# 我们来做一个练习

## 还是reactdemo3项目，在src目录下面新建一个文件夹：exercise-product-list在里面新建一个ManageProduct.jsx组件文件，在App.jsx文件在添加对组件的引用。我们先添加一下数据看看效果

|  |
| --- |
| **//**ManageProduct.jsx  **import *React***, {*useState*} **from 'react'**; **let** initProducts=[  {**id**:1,**name**:**'短袖T恤'**,**price**:50,**count**:10,**sale**:**true**},  {**id**:2,**name**:**'牛仔裤'**,**price**:100,**count**:20,**sale**:**true**},  {**id**:3,**name**:**'羽绒服'**,**price**:300,**count**:0,**sale**:**false**},  {**id**:4,**name**:**'运动服'**,**price**:300,**count**:12,**sale**:**false**}, ] **const** *ManageProduct* = () => {  **let** [products,setProducts] = *useState*(initProducts)  **return** (  <**div**>  <**h3**>ManageProduct</**h3**>  <**div**></**div**>  <**table**>  <**thead**>  <**tr**>  <**th**>编号</**th**>  <**th**>商品名称</**th**>  <**th**>单价</**th**>  <**th**>库存</**th**>  <**th**>状态</**th**>  <**th**>操作</**th**>  </**tr**>  </**thead**>  <**tbody**>  {products.map(item=>{  **return** (  <**tr key=**{item.**id**}>  <**td**>{item.**id**}</**td**>  <**td**>{item.**name**}</**td**>  <**td**>{item.**price**}</**td**>  <**td**>  <**button**>-</**button**>  {item.**count**}  <**button**>+</**button**></**td**>  <**td**></**td**>  <**td**>  <**button**>删除</**button**>  </**td**>  </**tr**>  )  })}  </**tbody**>  </**table**>  <**div**>  <**p**>当前库存0件，共计0元</**p**>  </**div**>  </**div**>  ); };  **export default** *ManageProduct*; |

### 初步效果

|  |
| --- |
|  |

### 状态是根据sale属性的true还是false来决定是否下架还是在销售中，如果库存为0表示买完了

### 我的版本

|  |
| --- |
| **import *React***, {*useState*} **from 'react'**; **let** initProducts=[  {**id**:1,**name**:**'短袖T恤'**,**price**:50,**count**:10,**sale**:**true**},  {**id**:2,**name**:**'牛仔裤'**,**price**:100,**count**:20,**sale**:**true**},  {**id**:3,**name**:**'羽绒服'**,**price**:300,**count**:0,**sale**:**false**},  {**id**:4,**name**:**'运动服'**,**price**:300,**count**:12,**sale**:**false**}, ] **const** *ManageProduct* = () => {  **let** [products,setProducts] = *useState*(initProducts)  **let** status = (item)=>{  **if**(item.**count** >0 ){  **if**(item.**sale**){  **return "在售中"** }  **return "已下架"** } **else**{  **return '已售罄'** }  }  **return** (  <**div**>  <**h3**>ManageProduct</**h3**>  <**div**></**div**>  <**table**>  <**thead**>  <**tr**>  <**th**>编号</**th**>  <**th**>商品名称</**th**>  <**th**>单价</**th**>  <**th**>库存</**th**>  <**th**>状态</**th**>  <**th**>操作</**th**>  </**tr**>  </**thead**>  <**tbody**>  {products.map(item=>{  **return** (  <**tr key=**{item.**id**}>  <**td**>{item.**id**}</**td**>  <**td**>{item.**name**}</**td**>  <**td**>{item.**price**}</**td**>  <**td**>  <**button**>-</**button**>  {item.**count**}  <**button**>+</**button**></**td**>  <**td**>{status(item)}</**td**>  <**td**>  <**button**>删除</**button**>  </**td**>  </**tr**>  )  })}  </**tbody**>  </**table**>  <**div**>  <**p**>当前库存0件，共计0元</**p**>  </**div**>  </**div**>  ); };  **export default** *ManageProduct*; |

### 效果：

|  |
| --- |
|  |

### 我的版本2

|  |
| --- |
| **import *React***, {*useState*} **from 'react'**; **let** initProducts=[  {**id**:1,**name**:**'短袖T恤'**,**price**:50,**count**:10,**sale**:**true**},  {**id**:2,**name**:**'牛仔裤'**,**price**:100,**count**:20,**sale**:**true**},  {**id**:3,**name**:**'羽绒服'**,**price**:300,**count**:0,**sale**:**false**},  {**id**:4,**name**:**'运动服'**,**price**:300,**count**:12,**sale**:**false**}, ] **const** *ManageProduct* = () => {  **let** [products,setProducts] = *useState*(initProducts)  **let** status = (item)=>{  *// if(item.count >0 ){  // if(item.sale){  // return "在售中"  // }  // return "已下架"  // } else{  // return '已售罄'  // }* **if**(item.**count**===0){  **return** <**span style=**{{**color**:**'orange'**}}>已售罄</**span**>  } **else** {  **if**(item.**sale**){  **return** <**span style=**{{**color**:**'blue'**}}>在售中</**span**>  }  **return** <**span style=**{{**color**:**'red'**}}>已下架</**span**>  }  }  **return** (  <**div**>  <**h3**>ManageProduct</**h3**>  <**div**></**div**>  <**table**>  <**thead**>  <**tr**>  <**th**>编号</**th**>  <**th**>商品名称</**th**>  <**th**>单价</**th**>  <**th**>库存</**th**>  <**th**>状态</**th**>  <**th**>操作</**th**>  </**tr**>  </**thead**>  <**tbody**>  {products.map(item=>{  **return** (  <**tr key=**{item.**id**}>  <**td**>{item.**id**}</**td**>  <**td**>{item.**name**}</**td**>  <**td**>{item.**price**}</**td**>  <**td**>  <**button**>-</**button**>  {item.**count**}  <**button**>+</**button**></**td**>  <**td**>{status(item)}</**td**>  <**td**>  <**button**>删除</**button**>  </**td**>  </**tr**>  )  })}  </**tbody**>  </**table**>  <**div**>  <**p**>当前库存0件，共计0元</**p**>  </**div**>  </**div**>  ); };  **export default** *ManageProduct*; |

### 效果：

|  |
| --- |
|  |

### 老师的版本

|  |
| --- |
| **import *React***, {*useState*} **from 'react'**; **let** initProducts=[  {**id**:1,**name**:**'短袖T恤'**,**price**:50,**count**:10,**sale**:**true**},  {**id**:2,**name**:**'牛仔裤'**,**price**:100,**count**:20,**sale**:**true**},  {**id**:3,**name**:**'羽绒服'**,**price**:300,**count**:0,**sale**:**false**},  {**id**:4,**name**:**'运动服'**,**price**:300,**count**:12,**sale**:**false**}, ] **const** *ManageProduct* = () => {  **let** [products,setProducts] = *useState*(initProducts)  **let** status = (item)=>{  *//写法1  // if(item.count >0 ){  // if(item.sale){  // return "在售中"  // }  // return "已下架"  // } else{  // return '已售罄'  // }  //写法2  // if(item.count===0){  // return <span style={{color:'orange'}}>已售罄</span>  // } else {  // if(item.sale){  // return <span style={{color:'blue'}}>在售中</span>  // }  // return <span style={{color:'red'}}>已下架</span>  // }  //写法2* **if**(item.**count**===0){  **return** <**span style=**{{**color**:**'orange'**}}>已售罄</**span**>  }  **if**(item.**sale**){  **return** <**span style=**{{**color**:**'blue'**}}>在售中</**span**>  }  **return** <**span style=**{{**color**:**'red'**}}>已下架</**span**>  }  **return** (  <**div**>  <**h3**>ManageProduct</**h3**>  <**div**></**div**>  <**table**>  <**thead**>  <**tr**>  <**th**>编号</**th**>  <**th**>商品名称</**th**>  <**th**>单价</**th**>  <**th**>库存</**th**>  <**th**>状态</**th**>  <**th**>操作</**th**>  </**tr**>  </**thead**>  <**tbody**>  {products.map(item=>{  **return** (  <**tr key=**{item.**id**}>  <**td**>{item.**id**}</**td**>  <**td**>{item.**name**}</**td**>  <**td**>{item.**price**}</**td**>  <**td**>  <**button**>-</**button**>  {item.**count**}  <**button**>+</**button**></**td**>  <**td**>{status(item)}</**td**>  <**td**>  <**button**>删除</**button**>  </**td**>  </**tr**>  )  })}  </**tbody**>  </**table**>  <**div**>  <**p**>当前库存0件，共计0元</**p**>  </**div**>  </**div**>  ); };  **export default** *ManageProduct*; |

### 效果是一样的，然后我们在操作一栏根据sale的真假来设置‘下架’或者‘上架’按钮还有，如果一个产品没有库存，不能让他上架

|  |
| --- |
| **import *React***, {*useState*} **from 'react'**; **let** initProducts=[  {**id**:1,**name**:**'短袖T恤'**,**price**:50,**count**:10,**sale**:**true**},  {**id**:2,**name**:**'牛仔裤'**,**price**:100,**count**:20,**sale**:**true**},  {**id**:3,**name**:**'羽绒服'**,**price**:300,**count**:0,**sale**:**false**},  {**id**:4,**name**:**'运动服'**,**price**:300,**count**:12,**sale**:**false**}, ] **const** *ManageProduct* = () => {  **let** [products,setProducts] = *useState*(initProducts)  **let** status = (item)=>{  *//写法1  // if(item.count >0 ){  // if(item.sale){  // return "在售中"  // }  // return "已下架"  // } else{  // return '已售罄'  // }  //写法2  // if(item.count===0){  // return <span style={{color:'orange'}}>已售罄</span>  // } else {  // if(item.sale){  // return <span style={{color:'blue'}}>在售中</span>  // }  // return <span style={{color:'red'}}>已下架</span>  // }  //写法3* **if**(item.**count**===0){  **return** <**span style=**{{**color**:**'red'**}}>已售罄</**span**>  }  **if**(item.**sale**){  **return** <**span style=**{{**color**:**'blue'**}}>在售中</**span**>  }  **return** <**span style=**{{**color**:**'orange'**}}>已下架</**span**>  }  *//下架函数* **let** soldOut = (id) =>{  setProducts(products.map(item=>{  **if**(item.**id**==id){  **return** {  ...item,  **sale**:**false** }  }  **return** item  }))  }  *//上架函数* **let** putOnSale = (it)=>{  *//库存为0不能上架* **if**(it.**count**==0){  *alert*(**'库存为0，不能上架，请增加库存'**)  **return** }  setProducts(products.map(item=>{  **if**(item.**id**==it.**id**){  **return** {...item, **sale**:**true**}  }  **return** item  }))  }  **return** (  <**div**>  <**h3**>ManageProduct</**h3**>  <**div**></**div**>  <**table**>  <**thead**>  <**tr**>  <**th**>编号</**th**>  <**th**>商品名称</**th**>  <**th**>单价</**th**>  <**th**>库存</**th**>  <**th**>状态</**th**>  <**th**>操作</**th**>  </**tr**>  </**thead**>  <**tbody**>  {products.map(item=>{  **return** (  <**tr key=**{item.**id**}>  <**td**>{item.**id**}</**td**>  <**td**>{item.**name**}</**td**>  <**td**>{item.**price**}</**td**>  <**td**>  <**button**>-</**button**>  {item.**count**}  <**button**>+</**button**></**td**>  <**td**>{status(item)}</**td**>  <**td**>  {item.**sale**? (<**button onClick=**{()=>soldOut(item.**id**)}>下架</**button**>)  :(<**button onClick=**{()=>putOnSale(item)}>上架</**button**> )}  <**button**>删除</**button**>  </**td**>  </**tr**>  )  })}  </**tbody**>  </**table**>  <**div**>  <**p**>当前库存0件，共计0元</**p**>  </**div**>  </**div**>  ); };  **export default** *ManageProduct*; |

### 添加增加或者减少数量的功能

|  |
| --- |
| **import *React***, {*useState*} **from 'react'**; **let** initProducts=[  {**id**:1,**name**:**'短袖T恤'**,**price**:50,**count**:10,**sale**:**true**},  {**id**:2,**name**:**'牛仔裤'**,**price**:100,**count**:20,**sale**:**true**},  {**id**:3,**name**:**'羽绒服'**,**price**:300,**count**:0,**sale**:**false**},  {**id**:4,**name**:**'运动服'**,**price**:300,**count**:12,**sale**:**false**}, ] **const** *ManageProduct* = () => {  **let** [products,setProducts] = *useState*(initProducts)  **let** status = (item)=>{  *//写法1  // if(item.count >0 ){  // if(item.sale){  // return "在售中"  // }  // return "已下架"  // } else{  // return '已售罄'  // }  //写法3  // if(item.count===0){  // return <span style={{color:'orange'}}>已售罄</span>  // } else {  // if(item.sale){  // return <span style={{color:'blue'}}>在售中</span>  // }  // return <span style={{color:'red'}}>已下架</span>  // }  //写法2* **if**(item.**count**===0){  **return** <**span style=**{{**color**:**'red'**}}>已售罄</**span**>  }  **if**(item.**sale**){  **return** <**span style=**{{**color**:**'blue'**}}>在售中</**span**>  }  **return** <**span style=**{{**color**:**'orange'**}}>已下架</**span**>  }  *//下架函数* **let** soldOut = (id) =>{  setProducts(products.map(item=>{  **if**(item.**id**==id){  **return** {  ...item,  **sale**:**false** }  }  **return** item  }))  }  *//上架函数* **let** putOnSale = (it)=>{  *//库存为0不能上架* **if**(it.**count**==0){  *alert*(**'库存为0，不能上架，请增加库存'**)  **return** }  setProducts(products.map(item=>{  **if**(item.**id**==it.**id**){  **return** {...item, **sale**:**true**}  }  **return** item  }))  }   *//减少库存* **let** decrement=(it)=>{  **if**(it.**count**==0){  *// alert('库存为0，不能减少，请增加库存')* **return** }  setProducts(products.map(item=>{  **if**(item.**id**==it.**id**){  **return** {  ...item,  **count**:item.**count**-1  }  }  **return** item  }))  }  *//增加库存* **let** increment=(id)=>{  setProducts(products.map(item=>{  **if**(item.**id**==id){  **return** {  ...item,  **count**:item.**count**+1  }  }  **return** item  }))  }  **return** (  <**div**>  <**h3**>ManageProduct</**h3**>  <**div**></**div**>  <**table**>  <**thead**>  <**tr**>  <**th**>编号</**th**>  <**th**>商品名称</**th**>  <**th**>单价</**th**>  <**th**>库存</**th**>  <**th**>状态</**th**>  <**th**>操作</**th**>  </**tr**>  </**thead**>  <**tbody**>  {products.map(item=>{  **return** (  <**tr key=**{item.**id**}>  <**td**>{item.**id**}</**td**>  <**td**>{item.**name**}</**td**>  <**td**>{item.**price**}</**td**>  <**td**>  <**button onClick=**{()=>decrement(item)}>-</**button**>  {item.**count**}  <**button onClick=**{()=>increment(item.**id**)}>+</**button**></**td**>  <**td**>{status(item)}</**td**>  <**td**>  {item.**sale**? (<**button onClick=**{()=>soldOut(item.**id**)}>下架</**button**>)  :(<**button onClick=**{()=>putOnSale(item)}>上架</**button**> )}  <**button**>删除</**button**>  </**td**>  </**tr**>  )  })}  </**tbody**>  </**table**>  <**div**>  <**p**>当前库存0件，共计0元</**p**>  </**div**>  </**div**>  ); };  **export default** *ManageProduct*; |

#### 注意：当库存为0的时候，我们不让它继续减少，需要在减少库存的函数里面设置一个判断，如果库存等于0就直接返回

### 添加删除功能

|  |
| --- |
| **import *React***, {*useState*} **from 'react'**; **let** initProducts=[  {**id**:1,**name**:**'短袖T恤'**,**price**:50,**count**:10,**sale**:**true**},  {**id**:2,**name**:**'牛仔裤'**,**price**:100,**count**:20,**sale**:**true**},  {**id**:3,**name**:**'羽绒服'**,**price**:300,**count**:0,**sale**:**false**},  {**id**:4,**name**:**'运动服'**,**price**:300,**count**:12,**sale**:**false**}, ] **const** *ManageProduct* = () => {  **let** [products,setProducts] = *useState*(initProducts)  **let** status = (item)=>{  *//写法1  // if(item.count >0 ){  // if(item.sale){  // return "在售中"  // }  // return "已下架"  // } else{  // return '已售罄'  // }  //写法3  // if(item.count===0){  // return <span style={{color:'orange'}}>已售罄</span>  // } else {  // if(item.sale){  // return <span style={{color:'blue'}}>在售中</span>  // }  // return <span style={{color:'red'}}>已下架</span>  // }  //写法2* **if**(item.**count**===0){  **return** <**span style=**{{**color**:**'red'**}}>已售罄</**span**>  }  **if**(item.**sale**){  **return** <**span style=**{{**color**:**'blue'**}}>在售中</**span**>  }  **return** <**span style=**{{**color**:**'orange'**}}>已下架</**span**>  }  *//下架函数* **let** soldOut = (id) =>{  setProducts(products.map(item=>{  **if**(item.**id**==id){  **return** {  ...item,  **sale**:**false** }  }  **return** item  }))  }  *//上架函数* **let** putOnSale = (it)=>{  *//库存为0不能上架* **if**(it.**count**==0){  *alert*(**'库存为0，不能上架，请增加库存'**)  **return** }  setProducts(products.map(item=>{  **if**(item.**id**==it.**id**){  **return** {...item, **sale**:**true**}  }  **return** item  }))  }   *//减少库存* **let** decrement=(it)=>{  **if**(it.**count**==0){  *// alert('库存为0，不能减少，请增加库存')* **return** }  setProducts(products.map(item=>{  **if**(item.**id**==it.**id**){  **return** {  ...item,  **count**:item.**count**-1  }  }  **return** item  }))  }  *//增加库存* **let** increment=(id)=>{  setProducts(products.map(item=>{  **if**(item.**id**==id){  **return** {  ...item,  **count**:item.**count**+1  }  }  **return** item  }))  }  *//删除货品* **let** remove = (id)=>{  setProducts(products.filter(item=>item.**id** != id))  }  **return** (  <**div**>  <**h3**>ManageProduct</**h3**>  <**div**></**div**>  <**table**>  <**thead**>  <**tr**>  <**th**>编号</**th**>  <**th**>商品名称</**th**>  <**th**>单价</**th**>  <**th**>库存</**th**>  <**th**>状态</**th**>  <**th**>操作</**th**>  </**tr**>  </**thead**>  <**tbody**>  {products.map(item=>{  **return** (  <**tr key=**{item.**id**}>  <**td**>{item.**id**}</**td**>  <**td**>{item.**name**}</**td**>  <**td**>{item.**price**}</**td**>  <**td**>  <**button onClick=**{()=>decrement(item)}>-</**button**>  {item.**count**}  <**button onClick=**{()=>increment(item.**id**)}>+</**button**></**td**>  <**td**>{status(item)}</**td**>  <**td**>  {item.**sale**? (<**button onClick=**{()=>soldOut(item.**id**)}>下架</**button**>)  :(<**button onClick=**{()=>putOnSale(item)}>上架</**button**> )}  <**button onClick=**{()=>remove(item.**id**)}>删除</**button**>  </**td**>  </**tr**>  )  })}  </**tbody**>  </**table**>  <**div**>  <**p**>当前库存0件，共计0元</**p**>  </**div**>  </**div**>  ); };  **export default** *ManageProduct*; |

### 效果：

#### 删除前

|  |
| --- |
|  |

#### 删除后

|  |
| --- |
|  |

### 添加计算总库存和总存货价值的功能，需要使用计算属性

|  |
| --- |
| **import *React***, {*useMemo*, *useState*} **from 'react'**; **let** initProducts=[  {**id**:1,**name**:**'短袖T恤'**,**price**:50,**count**:10,**sale**:**true**},  {**id**:2,**name**:**'牛仔裤'**,**price**:100,**count**:20,**sale**:**true**},  {**id**:3,**name**:**'羽绒服'**,**price**:300,**count**:0,**sale**:**false**},  {**id**:4,**name**:**'运动服'**,**price**:300,**count**:12,**sale**:**false**}, ] **const** *ManageProduct* = () => {  **let** [products,setProducts] = *useState*(initProducts)  *//计算库存的计算属性* **let** totalCount = *useMemo*(()=>{  **return** products.reduce((sum,item)=>sum + item.**count** ,0)  },[products])  *//计算总价的计算属性* **let** totalPrice = *useMemo*(()=>{  **return** products.reduce((sum,item)=>sum + item.**count** \* item.**price** ,0)  },[products])  **let** status = (item)=>{  *//写法1  // if(item.count >0 ){  // if(item.sale){  // return "在售中"  // }  // return "已下架"  // } else{  // return '已售罄'  // }  //写法3  // if(item.count===0){  // return <span style={{color:'orange'}}>已售罄</span>  // } else {  // if(item.sale){  // return <span style={{color:'blue'}}>在售中</span>  // }  // return <span style={{color:'red'}}>已下架</span>  // }  //写法2* **if**(item.**count**===0){  **return** <**span style=**{{**color**:**'red'**}}>已售罄</**span**>  }  **if**(item.**sale**){  **return** <**span style=**{{**color**:**'blue'**}}>在售中</**span**>  }  **return** <**span style=**{{**color**:**'orange'**}}>已下架</**span**>  }  *//下架函数* **let** soldOut = (id) =>{  setProducts(products.map(item=>{  **if**(item.**id**==id){  **return** {  ...item,  **sale**:**false** }  }  **return** item  }))  }  *//上架函数* **let** putOnSale = (it)=>{  *//库存为0不能上架* **if**(it.**count**==0){  *alert*(**'库存为0，不能上架，请增加库存'**)  **return** }  setProducts(products.map(item=>{  **if**(item.**id**==it.**id**){  **return** {...item, **sale**:**true**}  }  **return** item  }))  }   *//减少库存* **let** decrement=(it)=>{  **if**(it.**count**==0){  *// alert('库存为0，不能减少，请增加库存')* **return** }  setProducts(products.map(item=>{  **if**(item.**id**==it.**id**){  **return** {  ...item,  **count**:item.**count**-1  }  }  **return** item  }))  }  *//增加库存* **let** increment=(id)=>{  setProducts(products.map(item=>{  **if**(item.**id**==id){  **return** {  ...item,  **count**:item.**count**+1  }  }  **return** item  }))  }  *//删除货品* **let** remove = (id)=>{  setProducts(products.filter(item=>item.**id** != id))  }  **return** (  <**div**>  <**h3**>ManageProduct</**h3**>  <**div**></**div**>  <**table**>  <**thead**>  <**tr**>  <**th**>编号</**th**>  <**th**>商品名称</**th**>  <**th**>单价</**th**>  <**th**>库存</**th**>  <**th**>状态</**th**>  <**th**>操作</**th**>  </**tr**>  </**thead**>  <**tbody**>  {products.map(item=>{  **return** (  <**tr key=**{item.**id**}>  <**td**>{item.**id**}</**td**>  <**td**>{item.**name**}</**td**>  <**td**>{item.**price**}</**td**>  <**td**>  <**button onClick=**{()=>decrement(item)}>-</**button**>  {item.**count**}  <**button onClick=**{()=>increment(item.**id**)}>+</**button**></**td**>  <**td**>{status(item)}</**td**>  <**td**>  {item.**sale**? (<**button onClick=**{()=>soldOut(item.**id**)}>下架</**button**>)  :(<**button onClick=**{()=>putOnSale(item)}>上架</**button**> )}  <**button onClick=**{()=>remove(item.**id**)}>删除</**button**>  </**td**>  </**tr**>  )  })}  </**tbody**>  </**table**>  <**div**>  <**p**>当前库存{totalCount}件，共计{totalPrice}元</**p**>  </**div**>  </**div**>  ); };  **export default** *ManageProduct*; |

### 效果

|  |
| --- |
|  |
|  |
|  |

### 需要好好学习数组的reduce方法

## 我们再添加一下功能，我们添加一排链接点击每一个链接会过滤出对应的数据，需要使用计算属性，而且需要过滤数组得到一个新数组，用新数组在渲染的地方代替原数组，需要把运行计算属性也替换了

|  |
| --- |
| **import *React***, {*useMemo*, *useState*} **from 'react'**; **let** initProducts=[  {**id**:1,**name**:**'短袖T恤'**,**price**:50,**count**:10,**sale**:**true**},  {**id**:2,**name**:**'牛仔裤'**,**price**:100,**count**:20,**sale**:**true**},  {**id**:3,**name**:**'羽绒服'**,**price**:300,**count**:0,**sale**:**false**},  {**id**:4,**name**:**'运动服'**,**price**:300,**count**:12,**sale**:**false**}, ] **let** initFilters=[  {**id**:1,**title**:**'全部'**},  {**id**:2,**title**:**'在售中'**},  {**id**:3,**title**:**'已下架'**},  {**id**:4,**title**:**'已售罄'**}, ] **const** *ManageProduct* = () => {  **let** [products,setProducts] = *useState*(initProducts)  **let** [filters,setFilters] = *useState*(initFilters)  *//默认选中项* **let** [filterId,setFilterId] = *useState*(1)  *//获取符合条件的数据的计算属性* **let** filterProducts = *useMemo*(()=>{  **switch** (filterId) {  **case** 2:  **return** products.filter(item=>item.**sale**)  **break  case** 3:  **return** products.filter(item=>!item.**sale** && item.**count** !=0)  **break  case** 4:  **return** products.filter(item=>{  **return** item.**count**==0  })  **break  default**:  **return** products  **break** }  },[products,filterId])  *//计算库存的计算属性  // let totalCount = useMemo(()=>{ //这个已经没有用了  // return products.reduce((sum,item)=>sum + item.count ,0)  // },[products])* **let** totalCount = *useMemo*(()=>{  **return** filterProducts.reduce((sum,item)=>sum + item.**count** ,0)  },[filterProducts])   *//计算总价的计算属性  // let totalPrice = useMemo(()=>{//这个已经没有用了  // return products.reduce((sum,item)=>sum + item.count \* item.price ,0)  // },[products])* **let** totalPrice = *useMemo*(()=>{  **return** filterProducts.reduce((sum,item)=>sum + item.**count** \* item.**price** ,0)  },[filterProducts])   **let** status = (item)=>{  *//写法1  // if(item.count >0 ){  // if(item.sale){  // return "在售中"  // }  // return "已下架"  // } else{  // return '已售罄'  // }  //写法3  // if(item.count===0){  // return <span style={{color:'orange'}}>已售罄</span>  // } else {  // if(item.sale){  // return <span style={{color:'blue'}}>在售中</span>  // }  // return <span style={{color:'red'}}>已下架</span>  // }  //写法2* **if**(item.**count**===0){  **return** <**span style=**{{**color**:**'red'**}}>已售罄</**span**>  }  **if**(item.**sale**){  **return** <**span style=**{{**color**:**'blue'**}}>在售中</**span**>  }  **return** <**span style=**{{**color**:**'orange'**}}>已下架</**span**>  }  *//下架函数* **let** soldOut = (id) =>{  setProducts(products.map(item=>{  **if**(item.**id**==id){  **return** {  ...item,  **sale**:**false** }  }  **return** item  }))  }  *//上架函数* **let** putOnSale = (it)=>{  *//库存为0不能上架* **if**(it.**count**==0){  *alert*(**'库存为0，不能上架，请增加库存'**)  **return** }  setProducts(products.map(item=>{  **if**(item.**id**==it.**id**){  **return** {...item, **sale**:**true**}  }  **return** item  }))  }   *//减少库存* **let** decrement=(it)=>{  **if**(it.**count**==0){  *// alert('库存为0，不能减少，请增加库存')* **return** }  setProducts(products.map(item=>{  **if**(item.**id**==it.**id**){  **return** {  ...item,  **count**:item.**count**-1  }  }  **return** item  }))  }  *//增加库存* **let** increment=(id)=>{  setProducts(products.map(item=>{  **if**(item.**id**==id){  **return** {  ...item,  **count**:item.**count**+1  }  }  **return** item  }))  }  *//删除货品* **let** remove = (id)=>{  setProducts(products.filter(item=>item.**id** != id))  }  **return** (  <**div**>  <**h3**>ManageProduct</**h3**>  <**div**>  {filters.map(item=><**span style=**{{**backgroundColor**:item.**id**==filterId?**'#bbb'**:**'transparent'**}}  **key=**{item.**id**} **onClick=**{()=>setFilterId(item.**id**)}>  {item.**title**} </**span**>)}  </**div**>  <**table**>  <**thead**>  <**tr**>  <**th**>编号</**th**>  <**th**>商品名称</**th**>  <**th**>单价</**th**>  <**th**>库存</**th**>  <**th**>状态</**th**>  <**th**>操作</**th**>  </**tr**>  </**thead**>  <**tbody**>  {  filterProducts.map(item=>{  **return** (  <**tr key=**{item.**id**}>  <**td**>{item.**id**}</**td**>  <**td**>{item.**name**}</**td**>  <**td**>{item.**price**}</**td**>  <**td**>  <**button onClick=**{()=>decrement(item)}>-</**button**>  {item.**count**}  <**button onClick=**{()=>increment(item.**id**)}>+</**button**></**td**>  <**td**>{status(item)}</**td**>  <**td**>  {item.**sale**? (<**button onClick=**{()=>soldOut(item.**id**)}>下架</**button**>)  :(<**button onClick=**{()=>putOnSale(item)}>上架</**button**> )}  <**button onClick=**{()=>remove(item.**id**)}>删除</**button**>  </**td**>  </**tr**>  )  })}  </**tbody**>  </**table**>  <**div**>  <**p**>当前库存{totalCount}件，共计{totalPrice}元</**p**>  </**div**>  </**div**>  ); };  **export default** *ManageProduct*; |

## 练习的使用功能完成，思考一下，这个练习的代码有什么地方可以优化？例如把两个方法合并为一个方法？