|  |
| --- |
| React 实践 |

# 快速建项目

创建出一个能够运行的项目  
1、npx create-react-app my-app npm start

2、 npm init react-app my-app

3、 yarn create react-app my-app (*yarn create 在 Yarn 0.25+*)



项目文件：

my-app

├── README.md 文档说明书

├── node\_modules node包

├── package.json npm依赖包

├── .gitignore Git哪些文件不需要添加到版本管理中

├── public 静态资源

│ ├── favicon.ico

│ ├── index.html

│ └── manifest.json

└── src 源码-开发代码

├── App.css

├── App.js 根组件

├── App.test.js

├── index.css 全局样式

├── index.js 入口文件

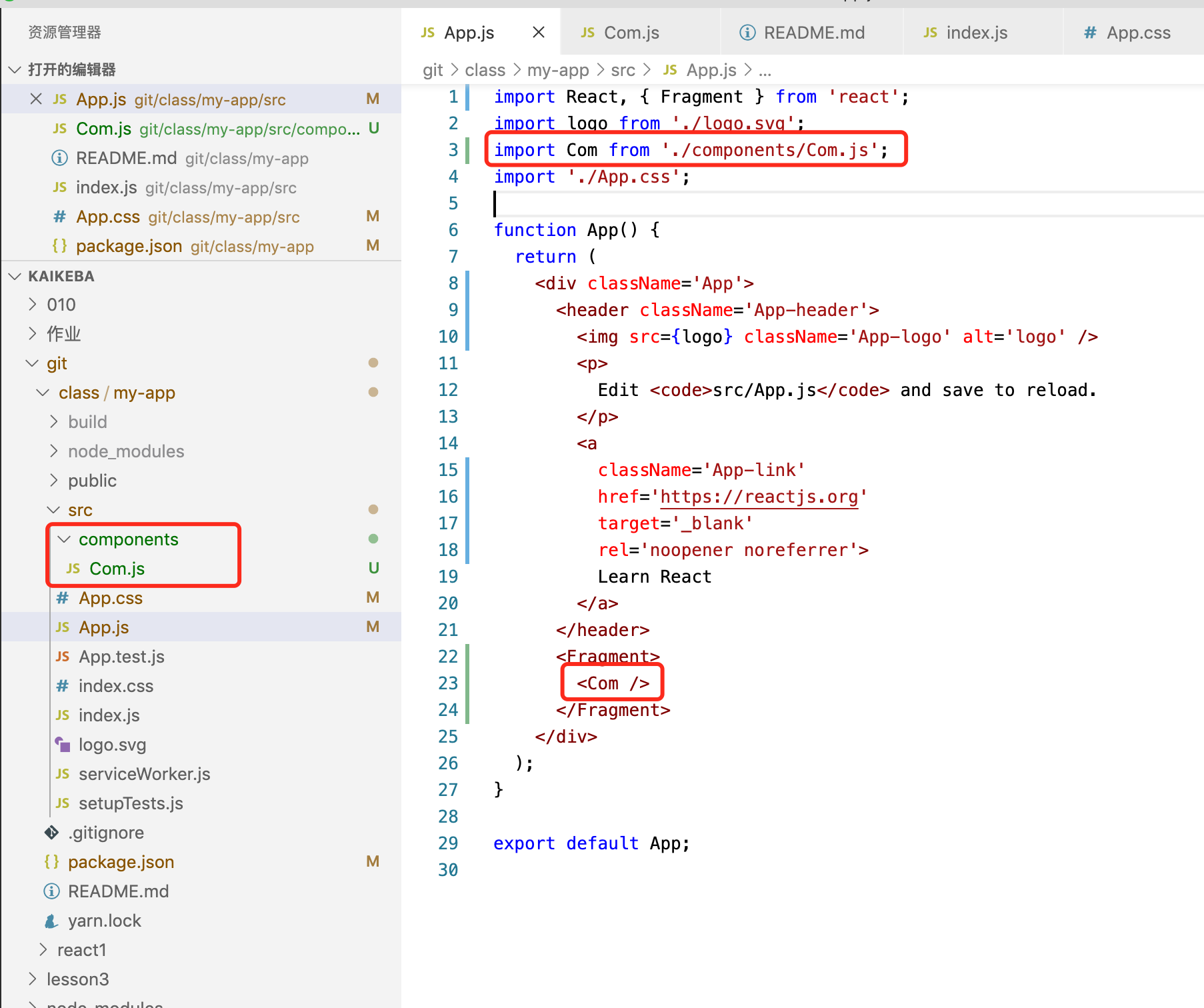
├── logo.svg

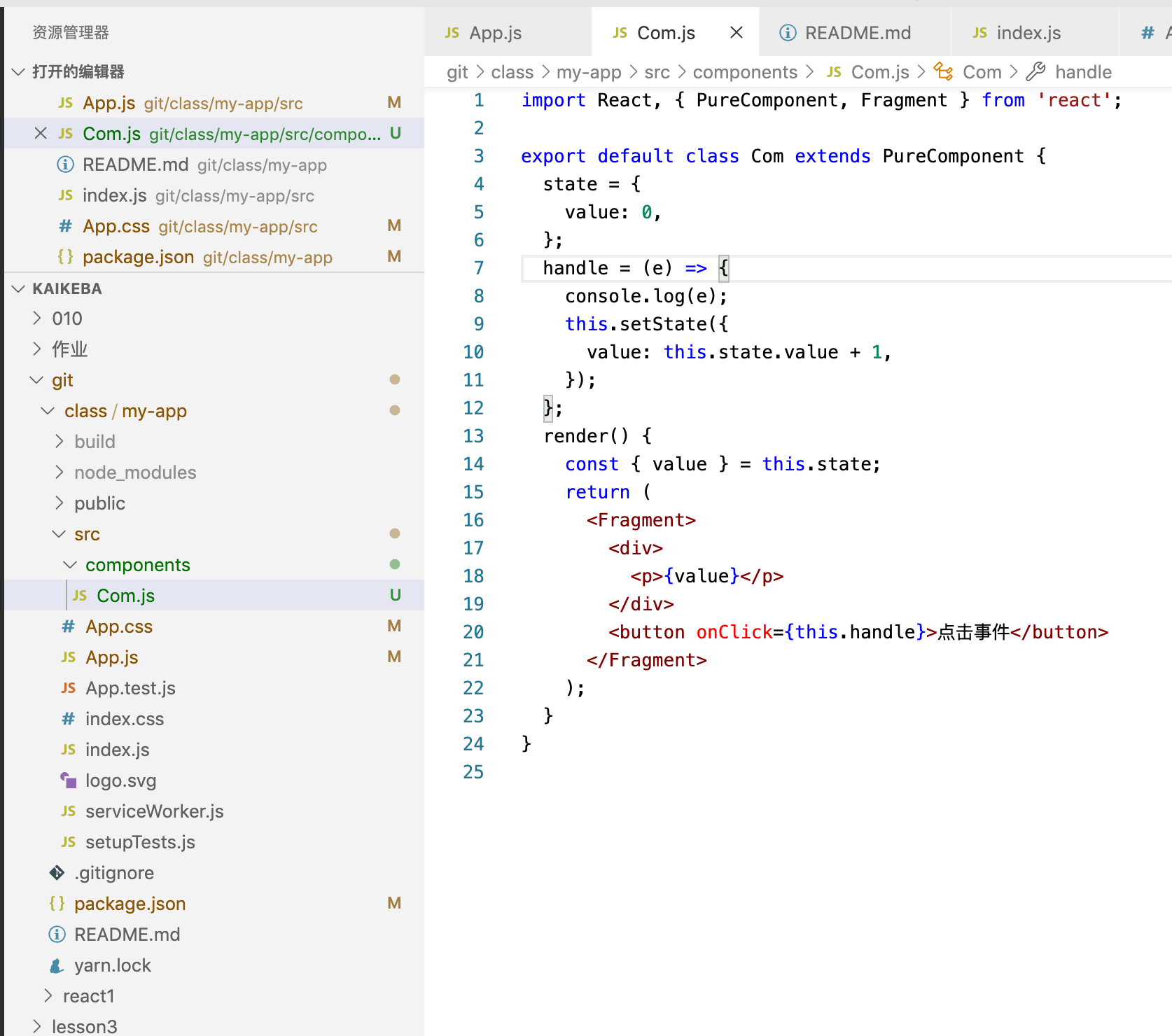
└── serviceWorker.js pwa支持

启动项目： yarn start(可以自已配置)

如何打包： Yarn build

# 创建和引用组建+事件运用





# React and ReactDom

1. 删除src下⾯面所有代码，新建index.js

import React from 'react';

import ReactDOM from 'react-dom';

//这⾥里里怎么没有出现React字眼? // JSX => React.createElement(...) ReactDOM.render(<h1>Hello React</h1>, document.querySelector('#root'));

1. React负责逻辑控制，数据 -> VDOM

ReactDom渲染实际DOM，VDOM -> DOM

React使⽤用JSX来描述UI

babel-loader把JSX 编译成相应的 JS 对象，React.createElement再把这个JS对象构造成React需 要的虚拟dom。

# JSX

<https://zh-hans.reactjs.org/docs/introducing-jsx.html>

JSX是⼀一种JavaScript的语法扩展，其格式⽐比较像模版语⾔言，但事实上完全是在JavaScript内部实现的。 JSX可以很好地描述UI，能够有效提⾼高开发效率

# Webpack.comfing.js

entry: [ // WebpackDevServer客户端，它实现开发时热更更新功能 isEnvDevelopment && require.resolve('react-dev-utils/webpackHotDevClient'), // 应⽤用程序⼊入⼝口:src/index paths.appIndexJs, ].filter(Boolean),

webpack.config.js 是webpack配置⽂文件，开头的常量量声明可以看出cra能够⽀支持ts、sass及css模块化。

//Check if TypeScript is setup

const useTypeScript = fs.existsSync(paths.appTsConfig);

//style files regexes

const cssRegex = /\.css$/;

const cssModuleRegex = /\.module\.css$/;

const sassRegex = /\.(scss|sass)$/;

const sassModuleRegex = /\.module\.(scss|sass)$/;