

02

OPEN ORIENTED

凹凸实验室

# 开始你的第一个 npm 脚手架

## 被 npm 包围的开发环境

在实际开发中，一般刚开始一个项目，我们会运行 `npm install` 下载安装所有依赖，在实际开发中，可能也会使用各种命令行…… 来提高我们开发的效率。

athena

ko3

.....

小程序新建文件都需要手动新建多个文件

# 理想的开发模式

```
→ nsp git:(master) ✕ nsp page index  
? 这个页面叫什么? 首页  
页面创建成功, 开始你的工作吧~  
→ nsp git:(master) ✕
```

```
→ usb diff:(master) ✕
```

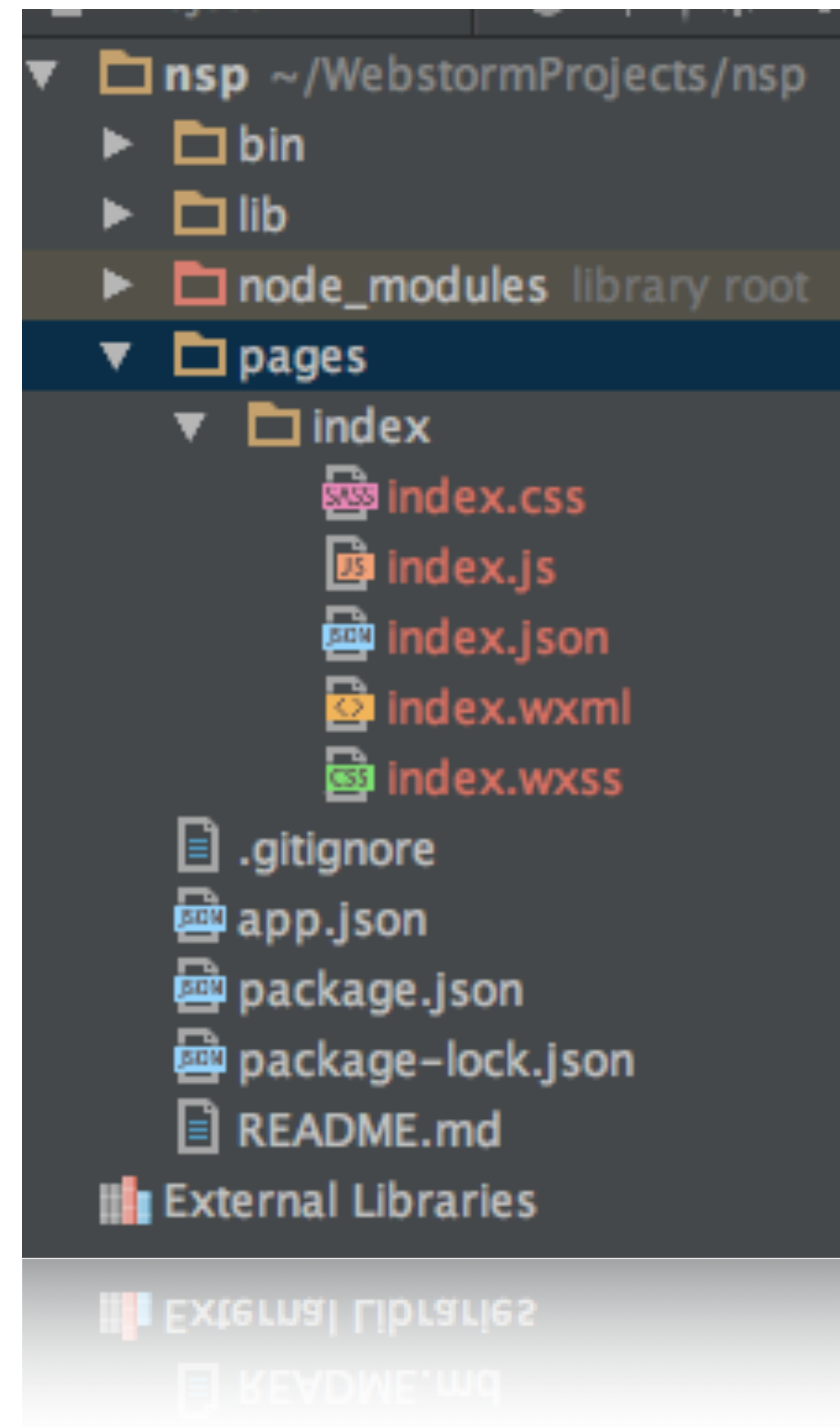
生成 5 个文件名相同的文件



自动添加页面路径【app.json-pages】



自动加上 title 名



```
"repository": {  
  "type": "git",  
  "url": ""  
},  
"bin": {  
  "nsp": "./bin/index.js"  
},  
"scripts": {  
  "build": "",  
}
```

指明代码被托管在何处  
npm docs 可以直接打开  
如果非 github 链接，就会去 npm 搜索name

执行 npm link 命令，它将会把 nsp  
这个字段复制到 npm 的全局模块安装  
文件夹 node\_modules 内，并创建  
符号链接（symbolic link，软链接）

# 解析命令行参数

```
program
```

```
  .usage('[command] [options]') //在help中告诉用户如何使用
```

```
  .command('page [pageName]') //定义命令行指令
```

```
  .alias('pa') //重命名, 定义一个更短的指令
```

```
  .description('新建页面') //描述
```

```
  .action((pageName) => {  
    console.log(pageName);  
  })
```

```
// 自定义帮助信息
```

```
  .on('--help', function () {  
    console.log('  Examples:');  
    console.log('    $ nsp page index');  
    console.log('    $ nsp pa index');  
  });
```

```
});
```

```
program.parse(process.argv); //解析命令行
```

process.argv

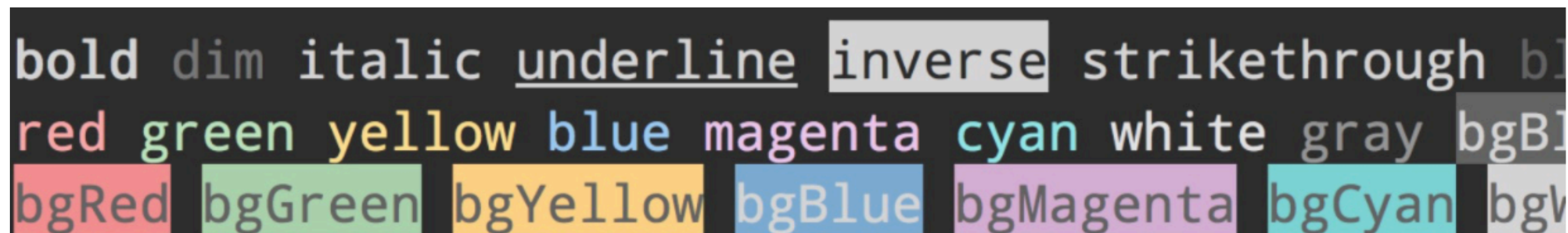
commander



```
var questions = [
  {
    type: 'input',      // input, confirm, list, rawlist, expand,
checkbox, password, editor
    name: 'selection',  //可以在answer用这个值获取输入的值
    message: '这个页面是做什么的?', //命令行中询问用户
    default: false,    //设置默认值,一般来说,会出现在选择中
  }
];

inquirer.prompt(questions).then((answers, input)=> { }) //解析问题
```

## chalk



bold dim italic underline inverse strikethrough b  
red green yellow blue magenta cyan white gray bgB  
bgRed bgGreen bgYellow bgBlue bgMagenta bgCyan bgV

```
console.log(chalk.red('该项目已存在，换个名字试试？'))
```



## 判断用户是否在正确目录下执行命令行

`process.cwd()`   //当前执行 node 命令时候的文件夹地址  
`__dirname`       //被执行的 js 文件的地址 —— 文件所在目录

```
var isCorrect = function () {  
  var baseUrl = path.resolve(__dirname, '../..'),  
      curUrl = process.cwd();  
  
  if(baseUrl !== curUrl) {  
    console.log(chalk.red('  请在根目录进行'));  
    process.exit(1);  
  }  
}
```

## 判断是否存在该目录

```
fs.existsSync(path)
```

## 新建文件夹，文件

```
fs.mkdir(path)  
fs.writeFile(path, content)
```

\$ npm who i am

\$ npm login

\$ npm publish

**T H A N K S**  
**FOR YOUR WATCHING**

