

React 项目实战

<https://cnnodejs.org/>

项目结构

1. 页面

```
view
|- not-found
|  |- index.tsx // 404 页面
|- about
|  |- index.tsx // 关于页面
|- user
|  |- index.tsx // 用户信息页面
|  |- style.tsx
|- topic
|  |- index.tsx // 各种主题的文章列表页
|  |- card
|  |  |- card.tsx // 列表项
|  |  |- style.tsx
|- article
|  |- index.tsx // 文章详情页
|  |- style.tsx
|  |- code-prettify-sunburst.css
|  |- comment
|  |  |- index.tsx // 评论详情
|  |  |- style.tsx
|  |- comment-panel // 评论面板
|  |  |- index.tsx
|  |  |- style.tsx
|  |- info-bar
|  |  |- index.tsx // 文章顶部info
|  |  |- style.tsx
```

2. 组件

```
components
|- loading
|   |- index.tsx // 加载中
|- header
|   |- index.tsx // 顶部header
|- image
|   |- index.tsx // 图片组件
|   |- style.tsx
|- scroll-list
|   |- index.tsx // 滚动列表
|- tabbar
|   |- index.tsx // 顶部nav tabbar
|   |- style.tsx
|- tag
|   |- index.tsx // 文章标签
```

hooks

React.memo

函数组件，使用 React.memo ，将函数组件传递给 memo 之后，就会返回一个新的组件，新组件的功能：如果接受到的属性不变，则不重新渲染函数

useCallback

接收一个内联回调函数参数和一个依赖项数组（子组件依赖父组件的状态，即子组件会使用到父组件的值），useCallback 会返回该回调函数的 memoized 版本，该回调函数仅在某个依赖项改变时才会更新

useRef

useRef 返回的 ref 对象在组件的整个生命周期内保持不变，也就是说每次重新渲染函数组件时，返回的 ref 对象都是同一个

自定义 hooks

1. useAsync 用来发起异步请求, 添加 loading 等各种通用信息
2. useLoadMore 用来下拉加载
3. useInitPosition 用来初始化位置

开始!

1. 路由
2. Tabbar
3. topic 列表页面
 - 3.1 service 请求封装

- 3.2 scrollList 滚动列表
 - 3.3 card 列表项
 - 3.4 useLoadMore 下拉刷新
- 4. article 文章详情页面
 - 4.1 useAsync 异步请求 loading
 - 4.2 useInitPosition
 - 4.3 infoBar
 - 4.4 CommentPanel
 - 4.5 Comment
- 5. user
- 6. not-found
- 7. about