



链滴

yarn 的常用命令

作者: [dianjiu](#)

原文链接: <https://ld246.com/article/1599487210183>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)



yarn安装

```
npm install -g yarn
```

查看版本

```
yarn -v
```

开始一个新工程

yarn init 与 npm init 一样通过交互式会话创建一个 package.json

```
yarn init # yarn  
npm init # npm
```

跳过会话，直接通过默认值生成 package.json

```
yarn init --yes # 简写 -y  
npm init -y
```

添加一个依赖

通过 yarn add 添加依赖会更新 package.json 以及 yarn.lock 文件

1).开发环境

yarn add <packageName> 依赖会记录在 package.json 的 dependencies 下 开发环境

```
yarn add webpack@2.3.3 # yarn --save 是 yarn 默认的，默认记录在 package.json 中
```

```
npm install webpack@2.3.3 --save # npm
```

2).生产环境

yarn add <packageName> --dev 依赖会记录在 package.json 的 devDependencies 下 生产环境

```
yarn add webpack --dev # yarn 简写 -D  
npm install webpack --save-dev # npm
```

3).全局

yarn global add <packageName> 全局安装依赖

```
yarn global add webpack # yarn  
npm install webpack -g # npm
```

更新一个依赖

```
yarn upgrade 用于更新包到基于规范范围的最新版本  
yarn upgrade # 升级所有依赖项, 不记录在 package.json 中  
npm update # npm 可以通过 '--save|--save-dev' 指定升级哪类依赖  
yarn upgrade webpack # 升级指定包  
npm update webpack --save-dev # npm  
yarn upgrade --latest # 忽略版本规则, 升级到最新版本, 并且更新 package.json
```

移除一个依赖

```
yarn remove
```

安装 package.json 中的所有文件

```
yarn 或者 yarn install  
yarn install # 或者 yarn 在 node_modules 目录安装 package.json 中列出的所有依赖  
npm install # npm  
yarn install 安装时, 如果 node_modules 中有相应的包则不会重新下载 --force 可以强制
```

重新下载安装

```
yarn install --force # 强制下载安装  
npm install --force # npm
```

运行脚本

yarn run 用来执行在 package.json 中 scripts 属性下定义的脚本

```
// package.json  
{  
  "scripts": {  
    "dev": "node app.js",  
    "start": "node app.js"  
  }  
}
```

```
}  
yarn run dev # yarn 执行 dev 对应的脚本 node app.js  
npm run # npm  
yarn start # yarn  
npm start # npm  
与 npm 一样 可以有 yarn start 和 yarn test 两个简写的运行脚本方式
```

显示某个包信息

```
yarn info
```

列出项目的所有依赖

```
yarn list  
yarn list # 列出当前项目的依赖  
npm list # npm  
yarn list --depth=0 # 限制依赖的深度  
sudo yarn global list # 列出全局安装的模块
```

管理 yarn 配置文件

```
yarn config  
yarn config set key value # 设置  
npm config set key value  
yarn config get key # 读取值  
npm config get key  
yarn config delete key # 删除  
npm config delete key  
yarn config list # 显示当前配置  
npm config list  
yarn config set registry https://registry.npm.taobao.org # 设置淘宝镜像  
npm config set registry https://registry.npm.taobao.org # npm
```

缓存

```
yarn cache  
sudo yarn cache list # 列出已缓存的每个包  
sudo yarn cache dir # 返回 全局缓存位置  
sudo yarn cache clean # 清除缓存
```