

全部开发者教程

09: 引入代码格式化工具：prettier 让你的代码结构更加规范

10: 模块打包器：rollup

11: 初见框架雏形：配置路径映射

12: 总结

第四章：响应系统 - 响应系统的核心设计原则

01: 前言

02: JS 的程序性

03: 如何让你的程序变得更加“聪明”？

04: vue 2 的响应性核心 API：Object.defineProperty

05: Object.defineProperty 在设计层的缺陷

06: vue3的响应性核心 API：proxy

Sunday • 更新于 2022-10-19

10: 模块打包器：rollup

rollup 是一个模块打包器，和 webpack 一样可以将 JavaScript 打包为指定的模块。

但是不同的是，对于 webpack 而言，它在打包的时候会产生许多 冗余的代码，这样的一种情况在我们开发大型项目的时候没有什么影响，但是如果我们是开发一个 库 的时候，那么这些冗余的代码就会大大增加库体积，这就不好美好了。

所以说我们需要一个 小而美 的模块打包器，这就是 rollup：

Rollup 是一个 JavaScript 模块打包器，可以将小块代码编译成大块复杂的代码，例如 library 或应用程序。

因为 rollup 并不是咱们课程的重点，所以我们不会花费大量的篇幅来讲解 rollup 的概念，只会讲解 rollup 的一些基础概念，能够满足我们当前的使用即可。大家可以把 rollup 理解为一个打包 库 的模块打包器，而在应用程序的打包中选择 webpack。

rollup

我们可以在项目根目录下，创建 rollup.config.js 文件作为 rollup 的配置文件（就像 webpack.config.js 一样）：

```
<> 代码块
1   import resolve from '@rollup/plugin-node-resolve'
2   import commonjs from '@rollup/plugin-commonjs'
3   import typescript from '@rollup/plugin-typescript'
4
5   /**
6    * 默认导出一个数组，数组的每一个对象都是一个单独的导出文件配置，详细可查：https://www.rollupjs.com/guide/#configuration
7    */
8   export default [
9     {
10       // 入口文件
11       input: 'packages/vue/src/index.ts',
12       // 打包出口
13       output: [
14         // 导出 iife 模式的包
15         {
16           // 开启 SourceMap
17           sourcemap: true,
18           // 导出的文件地址
19           file: './packages/vue/dist/vue.js',
20           // 生成的包格式：一个自动执行的功能，适合作为<script>标签
21           format: 'iife',
22           // 变量名
23           name: 'Vue'
24         }
25       ],
26       // 插件
27       plugins: [
28         // ts 支持
29         typescript({
30           sourceMap: true
```

意见反馈收藏教程标记书签

索引目录

10: 模块打包器：rollup

```
32         // 模块导入的路径补全
33         resolve(),
34         // 将 CommonJS 模块转换为 ES2015
35         commonjs()
36     ]
37 }
38 ]
```

依赖包的详细版本为：

<> 代码块

```
1  "devDependencies": {
2    "@rollup/plugin-commonjs": "^22.0.1",
3    "@rollup/plugin-node-resolve": "^13.3.0",
4    "@rollup/plugin-typescript": "^8.3.4"
5  }
```

那么至此我们就配置了一个基本的 rollup 的配置文件

然后我们可以在 input 路径下创建对应的 index.ts，并写入初始代码：

<> 代码块

```
1  console.log('hello vue-next-mini')
```

同时因为我们使用的是 ts，所以还需要安装：tslib typescript：

<> 代码块

```
1  npm i --save-dev tslib@2.4.0 typescript@4.7.4
```

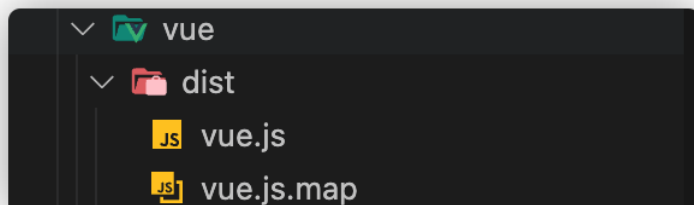
那么至此，所有的配置完成。

此时我们可以在 package.json 中新增一个 scripts：

<> 代码块

```
1  "build": "rollup -c"
```

执行 npm run build，即可打包成功。



09: 引入代码格式化工具：prettier 让你的... < 上一节 下一节 > 11: 初见框架雏形：配置路径映射

✎ 我要提出意见反馈