

20.打包优化和 dist 服务

学习要点：

- 1.打包优化
- 2.dist 服务

本节课我们来开始学习一下了解下打包优化和 dist 服务器的方法。

一. 打包优化

1. webpack 提供了一个选项: `optimiztion` 来帮助我们优化;
2. 只不过开启了生产环境, 就自动开启了压缩, 除了 `css` 需要我们自己弄;
3. 我们使用自带的 `Terser-webpack-plugin` 插件来设置自己的压缩选项;

```
//打包优化插件
const TerserWebpackPlugin = require('terser-webpack-plugin')

//优化
optimization : {
  //如果是生成模式, 自动为 true
  //minimize : true,

  //配置 TW 插件
  minimizer : [new TerserWebpackPlugin({
    //是否开启缓存, 开启缓存加速
    cache : false,
    //插件选项
    terserOptions : {
      //压缩方式
      compress : {
        //剔除无用代码
        unused : true,
        //剔除死代码
        dead_code : true,
        //剔除 console.log
        drop_console : true,
        //剔除 debugger
        drop_debugger : true,
      }
    }
  })]
},
```

二. dist 服务

1. 打包后，我们想要在静态服务器上测试，先要安装静态服务器；

```
npm i serve -g      //安装  
serve dist -s       //运行 dist 目录
```