link: null
title: 珠峰架构师成长计划
description: null
keywords: null
author: null
date: null
publisher: 珠峰架构师成长计划
stats: paragraph=216 sentences=647, words=3329

1. React路由

- 不同的路径渲染不同的组件
- 有两种实现方式
    - HashRouter:利用hash实现路由切换
    - BrowserRouter:实现h5 Api实现路由的切换

1.1 HashRouter

- 利用hash实现路由切换

```
    去a
    去b

      window.addEventListener('hashchange',()=>{
          console.log(window.location.hash);
      });
```

1.2 BrowserRouter

- 利用h5 Api实现路由的切换

1.2.1 history

- history (https://developer.mozilla.org/zh-CN/docs/Web/API/Window/history)对象提供了操作浏览器会话历史的接口。
- historylength 属性声明了浏览器历史列表中的元素数量
- pushState (https://developer.mozilla.org/zh-CN/docs/Web/API/History_API) HTML5引入了 history.pushState() 和 history.replaceState() 方法，它们分别可以添加和修改历史记录条目。这些方法通常与window.onpopstate 配合使用
- onpopstate (https://developer.mozilla.org/zh-CN/docs/Web/API/Window/onpopstate) window.onpopstate是popstate事件在window对象上的事件处理程序

1.2.2 pushState

- pushState会往History中写入一个对象，他造成的结果便是,History length +1、url 改变、该索引History对应有一个State对象,这个时候若是点击浏览器的后退，便会触发popstate事件，将刚刚的存入数据对象读出
- pushState 会改变History
- 每次使用时候会为该索引的State加入我们自定义数据
- 每次我们会根据State的信息还原当前的view，于是用户点击后退便有了与浏览器后退前进一致的感受
- pushState() 需要三个参数: 一个状态对象, 一个标题 (目前被忽略), 和 (可选的) 一个URL
- 调用history.pushState()或者history.replaceState()不会触发popstate事件. popstate事件只会在浏览器某些行为下触发, 比如点击后退、前进按钮(或者在JavaScript中调用history.back()、history.forward()、history.go()方法)

```

window.onpopstate = (event) => {
    console.log({state:event.state,pathname:window.location.pathname,type:'popstate'});
};
window.onpushstate = (event) => {
    console.log(event);
};
(function (history) {
    var pushState = history.pushState;
    history.pushState = function (state,title,pathname) {
        if (typeof window.onpushstate == "function") {
            window.onpushstate({state,pathname,type:'pushstate'});
        }
        return pushState.apply(history, arguments);
    };
})(window.history);

//绑定事件处理函数.

setTimeout(()=>{
    history.pushState({page: 1}, "title 1", "/page1");
},1000);
setTimeout(()=>{
    history.pushState({page: 2}, "title 2", "/page2");
},2000);

setTimeout(()=>{
    history.replaceState({page: 3}, "title 3", "/page3");
},3000);

setTimeout(()=>{
    history.back();
},4000);

setTimeout(()=>{
    history.go(1);
},5000);
```

2. 跑通路由

2.1 src\index.js

```
import React from 'react';
import ReactDOM from 'react-dom';
import {HashRouter as Router,Route} from 'react-router-dom';
import Home from './components/Home';
import User from './components/User';
import Profile from './components/Profile';
ReactDOM.render(
    <Router>
        <div>
          <Route path="/" component={Home} />
          <Route path="/user" component={User} />
          <Route path="/profile" component={Profile}/>
        div>
    Router>
,document.getElementById('root'));
```

2.2 components\Home.js

src\components\Home.js

```
import React,{Component} from 'react';
export default class Home extends Component{
    render() {
        return (
            <div>Homediv>
        )
    }
}
```

2.3 components\User.js

src\components\User.js

```
import React,{Component} from 'react';
export default class User extends Component{
    render() {
        console.log(JSON.stringify(this.props));
        return (
            <div>Userdiv>
        )
    }
}

```

2.4 components\Profile.js

src\components\Profile.js

```
import React,{Component} from 'react';
export default class Profile extends Component{
    render() {
        return (
            <div>Profilediv>
        )
    }
}
```

3. 实现路由

3.1 index.js

src\react-router-dom\index.js

```
import HashRouter from './HashRouter';
import Route from './Route';
export {
    HashRouter,
    Route
}
```

3.2 HashRouter.js

src\react-router-dom\HashRouter.js

```
import React, { Component } from 'react'
import Context from './context';
export default class HashRouter extends Component {
  state = {
      location:{pathname:window.location.hash.slice(1)}
  }
  componentWillMount(){
      window.addEventListener('hashchange',()=>{
          this.setState({
              location:{
                  ...this.state.location,
                  pathname:window.location.hash.slice(1) || '/'
              }
          });
      });
      window.location.hash = window.location.hash || '/';
  }
  render() {
    let value = {
        location:this.state.location
    }
    return (
      <Context.Provider value={value}>
        {this.props.children}
      Context.Provider>
    )
  }
}
```

3.4 Route.js

src\react-router-dom\Route.js

```
import React,{Component} from 'react';
import Context from './context';
export default class Route extends Component{
    static contextType =  Context;
    render() {
        let {path,component: Component}=this.props;
        let pathname = this.context.location.pathname;
        if (path == pathname) {
            return <Component/>
        } else {
            return null;
        }
    }
}
```

3.4 context.js

src\react-router-dom\context.js

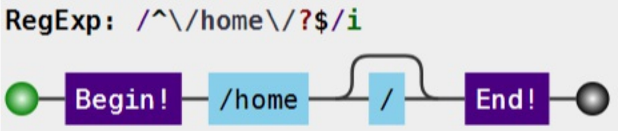
```
import {createContext} from 'react';
export default  createContext();
```

4.path-to-regexp

- path-to-regexp (https://www.npmjs.com/package/path-to-regexp)
- regulex (https://jex.im/regulex)

4.1 /home结束

```
let pathToRegExp = require('path-to-regexp');
let regx = pathToRegExp('/home',[],{end:true});
console.log(regx);
console.log(regx.test('/home'));
console.log(regx.test('/home/2'));
```



4.2 /home非结束

```
let pathToRegExp = require('path-to-regexp');
let regx2 = pathToRegExp('/home',[],{end:false});
console.log(regx2);
console.log(regx2.test('/home'));
console.log(regx2.test('/home/'));
console.log(regx2.test('/home//'));
console.log(regx2.test('/home/2'));
```

4.3 路径参数

```
let params = [];
let regx3 = pathToRegExp('/user/:id',params,{end:true});
console.log(regx3,params);
```

4.4 正则匹配

表达式 含义 () 表示捕获分组，()会把每个分组里的匹配的值保存起来，使用$n(n是一个数字，表示第n个捕获组的内容) (?:) 表示非捕获分组，和捕获分组唯一的区别在于，非捕获分组匹配的值不会保存起来

- 正则匹配的前瞻就是给正则匹配的选项定义一个断言，或者说是一个条件比如：我要匹配一个字母，但是我的需求是字母后面必须是跟着一个数字的情况，那么这种场景是怎么实现了，就是用到前瞻的概念，那么我想要他的前面也要是一个数字怎么办了，这就是后顾

表达式 含义 (?=pattern) 正向肯定查找(前瞻),后面必须跟着什么 (?!pattern) 正向否定查找(前瞻)，后面不能跟着什么 (?

```
console.log('1a'.match(/\d(?=[a-z])/));
console.log('1@'.match(/\d(?![a-z])/));
console.log('a1'.match(/(?));
console.log('$1'.match(/(?));
```

5.正则匹配

5.1 Route.js

src\react-router-dom\Route.js

```
import React, { Component } from 'react'
import Context from './context';
import pathToRegexp from 'path-to-regexp';
export default class Route extends Component {
  static contextType =  Context;
  render() {
    let {path,component:Component,exact=false} = this.props;
    let pathname = this.context.location.pathname;
    let regxp = pathToRegexp(path,[],{end:exact});
    let result = pathname.match(regxp);
    console.log('regxp,pathname,result',regxp,pathname,result);
    if(result){
        return <Component/>;
    }
    return null;
  }
}
```

6.1 src/index.js

src/index.js

```
import React from 'react';
import ReactDOM from 'react-dom';
import {HashRouter as Router,Route,Link} from './react-router-dom';
import Home from './components/Home';
import User from './components/User';
import Profile from './components/Profile';
ReactDOM.render(

        <>
          HomeUserProfile

        </>

,document.getElementById('root'));
```

src/react-router-dom/Link.js

```
import React, { Component } from 'react'
import Context from './context';
export default class Link extends Component {
  static contextType =  Context;
  render() {
    return (
        <a onClick={()=>this.context.history.push(this.props.to)}>{this.props.children}a>
    )
  }
}
```

6.3 index.js

src/react-router-dom/index.js

```
import HashRouter from './HashRouter';
import Route from './Route';
import Link from './Link';
export {
    HashRouter,
    Route,
    Link
}
```

6.4 HashRouter.js

src/react-router-dom/HashRouter.js

```
import React, { Component } from 'react'
import Context from './context';
export default class HashRouter extends Component {
  state = {
      location:{pathname:window.location.hash.slice(1) || '/'},
  }
  componentDidMount(){
      window.location.hash = window.location.hash || '/';
      window.addEventListener('hashchange',()=>{
          this.setState({
              location:{
                  ...this.state.location,
                  pathname:window.location.hash.slice(1) || '/'
              }
          });
      });
  }
  render() {
     let value={
            location: this.state.location,
            history: {
                push(to) {
                    window.location.hash=to;
                }
            }
    }
    return (
      <Context.Provider value={value}>
        {this.props.children}
      Context.Provider>
    )
  }
}
```

7. 引入bootstrap

7.1 src\index.js

src\index.js

```
import React from 'react';
import ReactDOM from 'react-dom';
import {HashRouter as Router,Route,Link} from './react-router-dom';
import Home from './components/Home';
import User from './components/User';
import Profile from './components/Profile';
import 'bootstrap/dist/css/bootstrap.css';
ReactDOM.render(

        <>

                  珠峰架构

                Home
                User
                Profile

        </>

,document.getElementById('root'));
```

8. Redirect&Switch

8.1 src\index.js

src\index.js

```
import React from 'react';
import ReactDOM from 'react-dom';
import {HashRouter as Router,Route,Link,Redirect,Switch} from './react-router-dom';
import Home from './components/Home';
import User from './components/User';
import Profile from './components/Profile';
import 'bootstrap/dist/css/bootstrap.css';
ReactDOM.render(

        <>

                  珠峰架构

                Home
                User
                Profile

        </>

,document.getElementById('root'));
```

8.2 index.js

src/react-router-dom/index.js

```
import HashRouter from './HashRouter';
import Route from './Route';
import Link from './Link';
import Redirect from './Redirect';
import Switch from './Switch';
export {
    HashRouter,
    Route,
    Link,
    Redirect,
    Switch
}
```

src/react-router-dom/Link.js

```
import React, { Component } from 'react'
import Context from './context';
export default class Link extends Component {
  static contextType =  Context;
  render() {
    return (
        <a onClick={()=>this.context.history.push(this.props.to)}>{this.props.children}a>
    )
  }
}
```

8.4 Route.js

src/react-router-dom/Route.js

```
import React, { Component } from 'react'
import Context from './context';
import pathToRegexp from 'path-to-regexp';
export default class Route extends Component {
  static contextType =  Context;
  render() {
    let {path,component:Component,exact=false} = this.props;
    let pathname = this.context.location.pathname;
    let regxp = pathToRegexp(path,[],{end:exact});
    let result = pathname.match(regxp);
    if(result){
        return <Component {...this.context}/>;
    }
    return null;
  }
}
```

8.5 Redirect.js

src/react-router-dom/Redirect.js

```
import React, { Component } from 'react'
import Context from './context';
export default class Link extends Component {
  static contextType = Context;
  render() {
    this.context.history.push(this.props.to);
    return null;
  }
}
```

8.6 Switch.js

src/react-router-dom/Switch.js

```
import React, { Component } from 'react'
import Context from './context';
import pathToRegexp from 'path-to-regexp';
export default class Switch extends Component {
  static contextType =  Context;
  render() {
    let pathname = this.context.location.pathname;
    for(let i=0;i<this.props.children.length;i++){
        let child = this.props.children[i];
        let {path='/',component:Component,exact=false} = child.props;
        let regxp = pathToRegexp(path,[],{end:exact});
        let result = pathname.match(regxp);
        if(result){
            return child;
        }
    }
    return null;
  }
}
```

9. 路径参数

9.1 User.js

src/components/User.js

```
import React,{Component} from 'react';
import {Link,Route} from '../react-router-dom';
import UserAdd from './UserAdd';
import UserDetail from './UserDetail';
import UserList from './UserList';
export default class User extends Component{
    render() {
        return (
            <div className="row">
               <div className="col-md-2">
                       <ul className="nav nav-stack">
                        <li><Link to="/user/list">用户列表Link>li>
                        <li><Link to="/user/add">添加用户Link>li>
                    ul>
               div>
               <div className="col-md-10">
                       <Route path="/user/add" component={UserAdd}/>
                    <Route path="/user/list" component={UserList}/>
                    <Route path="/user/detail/:id" component={UserDetail}/>
               div>
            div>
        )
    }
}
```

9.2 UserAdd.js

src/components/UserAdd.js

```
import React, { Component } from 'react'

export default class UserAdd extends Component {
  constructor(){
    super();
    this.usernameRef = React.createRef();
  }
  handleSubmit = (event)=>{
    event.preventDefault();
    let username = this.usernameRef.current.value;
    let usersStr = localStorage.getItem('users');
    let users = usersStr?JSON.parse(usersStr):[];
    users.push({id:Date.now()+'',username});
    localStorage.setItem('users',JSON.stringify(users));
    this.props.history.push('/user/list');
  }
  render() {
    return (
      <form onSubmit={this.handleSubmit}>
        <input className="form-control" type="text" ref={this.usernameRef}/>
        <button type="submit" className="btn btn-primary">提交button>
      form>
    )
  }
}
```

9.3 UserList.js

src/components/UserList.js

```
import React, { Component } from 'react'
import {Link} from '../react-router-dom';
export default class UserList extends Component {
  state = {
      users:[]
  }
  componentDidMount(){
    let usersStr = localStorage.getItem('users');
    let users = usersStr?JSON.parse(usersStr):[];
    this.setState({users});
  }
  render() {
    return (
      <div>
        UserList
        <ul className="list-group">
        {
            this.state.users.map((user,index)=>(
                <li className="list-group-item" key={index}>
                 <Link to={{pathname:`/user/detail/${user.id}`,state :user}}>{user.username}Link>
                li>
            ))
        }
        ul>
      div>
    )
  }
}
```

9.3 UserDetail.js

src/components/UserDetail.js

```
import React, { Component } from 'react'

export default class UserDetail extends Component {
  state = {
    user:{}
  }
  componentDidMount(){
    let user = this.props.location.state;
    if(!user){
      let usersStr = localStorage.getItem('users');
      let users = usersStr?JSON.parse(usersStr):[];
      let id = this.props.match.params.id;
      user = users.find(user=>user.id === id);
    }
    if(user) this.setState({user});
  }
  render() {
    let user = this.state.user;
    return (
      <div>
        {user.id}:{user.username}
      div>
    )
  }
}
```

9.4 HashRouter.js

src/react-router-dom/HashRouter.js

```
import React, { Component } from 'react'
import Context from './context';
export default class HashRouter extends Component {
  state = {
      location:{pathname:window.location.hash.slice(1) || '/'}
  }
  locationState=undefined
  componentDidMount(){
      window.location.hash = window.location.hash || '/';
      window.addEventListener('hashchange',()=>{
          this.setState({
              location:{
                  ...this.state.location,
                  pathname:window.location.hash.slice(1) || '/',
                  state:this.locationState
              }
          });
      });
  }
  render() {
    let that = this;
    let value = {
      location:that.state.location,
      history:{
        push(to){
          if(typeof to === 'object'){
            let {pathname,state}= to;
            that.locationState = state;
            window.location.hash = pathname;
          }else{
            window.location.hash = to;
          }
      }
      }
    }
    return (
      <Context.Provider value={value}>
        {this.props.children}
      Context.Provider>
    )
  }
}
```

9.5 Route.js

src/react-router-dom/Route.js

```
import React, { Component } from 'react'
import Context from './context';
import pathToRegexp from 'path-to-regexp';
export default class Route extends Component {
  static contextType = Context;
  render() {
    let { path, component: Component, exact = false } = this.props;
    let pathname = this.context.location.pathname;
    let keys = [];
    let regxp = pathToRegexp(path, keys, { end: exact });
    let result = pathname.match(regxp);
    if (result) {
      let [url, ...values] = result;
      keys = keys.map(item=>item.name);
      let params = values.reduce((memo, val, index) => {
        memo[keys[index]] = val;
        return memo;
      }, {});
      let match = {
        url:pathname,
        isExact:pathname===url,
        path,
        params
      }
      let props = {
        location: this.context.location,
        history: this.context.history,
        match
      }
      return <Component {...props} />;
    }
    return null;
  }
}
```

9.6 Switch.js

src/react-router-dom/Switch.js

```
import React, { Component } from 'react'
import Context from './context';
import pathToRegexp from 'path-to-regexp';
export default class Switch extends Component {
  static contextType =  Context;
  render() {
    let pathname = this.context.location.pathname;
    let children = Array.isArray(this.props.children)?this.props.children:[this.props.children]
    for(let i=0;ilet child = children[i];
        let {path='/',exact=false} = child.props;
        let regxp = pathToRegexp(path,[],{end:exact});
        let result = pathname.match(regxp);
        if(result){
            return child;
        }
    }
    return null;
  }
}
```

10. 受保护的路由

10.1 src/index.js

src/index.js

```
import React from 'react';
import ReactDOM from 'react-dom';
import {HashRouter as Router,Route,Link,Redirect,Switch} from './react-router-dom';
import Home from './components/Home';
import User from './components/User';
import Profile from './components/Profile';
import Protected from './components/Protected';
import Login from './components/Login';
import 'bootstrap/dist/css/bootstrap.css';
ReactDOM.render(

        <>

                  珠峰架构

                Home
                User
                Profile

        </>

,document.getElementById('root'));
```

10.2 Route.js

src/react-router-dom/Route.js

```
      let props = {
        location: this.context.location,
        history: this.context.history,
        match
      }
      if(Component){
        return <Component {...props} />;
      }else if(render){
        return render(props);
      }else{
        return null;
      }
```

10.3 Protected.js

src/components/Protected.js

```
import React, { Component } from 'react'
import {Route,Redirect} from '../react-router-dom';
export default ({component:Component,...rest})=>(
    <Route {...rest} render={
        props => (
            localStorage.getItem('logined')?<Component {...props}/>:<Redirect to={{pathname:'/login',state:{from:props.location.pathname}}}/>
        )
    }/>
)
```

10.4 Login.js

src/components/Login.js

```
import React, { Component } from 'react'

export default class Login extends Component {
  handleClick = ()=>{
    localStorage.setItem('logined','true');
    if(this.props.location.state)
      this.props.history.push(this.props.location.state.from);
  }
  render() {
    return (
      <button className="btn btn-primary" onClick={this.handleClick}>登录button>
    )
  }
}
```

11. 自定义导航

11.1 index.js

src/index.js

```
+    Home
+    User
+    Profile
```

src/components/MenuLink.js

```
import React from 'react'
import {Route,Link} from '../react-router-dom';
import './MenuLink.css'
export default ({to,exact,children}) => (
    <Route
        path={to}
        exact={exact}
        children={
            props => (
                <Link className={props.match?'active':''} to={to}>{children}Link>
            )
        }
    />
)
```

src/components/MenuLink.css

```
.navbar-inverse .navbar-nav > li > .active{
    background-color: green;
    color:red;
}
```

src\react-router-dom\Link.js

```
import React, { Component } from 'react'
import Context from './context';
export default class Link extends Component {
  static contextType =  Context;
  render() {
    return (
+         this.context.history.push(this.props.to)}>{this.props.children}
    )
  }
}
```

11.5 Route.js

src/react-router-dom/Route.js

```
import React, { Component } from 'react'
import Context from './context';
import pathToRegexp from 'path-to-regexp';
export default class Route extends Component {
  static contextType = Context;
  render() {
    let { path='/', component: Component, exact = false,render,children } = this.props;
    let pathname = this.context.location.pathname;
    let keys = [];
    let regxp = pathToRegexp(path, keys, { end: exact });
    let result = pathname.match(regxp);
+    let props = {
+      location: this.context.location,
+      history: this.context.history
+    }
    if (result) {
      let [url, ...values] = result;
      keys = keys.map(item=>item.name);
      let params = values.reduce((memo, val, index) => {
        memo[keys[index]] = val;
        return memo;
      }, {});
      let match = {
        url:pathname,
        isExact:pathname
        path,
        params
      }
+      props.match = match;
      if(Component){
        return ;
      }else if(render){
        return render(props);
+      }else if(children){
+        return children(props);
+      }else{
+        return null;
+      }
+    }else if(children){
+      return children(props);
+    }else{
+      return null;
+    }
  }
}
```

12. withRouter

12.1 src/index.js

```
+import NavHeader from './components/NavHeader';

+


                  Home
                  User
                  Profile

```

src/components/NavHeader.js

```
import React, { Component } from 'react'
import {withRouter} from '../react-router-dom';
class NavHeader extends Component {
  render() {
    return (
        <div className="navbar-heading">
            <div className="navbar-brand" onClick={()=>this.props.history.push('/')}>珠峰架构div>
        div>
    )
  }
}
export default withRouter(NavHeader)
```

12.3 withRouter.js

src/react-router-dom/withRouter.js

```
import React from 'react'
import {Route} from '../react-router-dom';
export default function (Component) {
    return props=>(
        <Route render={routeProps=><Component {...routeProps}>}/>
    )
}
```

12.4 index.js

src/react-router-dom/index.js

```
import withRouter from './withRouter';
export {
    HashRouter,
    Route,
    Link,
    Redirect,
    Switch,
    withRouter
}
```

13. 阻止跳转

13.1 UserAdd.js

src/components/UserAdd.js

```
import React, { Component } from 'react'
import {Prompt} from '../react-router-dom';
export default class UserAdd extends Component {
  constructor(){
    super();
    this.usernameRef = React.createRef();
  }
  state = {
    isBlocking:false
  }
  handleSubmit = (event)=>{
    event.preventDefault();
    this.setState({
      isBlocking:false
    },()=>{
      let username = this.usernameRef.current.value;
      let usersStr = localStorage.getItem('users');
      let users = usersStr?JSON.parse(usersStr):[];
      users.push({id:Date.now()+'',username});
      localStorage.setItem('users',JSON.stringify(users));
      this.props.history.push('/user/list');
    })

  }
  render() {
    let {isBlocking}=this.state;
    return (
      <form onSubmit={this.handleSubmit}>
       <Prompt
              when={isBlocking}
              message={location=>`你确定要跳转到${location.pathname}吗？`}
        />
        <input className="form-control" type="text"
           onChange={event => {
            this.setState({isBlocking:event.target.value.length>0});
           }}
           ref={this.usernameRef}/>
        <button type="submit" className="btn btn-primary">提交button>
      form>
    )
  }
}
```

13.2 HashRouter.js

src/react-router-dom/HashRouter.js

```
import React, { Component } from 'react'
import Context from './context';
export default class HashRouter extends Component {
  state = {
    location: { pathname: window.location.hash.slice(1) || '/' }
  }
  locationState = undefined
  getMessage=null
  componentDidMount() {
    window.location.hash = window.location.hash || '/';
    window.addEventListener('hashchange', () => {
      this.setState({
        location: {
          ...this.state.location,
          pathname: window.location.hash.slice(1) || '/',
          state: this.locationState
        }
      });
    });
  }
  render() {
    let that = this;
    let value = {
      location: that.state.location,
      history: {
        push(to) {
          if (that.getMessage) {
              let allow=window.confirm(that.getMessage(that.state.location)+`，并且跳转到${typeof to === 'object'?to.pathname:to}吗?`);
              if (!allow) return;
          }
          if (typeof to === 'object') {
            let { pathname, state } = to;
            that.locationState = state;
            window.location.hash = pathname;
          } else {
            window.location.hash = to;
          }
        },
        block(message) {
          that.getMessage=message;
        },
        unblock() {
          that.getMessage=null;
        }
      }
    }
    return (
      <Context.Provider value={value}>
        {this.props.children}
      Context.Provider>
    )
  }
}
```

13.3 index.js [#](#)

src/react-router-dom/index.js

```
import HashRouter from './HashRouter';
import Route from './Route';
import Link from './Link';
import Redirect from './Redirect';
import Switch from './Switch';
import withRouter from './withRouter';
import Prompt from './Prompt';
export {
    HashRouter,
    Route,
    Link,
    Redirect,
    Switch,
    withRouter,
    Prompt
}
```

13.4 Prompt.js [#](#)

src/react-router-dom/Prompt.js

```
import React from 'react'
import Context from './context';
export default class Prompt extends React.Component{
    static contextType = Context;
    componentWillUnmount() {
        this.history.unblock();
    }
    render() {
        this.history=this.context.history;
        const {when,message}=this.props;
        if (when) {
            this.history.block(message);
        } else {
            this.history.block(null);
        }
        return null;
    }
}
```

14. BrowserRouter [#](#)

14.1 index.js [#](#)

src/index.js

```
import {BrowserRouter as Router,Route,Redirect,Switch} from './react-router-dom';
```

14.2 BrowserRouter.js

src/react-router-dom/BrowserRouter.js

```
import React, { Component } from 'react'
import Context from './context';
(function (history) {
    var pushState = history.pushState;
    history.pushState = function (state,title,pathname) {
        if (typeof window.onpushstate == "function") {
            window.onpushstate(state,pathname);
        }
        return pushState.apply(history, arguments);
    };
})(window.history);
export default class BrowserRouter extends Component {
    state = {
        location: { pathname: '/' }
    }
    getMessage = null
    componentDidMount() {
        window.onpopstate = (event) => {
            this.setState({
                location: {
                    ...this.state.location,
                    pathname:document.location.pathname,
                    state:event.state
                }
            });
        };
        window.onpushstate = (state,pathname) => {
            this.setState({
                location: {
                    ...this.state.location,
                    pathname,
                    state
                }
            });
        };
    }
    render() {
        let that = this;
        let value = {
            location: that.state.location,
            history: {
                push(to) {
                    if (that.block) {
                        let allow = window.confirm(that.getMessage(that.state.location));
                        if (!allow) return;
                    }
                    if (typeof to === 'object') {
                        let { pathname, state } = to;
                        window.history.pushState(state, '', pathname);
                    } else {
                        window.history.pushState('', '', to);
                    }
                },
                block(getMessage) {
                    that.block = getMessage;
                },
                unblock() {
                    that.block = null;
                }
            }
        }
        return (
            <Context.Provider value={value}>
                {this.props.children}
            Context.Provider>
        )
    }
}
```

14.3 index.js

src/react-router-dom/index.js

```
import HashRouter from './HashRouter';
import Route from './Route';
import Link from './Link';
import Redirect from './Redirect';
import Switch from './Switch';
import withRouter from './withRouter';
import Prompt from './Prompt';
import BrowserRouter from './BrowserRouter';

export {
    HashRouter,
    Route,
    Link,
    Redirect,
    Switch,
    withRouter,
    Prompt,
    BrowserRouter
}
```