

# React 相关

## 推荐的工具链

React 团队主要推荐这些解决方案：

- 如果你是在学习 React 或创建一个新的单页应用，请使用[Create React App](#)。
- 如果你是在用 Node.js 构建服务端渲染的网站，试试[Next.js](#)。
- 如果你是在构建面向内容的静态网站，试试[Gatsby](#)。
- 如果你是在打造组件库或将 React 集成到现有代码仓库，尝试[更灵活的工具链](#)。

reactRouteDom-v6

<https://yanxiaolazy.github.io/2022/02/15/react-router%20v6%20api%E5%AD%A6%E4%B9%A0/>

1. 路由实现 BrowserRouter
  - a. Route Routes 的区别
  - b. use
2. 路由嵌套
  - a. Outlet 的用法
    - i. **useOutlet()**
    - ii. **useOutletContext()**
  - b. other way

```
export const Team = () => {
  return (
    <Layout style={{marginTop: 48, padding: '24px', background: '#e5e5e5'}}>
      <Header><WelcomeHeader title="FlexGalaxy" /></Header>
      <Content
        className="site-layout-background"
        style={{
          margin: 0,
          minHeight: '280px',
        }}>
        { /*<Outlet />*/ }
        <Routes>
          <Route path="teamOperation"></Route>
        </Routes>
      </Content>
    </Layout>
  );
}
```

3. 路由守卫的实现：<https://www.robinwieruch.de/react-router-private-routes/>
4. useNavigate 中的 state 的使用方法与 useParams 的区别
5. useSearchParams search
6. useLocation 的使用方式（获取 useNavigate 的 state）
7. useMatch("/team"); useEffect()

## 8. 404 实现

- a. Not Found Routes 概念
- b. Index Routes 概念

## 9. hooks

- a. **useLinkClickHandler**
- b. useHref

## 10. 为什么不能再循环，条件或嵌套函数中调用 React Hook

- a. 比如：

- i. [https://blog.csdn.net/weixin\\_43484007/article/details/124354511](https://blog.csdn.net/weixin_43484007/article/details/124354511)

```
return (  
  <Button onClick={()=>{  
    const navigate = useNavigate()  
  }}></Button>  
  
);
```

## 11. 自定义hook 相互调用