

## REACT

### #1 - Indecision APP

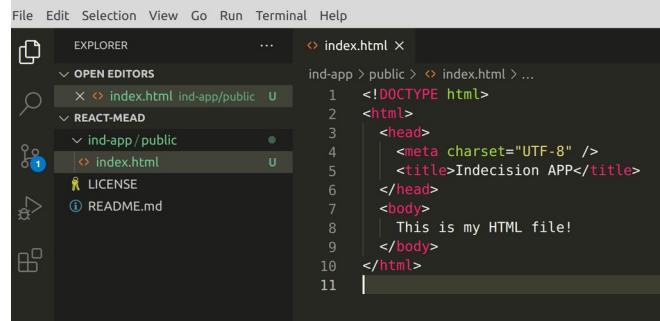
## Table of Contents

3 Start with REACT.....	3
3.1 Basic setup (before REACT) 007.....	3
3.2 Create a React APP (scripts) 008.....	4

## 3 Start with REACT

### 3.1 Basic setup (before REACT) 007

Create a new repository on GitHub. Clone it to a local folder. Create a simple HTML file.



The screenshot shows the Visual Studio Code interface. The Explorer sidebar on the left lists a project structure: 'REACT-MEAD' with a 'public' folder containing 'index.html'. The 'index.html' file is open in the main editor area. The code content is:

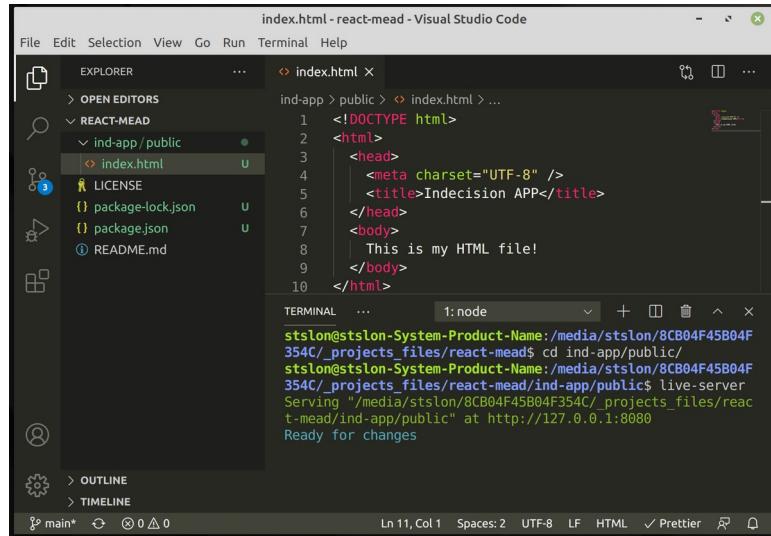
```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8" />
<title>Indecision APP</title>
</head>
<body>
    This is my HTML file!
</body>
</html>
```

Install yarn [CLI=> **npm i -g yarn**]

Install live server [CLI=> **sudo npm install -g live-server**]

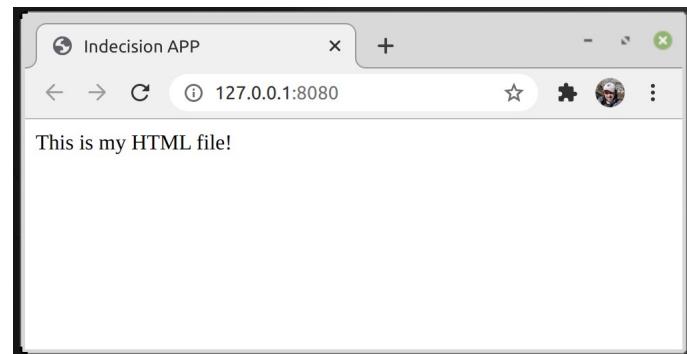
Go to a folder ind-app/public [CLI=> **cd ind-app/public**]

Start a server [CLI=> **live-server**]



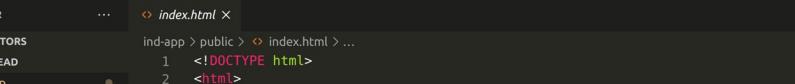
The screenshot shows the Visual Studio Code interface with the 'REACT-MEAD' project loaded. The terminal at the bottom shows the command 'live-server' being run and its output:

```
stslon@stslon-System-Product-Name:/media/stslon/8CB04F45B04F354C/_projects_files/react-mead$ cd ind-app/public/
stslon@stslon-System-Product-Name:/media/stslon/8CB04F45B04F354C/_projects_files/react-mead/ind-app/public$ live-server
Serving "/media/stslon/8CB04F45B04F354C/_projects_files/react-mead/ind-app/public" at http://127.0.0.1:8080
Ready for changes
```



## 3.2 Create a React APP (scripts) 008

Add div 'app' and 3 scripts to index.html and create **public/scripts/app.js**



The screenshot shows a code editor interface with the following details:

- File Bar:** File, Edit, Selection, View, Go, Run, Terminal, Help
- Sidebar:** EXPLORER, OPEN EDITORS, REACT-MEAD, ind-app, public, scripts, index.html (selected), LICENSE, package-lock.json, package.json, README.md.
- Content Area:** The file `index.html` is open, showing the following code:

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8" />
    <title>Indecision APP</title>
  </head>
  <body>
    <div id="app"></div>
    <script src="http://unpkg.com/react@16.0.0/umd/react.development.js"></script>
    <script src="http://unpkg.com/react-dom@16.0.0/umd/react-dom.development.js"></script>
    <script src="/scripts/app.js"></script>
  </body>
</html>
```

JSX is a JavaScript XML. BABEL is a JavaScript compiler (it convert modern simple ES6 or ES7 to ES5)

The screenshot shows a browser window with the title "Babel - The compiler for next...". The address bar contains the URL "browsers=defaults%2C%20not%20ie%2011%2C%20not%20i...". The page content displays a snippet of JSX code:

```
1 const template = <h1 id='someID'>This  
is JSX from app.js</h1>
```

Below this, a code editor window titled "app.js" shows the same code with Babel annotations:

```
1 "use strict";  
2  
3 const template =  
4 /#_PURE_*/React.createElement("h1",  
5 { id: "someID" }, "This is JSX from app.js");
```

The code editor has syntax highlighting for JavaScript and JSX. To the right of the code editor is a file tree and a preview pane. The file tree shows the project structure:

- EXPLORER
- REACT-MEAD
- ind-app
- \_description
- public
- scripts
- JS app.js
- index.html

The "app.js" file is selected in the file tree. The preview pane shows the rendered output of the JSX code:

```
1 console.log('App.js is running ...');  
2 //JSX - JavaScript XML  
3 const template = React.createElement(  
4 'h1',  
5 {  
6 id: 'someID',  
7 },  
8 'This is JSX from app.js'  
9 );  
10  
11 let appRoot = document.getElementById('app');  
12  
13 ReactDOM.render(template, appRoot);
```

