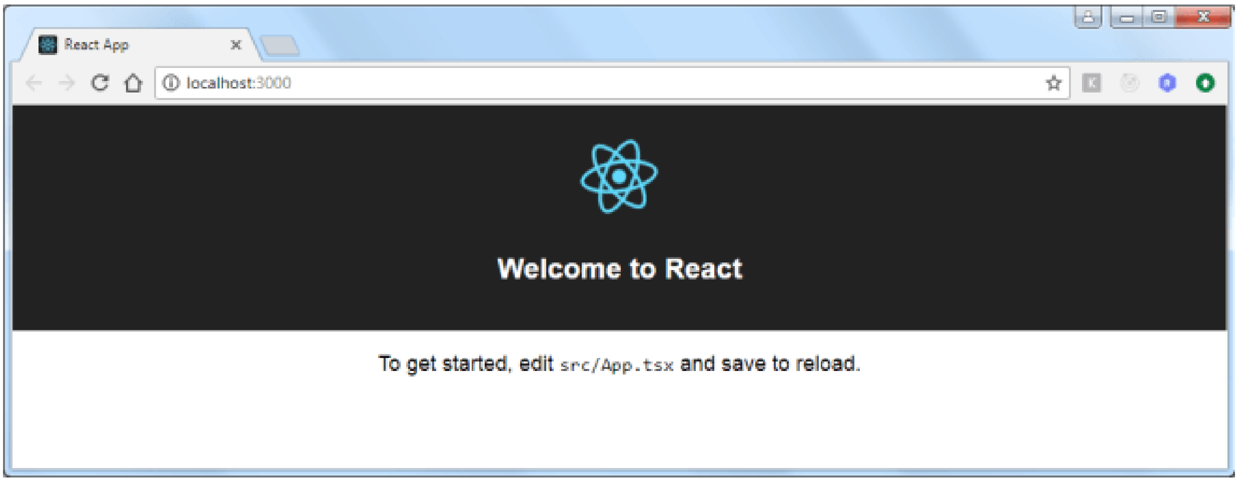
📄 .gitignore contains code that Git will ignore.

📄 package.json JSON file which contains a description of the project and a list of modules needed. • 📄 yarn.lock is the file that Yarn uses to lock the versions of the Nodejs module being used. To run type:

To Start

```
npm start
```

Result



Welcome to React

To get started, edit `src/App.tsx` and save to reload.

D. Project React Structure

First react will run on the src folder index.js

On line 7

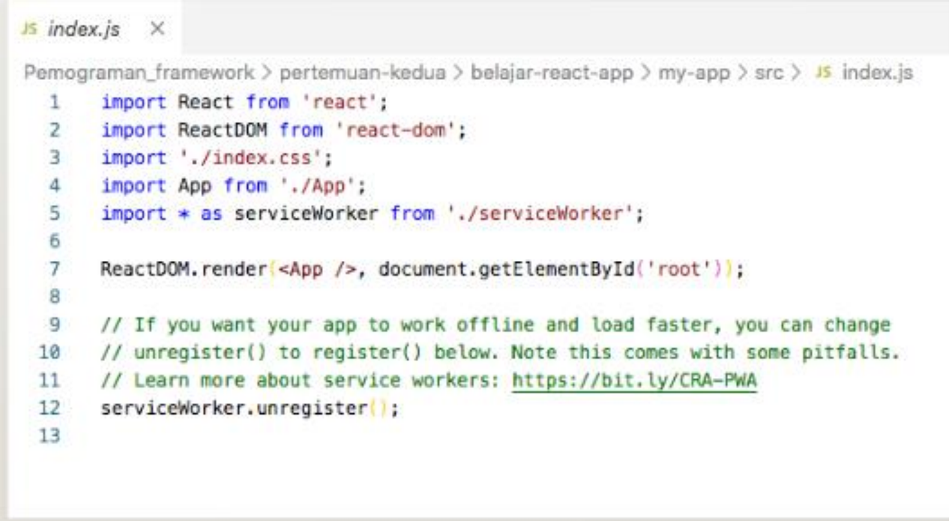
- There is a code rendering process in the DOM
- There are two rendering parameters

ReactDOM.render (element, container [, callback])

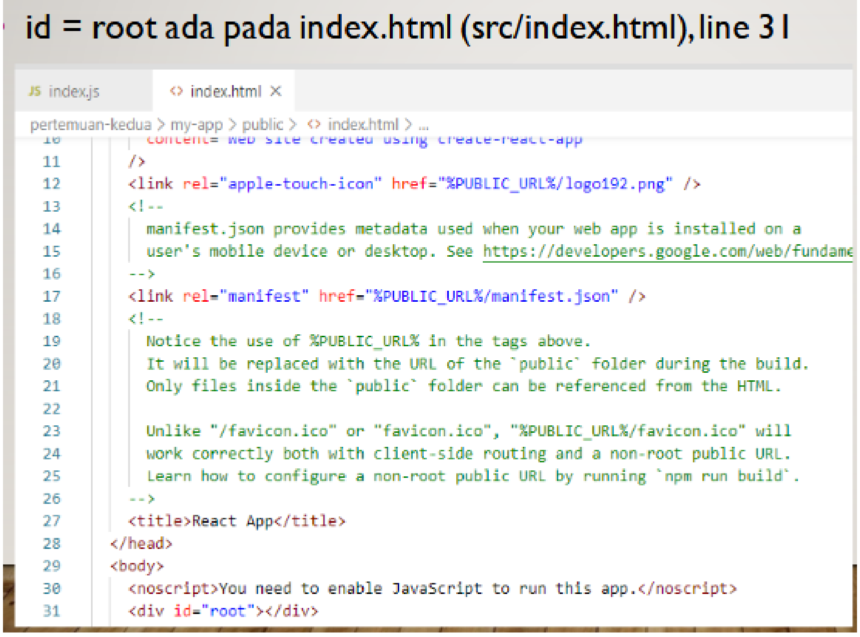
- https://reactjs.org/docs/react-dom.html#render

So on line 7 there are 2 to be rendered

- Component App (in src / App.js)
- Id = root in index.html (in public / index.html)


index.js
Pemograman_framework > pertemuan-kedua > belajar-react-app > my-app > src > index.js

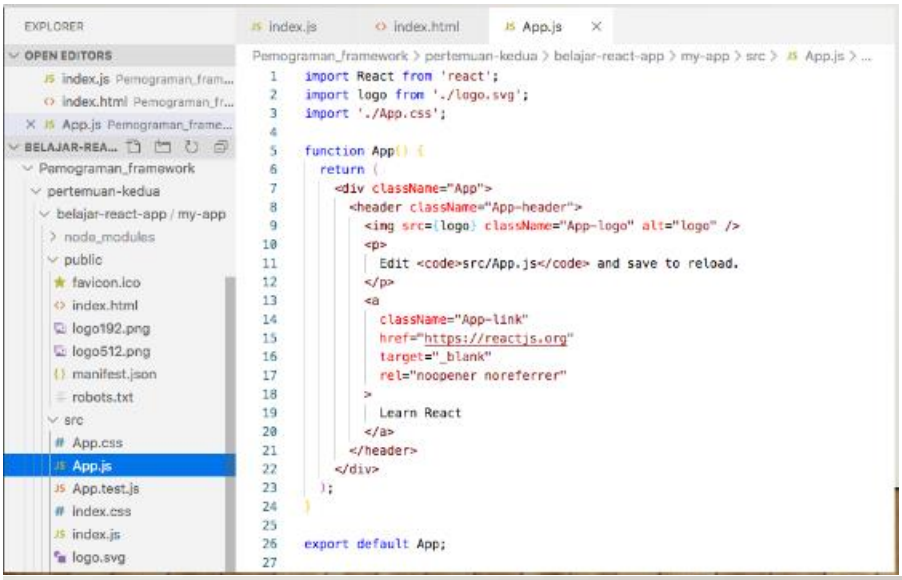
```
1   import React from 'react';
2   import ReactDOM from 'react-dom';
3   import './index.css';
4   import App from './App';
5   import * as serviceWorker from './serviceWorker';
6
7   ReactDOM.render(<App />, document.getElementById('root'));
8
9   // If you want your app to work offline and load faster, you can change
10  // unregister() to register() below. Note this comes with some pitfalls.
11  // Learn more about service workers: https://bit.ly/CRA-PWA
12  serviceWorker.unregister();
13
```



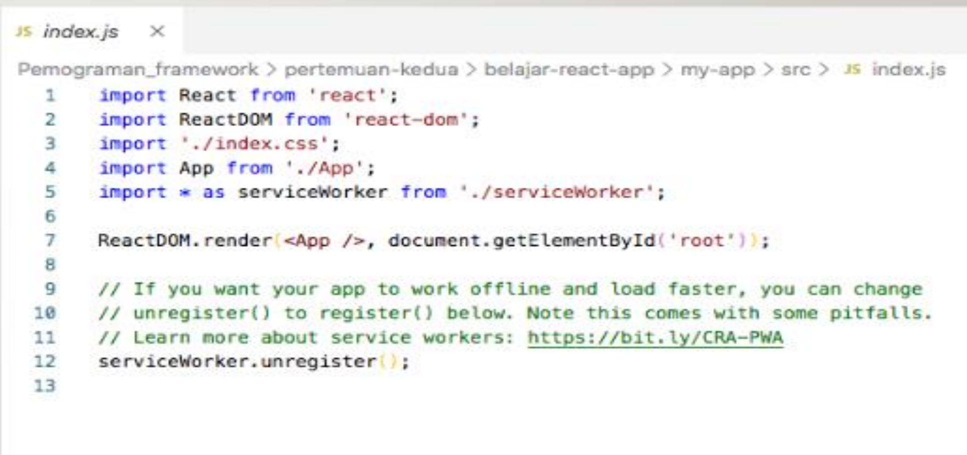
id = root ada pada index.html (src/index.html), line 31

```
10        content="Web site created using create-react-app
11      />
12      <link rel="apple-touch-icon" href="%PUBLIC_URL%/logo192.png" />
13      <!--
14        manifest.json provides metadata used when your web app is installed on a
15        user's mobile device or desktop. See https://developers.google.com/web/fundame
16      -->
17      <link rel="manifest" href="%PUBLIC_URL%/manifest.json" />
18      <!--
19        Notice the use of %PUBLIC_URL% in the tags above.
20        It will be replaced with the URL of the `public` folder during the build.
21        Only files inside the `public` folder can be referenced from the HTML.
22
23        Unlike "/favicon.ico" or "favicon.ico", "%PUBLIC_URL%/favicon.ico" will
24        work correctly both with client-side routing and a non-root public URL.
25        Learn how to configure a non-root public URL by running `npm run build`.
26      -->
27      <title>React App</title>
28    </head>
29    <body>
30      <noscript>You need to enable JavaScript to run this app.</noscript>
31      <div id="root"></div>
```

Component / App functions in src / App.js, then index.js on line 4 of its importation App



```
import React from 'react';
import logo from './logo.svg';
import './App.css';

function App() {
  return (
    <div className="App">
      <header className="App-header">
        <img src={logo} className="App-logo" alt="logo" />
        <p>
          Edit <code>src/App.js</code> and save to reload.
        </p>
        <a
          className="App-link"
          href="https://reactjs.org"
          target="_blank"
          rel="noopener noreferrer"
        >
          Learn React
        </a>
      </header>
    </div>
  );
}

export default App;
```

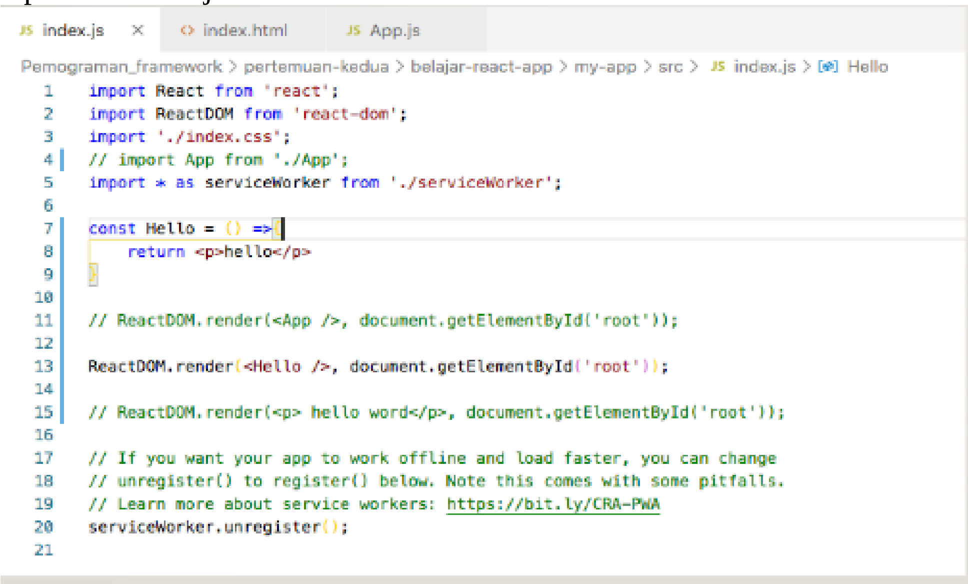


index.js
Pemograman_framework > pertemuan-kedua > belajar-react-app > my-app > src > index.js

```
1   import React from 'react';
2   import ReactDOM from 'react-dom';
3   import './index.css';
4   import App from './App';
5   import * as serviceWorker from './serviceWorker';
6
7   ReactDOM.render(<App />, document.getElementById('root'));
8
9   // If you want your app to work offline and load faster, you can change
10  // unregister() to register() below. Note this comes with some pitfalls.
11  // Learn more about service workers: https://bit.ly/CRA-PWA
12  serviceWorker.unregister();
13
```

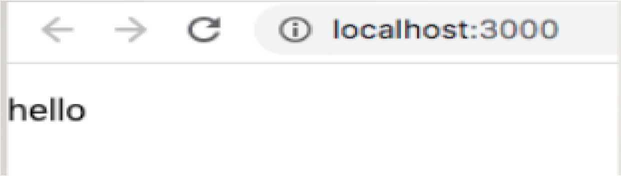
MAKE HELLO COMPONENTS USING ARROW FUNCTION

Open the index.js file in src and add an arrow function called Hello


index.js
index.html
App.js
Pemograman_framework > pertemuan-kedua > belajar-react-app > my-app > src > index.js > Hello

```
import React from 'react';
import ReactDOM from 'react-dom';
import './index.css';
// import App from './App';
import * as serviceWorker from './serviceWorker';

const Hello = () =>{
    return <p>hello</p>
}

// ReactDOM.render(<App />, document.getElementById('root'));

ReactDOM.render(<Hello />, document.getElementById('root'));

// ReactDOM.render(<p> hello word</p>, document.getElementById('root'));

// If you want your app to work offline and load faster, you can change
// unregister() to register() below. Note this comes with some pitfalls.
// Learn more about service workers: https://bit.ly/CRA-PWA
serviceWorker.unregister();

```

The result will be as follow



hello

Collect the results of the source code and a screenshot for every step practicum collected later than one week after the lab on github and in the LMS.

***** Keep Spirit *****