

zh-CN

使用自定义元素，我们可以集成 [react-dnd](#) 来实现拖拽排序。

en-US

By using `components`, we can integrate table with [react-dnd](#) to implement drag sorting function.

```
import React, { useState, useCallback, useRef } from 'react';
import { Table } from 'antd';
import { DndProvider, useDrag, useDrop } from 'react-dnd';
import { HTML5Backend } from 'react-dnd-html5-backend';
import update from 'immutability-helper';

const type = 'DraggableBodyRow';

const DraggableBodyRow = ({ index, moveRow, className, style, ...restProps }) => {
  const ref = useRef();
  const [{ isOver, dropClassName }, drop] = useDrop({
    accept: type,
    collect: monitor => {
      const { index: dragIndex } = monitor.getItem() || {};
      if (dragIndex === index) {
        return {};
      }
      return {
        isOver: monitor.isOver(),
        dropClassName: dragIndex < index ? ' drop-over-downward' : ' drop-over-upward',
      };
    },
    drop: item => {
      moveRow(item.index, index);
    },
  });
  const [, drag] = useDrag({
    type,
    item: { index },
    collect: monitor => ({
      isDragging: monitor.isDragging(),
    }),
  });
  drop(drag(ref));

  return (
    <tr
      ref={ref}
      className={` ${className} ${isOver ? dropClassName : ''} `}
      style={{ cursor: 'move', ...style }}
      {...restProps}
    />
  );
}
```

```

    );
  };

const columns = [
  {
    title: 'Name',
    dataIndex: 'name',
    key: 'name',
  },
  {
    title: 'Age',
    dataIndex: 'age',
    key: 'age',
  },
  {
    title: 'Address',
    dataIndex: 'address',
    key: 'address',
  },
];

const DragSortingTable: React.FC = () => {
  const [data, setData] = useState([
    {
      key: '1',
      name: 'John Brown',
      age: 32,
      address: 'New York No. 1 Lake Park',
    },
    {
      key: '2',
      name: 'Jim Green',
      age: 42,
      address: 'London No. 1 Lake Park',
    },
    {
      key: '3',
      name: 'Joe Black',
      age: 32,
      address: 'Sidney No. 1 Lake Park',
    },
  ]);

  const components = {
    body: {
      row: DraggableBodyRow,
    },
  };

  const moveRow = useCallback(
    (dragIndex, hoverIndex) => {
      const dragRow = data[dragIndex];

```

```

    setData(
      update(data, {
        $splice: [
          [dragIndex, 1],
          [hoverIndex, 0, dragRow],
        ],
      }),
    );
  },
  [data],
);

return (
  <DndProvider backend={HTML5Backend}>
    <Table
      columns={columns}
      dataSource={data}
      components={components}
      onRow={(record, index) => ({
        index,
        moveRow,
      })}
    />
  </DndProvider>
);
};

export default () => <DragSortingTable />;

```

```

#components-table-demo-drag-sorting tr.drop-over-downward td {
  border-bottom: 2px dashed #1890ff;
}

#components-table-demo-drag-sorting tr.drop-over-upward td {
  border-top: 2px dashed #1890ff;
}

```