

页面结构

```
1  example-page
2      |-api
3          |- index.ts
4      |-components
5          |-delete-confirm
6              |-index.module.scss
7              |-index.tsx
8          |-main.module.scss
9          |-Main.tsx
10     |-interfaces
11         |-Data.ts
12     |-stores
13         |-index.ts
14         |-MainStore.ts
15     |-styles
16     |-index.tsx
```

api

index.ts

```
1  import axios from 'axios';
2  /** post接口 */
3  export const A = (
4      param: ITransferRule
5  ): Promise<IResult<null>> => {
6      return axios
7          .post('/xx', param)
8          .then(res => res.data);
9  };
10
11 /** delete接口 */
12 export const B = (id: number): Promise<IResult<null>> => {
13     return axios
14         .delete(`/xx/${id}`)
15         .then(res => res.data);
16 };
17
18 /** put接口 */
19 export const C = (
20     param: ITransferRule
21 ): Promise<IResult<null>> => {
22     return axios
23         .put('/xx', param)
24         .then(res => res.data);
25 };
26
27 /** get接口 */
28 export const D = (): Promise<IResult<null>> => {
```

```

29     return axios
30       .get('/xx?typeId=PRODUCT_TYPE')
31       .then(res => res.data);
32   };

```

components

Main.tsx

```

1  import React from 'react';
2  import { observer, inject } from 'mobx-react';
3  import { RouteConfigComponentProps } from 'react-router-config';
4  import { Spinner } from '@casstime/bricks';
5
6  import styles from './main.module.scss';
7  import MainStore from '../stores/MainStore';
8
9  import DeleteConfirm from './DeleteConfirm';
10
11 interface ITitleProps extends RouteConfigComponentProps<void> {
12   mainStore?: MainStore;
13 }
14
15 /** 组件备注 */
16 @inject('mainStore')
17 @observer
18 export default class Main extends React.Component<ITitleProps> {
19   componentDidMount() {
20     const { mainStore } = this.props;
21     const { init } = mainStore as MainStore;
22     /** 获取初始化数据 */
23     init ();
24   }
25
26   render() {
27     const { mainStore } = this.props;
28     const { isSaving } = mainStore as MainStore;
29     return (
30       <div className={styles.page}>
31         <DeleteConfirm />
32         {isSaving && <Spinner className={styles.spinner} />}
33       </div>
34     );
35   }
36 }
37

```

main.module.scss

```

1  :global{
2  }
3  .page{

```

```

4   padding: 24px 16px;
5 }
6 .spinner{
7   z-index: 1100;
8   position: fixed;
9   left: 50%;
10  top:50%;
11  transform: translate(-50%,-50%);
12 }
13

```

delete-confirm

index.tsx

```

1  import React from 'react';
2  import { observer, inject } from 'mobx-react';
3  import { Confirm, Icon, Backdrop } from '@casstime/bricks';
4
5  import styles from './index.module.scss';
6
7  import MainStore from '../stores/MainStore';
8
9  interface IDeleteConfirmProps {
10    mainStore?: MainStore;
11  }
12
13  @inject('mainStore')
14  @observer
15  class DeleteConfirm extends React.Component<IDeleteConfirmProps> {
16    render() {
17      const { mainStore } = this.props;
18      const { confirmVisible, toggleConfirm } = mainStore as MainStore;
19      return (
20        confirmVisible && (
21          <Backdrop visible={true}>
22            <Confirm
23              className={styles.confirm}
24              showShade={false}
25              topTitle="删除"
26              onClose={() => toggleConfirm(false)}
27              onOk={this.confirmDelete}
28              onCancel={() => toggleConfirm(false)}
29              tipIcon={
30                <Icon type="question-circle" color="#ffc602" size={44}></Icon>
31              }
32            >
33              <Confirm.Header> 确认删除? </Confirm.Header>
34              <Confirm.Content> 确认删除吗? </Confirm.Content>
35            </Confirm>
36          </Backdrop>
37        )
38      );

```

```

39   }
40   /** 点击确认 */
41   confirmDelete = () => {
42     const { mainStore } = this.props;
43     const { isSaving } = mainStore as MainStore;
44     if (!isSaving) {
45       // 调删除接口
46     }
47   };
48 }
49
50 export default DeleteConfirm;

```

index.module.scss

```

1  :global{
2    .br-confirm__content{
3      word-break: break-all;
4    }
5  }
6  .confirm{
7    width: 420px;
8  }

```

interfaces

IData.ts

```

1  export interface IData{
2    id: string;
3    typeId?: string;
4    dataKey: string;
5    dataName: string;
6    mismatchTips: string;
7  }
8

```

stores

index.ts

```

1  import MainStore from './MainStore';
2
3  export default {
4    mainStore: new MainStore(),
5  };

```

MainStore.ts

```

1  // import { Toast } from '@casstime/bricks';
2  import { observable, action } from 'mobx';

```

```
3
4
5 class MainStore {
6   /** 是否正在调保存, 删除接口 */
7   @observable isSaving = false;
8
9   /** 删除弹框是否显示 */
10  @observable confirmVisible = false;
11
12  /** 切换删除弹框显示状态 */
13  @action
14  toggleConfirm = (value: boolean) => {
15    this.confirmVisible = value;
16  };
17 }
18
19 export default MainStore;
```

index.tsx

```
1 import React from 'react';
2 import { Provider } from 'mobx-react';
3 import { RouteConfigComponentProps } from 'react-router-config';
4 import Main from './components/Main';
5 import stores from './stores';
6 import { catchError } from '../.../hoc';
7
8 const Page = (props: RouteConfigComponentProps<void>) => {
9   return (
10     <Provider {...stores}>
11       <Main {...props} />
12     </Provider>
13   );
14 };
15
16 export default catchError(Page);
17
```