zh-CN

点击展开节点，动态加载数据。

en-US

To load data asynchronously when click to expand a treeNode.

```
import React, { useState } from 'react';
import { Tree } from 'antd';

interface DataNode {
  title: string;
  key: string;
  isLeaf?: boolean;
  children?: DataNode[];
}

const initTreeData: DataNode[] = [
  { title: 'Expand to load', key: '0' },
  { title: 'Expand to load', key: '1' },
  { title: 'Tree Node', key: '2', isLeaf: true },
];

// It's just a simple demo. You can use tree map to optimize update perf.
function updateTreeData(list: DataNode[], key: React.Key, children: DataNode[]):
DataNode[] {
  return list.map(node => {
    if (node.key === key) {
      return {
        ...node,
        children,
      };
    }
    if (node.children) {
      return {
        ...node,
        children: updateTreeData(node.children, key, children),
      };
    }
    return node;
  });
}

const Demo: React.FC<{}> = () => {
  const [treeData, setTreeData] = useState(initTreeData);

  const onLoadData = ({ key, children }: any) =>
    new Promise<void>(resolve => {
      if (children) {
        resolve();
        return;
      }
      setTimeout(() => {
        setTreeData(origin =>
          updateTreeData(origin, key, [
            { title: 'Child Node', key: `${key}-0` },
            { title: 'Child Node', key: `${key}-1` },
          ]),
        );

        resolve();
      }, 1000);
    });

  return <Tree loadData={onLoadData} treeData={treeData} />;
};

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