1. nanoid：生成独一无二的id npm i nanoid
   1. 引入：import {nanoid} from 'nanoid'
   2. 调用：nanoid（），返回字符串
2. animate：动画库，配合Vue使用 npm i animate.css
   1. 引入：import ‘animate.css’
   2. 使用：transition标签中添加name=“animate\_\_animated animate\_\_bounce” enter/leave-active-class=“指定动画的类名”（官网上找）
3. eslint：语法检查，npm i eslint
   1. package.json中添加："scripts": {"lint": "eslint ./src"，"lint:create": "eslint --init" }
   2. 初始化：npm run lint:create
   3. 使用： npm run lint检查指定src目录下的所有文件
4. husky：在git提交前使用eslint检查语法，在git仓库中npm i husky -D
   1. 注册：
      1. npm set-script prepare "husky install" npm run prepare 初始化
      2. npx husky add .husky/pre-commit "npm run lint" 添加钩子，钩子名同eslint的命令
      3. git add .husky/pre-commit 添加对钩子的修改到暂存区
      4. git commit -m "xx" 提交，此后在每次提交前都会调用npm run lint
5. serve：用于运行webpack打包后的文件。npm i -g serve
   1. 使用：serve dist
6. swiper：用于移动端网站的内容触摸滑动。npm i swiper
   1. 使用：见官网
7. nodemon：用于项目热更新，检测目录中的文件更改时重新启动应用程序。npm i nodemon
   1. 注册：package.json的“script”中添加"devstart": "nodemon ./server/xx.js"
   2. 使用：npm run devstart
8. qs：是一个增加了一些安全性的查询字符串解析和序列化字符串的库。npm i qs
   1. 使用：直接import引入（vue中使用axios与JSON.stringify传递参数出错时可用qs.stringify（）替代）
   2. 方法：qs.stringify（对象）； qs.parse（）；