# parcel 与 React 搭建项目

[parcel](https://github.com/parcel-bundler/parcel) 支持热加载、source map、code split (需要按既定格式书写代码).

## 基础

1. 初始化项目

* npm init -y

1. 安装资源依赖: react 相关依赖

* npm i --save react react-dom

1. 安装工具依赖: parcel

* npm i --save-dev parcel-bundler

1. 启动命令: npm run start

* "scripts": {  
   "start": "parcel src/index.html"  
  }

src/index.js

import React from "react";  
import ReactDOM from "react-dom";  
  
const App = () => {  
 return <h1>Hello World! HOT</h1>;  
};  
  
ReactDOM.render(<App />, document.getElementById("root"));

src/index.html

<!DOCTYPE html>  
<html lang="zh">  
<head>  
 <meta charset="UTF-8">  
 <meta name="viewport" content="width=device-width, initial-scale=1.0">  
 <meta http-equiv="X-UA-Compatible" content="ie=edge">  
 <title>APP</title>  
</head>  
<body>  
 <div id="root"></div>  
 <script src="./index.js"></script>  
</body>  
</html>

# Visual Studio Code

1. 主页: <https://code.visualstudio.com/>
2. 快捷键: <https://code.visualstudio.com/shortcuts/keyboard-shortcuts-windows.pdf>

## 基本用法

1. 命令搜索工具: F1.
2. 扩展安装: ext install extensionname

## Vue.js 配置

### 代码高亮

1. 安装 Vetur 扩展.

### ESLint 问题自动修复

{  
 "eslint.enable": true,  
 "eslint.options": {  
 "extensions": [  
 ".html",  
 ".js",  
 ".vue"  
 ]  
 },  
 "eslint.validate": [{  
 "language": "html",  
 "autoFix": true  
 },  
 {  
 "language": "vue",  
 "autoFix": true  
 },  
 {  
 "language": "javascript",  
 "autoFix": true  
 }  
 ]  
}

# Vue-CLI

1. GitHub: <https://github.com/vuejs/vue-cli>

## 代理

配置代理可解决跨域问题, 需要服务端配置跨域.

// config/index.js  
proxyTable: {  
 '/api': {  
 target: 'http://stage.xxxx.com',  
 changeOrigin: true,  
 pathRewrite: {  
 '^/api': '/api'  
 }  
 }  
}