zh-CN

通过 `oneWay` 将 Transfer 转为单向样式。

en-US

Use `oneWay` to makes Transfer to one way style.

```
import { Transfer, Switch } from 'antd';

const mockData = [];
for (let i = 0; i < 20; i++) {
  mockData.push({
    key: i.toString(),
    title: `content${i + 1}`,
    description: `description of content${i + 1}`,
    disabled: i % 3 < 1,
  });
}

const oriTargetKeys = mockData.filter(item => +item.key % 3 > 1).map(item =>
item.key);

class App extends React.Component {
  state = {
    targetKeys: oriTargetKeys,
    selectedKeys: [],
    disabled: false,
  };

  handleChange = (nextTargetKeys, direction, moveKeys) => {
    this.setState({ targetKeys: nextTargetKeys });

    console.log('targetKeys: ', nextTargetKeys);
    console.log('direction: ', direction);
    console.log('moveKeys: ', moveKeys);
  };

  handleSelectChange = (sourceSelectedKeys, targetSelectedKeys) => {
    this.setState({ selectedKeys: [...sourceSelectedKeys, ...targetSelectedKeys] });

    console.log('sourceSelectedKeys: ', sourceSelectedKeys);
    console.log('targetSelectedKeys: ', targetSelectedKeys);
  };

  handleScroll = (direction, e) => {
    console.log('direction:', direction);
    console.log('target:', e.target);
  };

  handleDisable = disabled => {
```

```jsx
      this.setState({ disabled });
    };

    render() {
      const { targetKeys, selectedKeys, disabled } = this.state;
      return (
        <>
          <Transfer
            dataSource={mockData}
            titles={['Source', 'Target']}
            targetKeys={targetKeys}
            selectedKeys={selectedKeys}
            onChange={this.handleChange}
            onSelectChange={this.handleSelectChange}
            onScroll={this.handleScroll}
            render={item => item.title}
            disabled={disabled}
            oneWay
            style={{ marginBottom: 16 }}
          />
          <Switch
            unCheckedChildren="disabled"
            checkedChildren="disabled"
            checked={disabled}
            onChange={this.handleDisable}
          />
        </>
      );
    }
}

export default () => <App />;
```