

侧边条 Sidebar

```
/**
 * iframe: 1000
 */
import React from 'react'
import { Sidebar,Iconfont } from '@magician/ui'
import { Random } from '@magician/utils'
const iconTypes = [
  'bookmarkLine',
  'yapi',
  'qrcode',
  'barChart',
  'hand',
  'bookmark',
  'shieldCheck',
  'rocket',
  'resource',
  'apollo',
  'archery',
  'add',
  'api',
  'bug',
  'data',
  'database',
  'devops',
  'governance',
  'log',
  'message',
  'tapd',
  'successful',
  'time',
  'nol',
  'gitlab',
  'ump',
  'loki',
  'project',
  'arrowRight',
  'menu2',
]
const { getOneFactroy, genRangeNumber, idGeneratorFactroy, genArr } =
Random
const getIcon = getOneFactroy(iconTypes)
const genId = idGeneratorFactroy()

const data = {
  items: genArr(10, i=>{
    return {
      id: genId(),
      title: '产品' + i,
    }
  })
}
```

```
      icon: <Iconfont type={getIcon()}/>
    }
  )),
  categories: genArr(15, i=> {
    return {
      id: genId(),
      title: '分类' + i,
      items: genArr(genRangeNumber(1, 10), i=>{
        return {
          id: genId(),
          title: '产品' + i,
          icon: <Iconfont type={getIcon()}/>
        }
      })
    }
  })
}
export default () => {
  return (
    <div style={{height: '100vh'}}>
      <Sidebar
        items={data.items}
        categories={data.categories}
      />
    </div>
  )
}
```

API

数据结构

```
interface Item {
  id: number | string
  title: string
  icon?: ReactNode
  [key:string]: any
}

interface Category {
  id: number | string
  title: string
  items: Item[]
}
```

Props

名称	描述	类型	默认值
zIndex	层级	number	10

名称	描述	类型	默认值
height	高度	string number	100vh
items	左侧展示的数据	Item[]	[]

Ref 暴露的方法

名称	描述	参数	返回值
getWrapper	获取瀑布流容器		HTMLDIVElement
refresh	刷新布局		void