

React-3

上节课重点内容回顾

- 组件间的信息传递
- 生命周期
- 受控组件

课程目标

- 掌握React其他 API 使用：PureComponent、ref、children、dangerouslySetInnerHTML、key
- 掌握函数式组件及常见 hooks 的使用
- 可以独立开发基于Hooks 或 Component 的 todoList 应用

课程内容

PureComponent

PureComponent 提供了一个具有浅比较的 shouldComponentUpdate 方法,其他和 Component 完全一样

key 的问题

ref

- 旧版 - string ref
- 主流 - createRef()
- 注意：在 组件挂载完成之后及更新之后使用

children

- 组件标签对之间的内容会被当做一个特殊的属性 props.children 传入组件内容
- 可以自定义结构的组件的常用形式
 - children
 - 传递函数
 - 传递子组件

实战 - 完整todoList

函数式组件

函数式组件中，没有 state 和 生命周期，所以又被称为 无状态组件，在早期 react 版本中，被当前纯渲染组件来用

React hooks

React hooks 是React 16.8中的新增功能。它们使您无需编写类即可使用状态和其他React功能 参考：
<https://reactjs.org/docs/hooks-intro.html>

React Hooks 优势

- 简化组件逻辑
- 复用状态逻辑
- 无需使用类组件编写

常用 hook

- useState

```
const [state, setState] = useState(initialState);
```
- useEffect 类组件 componentDidMount、componentDidUpdate 和 componentWillUnmount 需要清除的副作用
- useRef

Hook 使用规则

- 只在最顶层使用 Hook
- 只在 React 函数中调用 Hook
 - React 函数组件中
 - React Hook 中

练习

- todoList
- 观看：react 录播的前13集

总结

下节课内容

- React-router-dom

练习

利用 class 和 hooks 各自实现一遍 todoList