## When To Use

- When the system has more than two layers in a hierarchy.
- When you need to inform the user of where they are.
- When the user may need to navigate back to a higher level.

```
// works when >=5.3.0, recommended ✅
return <Breadcrumb items={[{ title: 'sample' }]} />;

// works when <5.3.0, deprecated when >=5.3.0 🙍‍♀️
return (
  <Breadcrumb>
    <Breadcrumb.Item>sample</Breadcrumb.Item>
  </Breadcrumb>
);

// or

return <Breadcrumb routes={[{ breadcrumbName: 'sample' }]} />;
```

## Examples

### Basic Usage

```
import React from 'react';
import { Breadcrumb } from 'antd';

const App: React.FC = () => (
  <Breadcrumb
    items={[
      {
        title: 'Home',
      },
      {
        title: <a href="">Application Center</a>,
      },
      {
        title: <a href="">Application List</a>,
      },
      {
        title: 'An Application',
      },
    ]}
  />
);
```

```
export default App;
```

**With an Icon**

```
import React from 'react';
import { HomeOutlined, UserOutlined } from '@ant-design/icons';
import { Breadcrumb } from 'antd';

const App: React.FC = () => (
  <Breadcrumb
    items={[
      {
        href: '',
        title: <HomeOutlined />,
      },
      {
        href: '',
        title: (
          <>
            <UserOutlined />
            <span>Application List</span>
          </>
        ),
      },
      {
        title: 'Application',
      },
    ]}
  />
);

export default App;
```

**With Params**

```
import React from 'react';
import { Breadcrumb } from 'antd';

const App: React.FC = () => (
  <Breadcrumb
    items={[
      {
        title: 'Users',
      },
```

```
      {
        title: ':id',
        href: '',
      },
    ]}
    params={{ id: 1 }}
  />
);

export default App;
```

## Configuring the Separator

```
import React from 'react';
import { Breadcrumb } from 'antd';

const App: React.FC = () => (
  <Breadcrumb
    separator=">"
    items={[
      {
        title: 'Home',
      },
      {
        title: 'Application Center',
        href: '',
      },
      {
        title: 'Application List',
        href: '',
      },
      {
        title: 'An Application',
      },
    ]}
  />
);

export default App;
```

## Bread crumbs with drop down menu

```
import React from 'react';
import { Breadcrumb } from 'antd';
```

```
const menuItems = [
  {
    key: '1',
    label: (
      <a target="_blank" rel="noopener noreferrer"
href="http://www.alipay.com/">
        General
      </a>
    ),
  },
  {
    key: '2',
    label: (
      <a target="_blank" rel="noopener noreferrer"
href="http://www.taobao.com/">
        Layout
      </a>
    ),
  },
  {
    key: '3',
    label: (
      <a target="_blank" rel="noopener noreferrer"
href="http://www.tmall.com/">
        Navigation
      </a>
    ),
  },
];

const App: React.FC = () => (
  <Breadcrumb
    items={[
      {
        title: 'Ant Design',
      },
      {
        title: <a href="">Component</a>,
      },
      {
        title: <a href="">General</a>,
        menu: { items: menuItems },
      },
      {
        title: 'Button',
      },
```

```
    ]}
  />
);

export default App;
```

**Configuring the Separator Independently**

```
import React from 'react';
import { Breadcrumb } from 'antd';

const App: React.FC = () => (
  <Breadcrumb
    separator=""
    items={[
      {
        title: 'Location',
      },
      {
        type: 'separator',
        separator: ':',
      },
      {
        href: '',
        title: 'Application Center',
      },
      {
        type: 'separator',
      },
      {
        href: '',
        title: 'Application List',
      },
      {
        type: 'separator',
      },
      {
        title: 'An Application',
      },
    ]}
  />
);

export default App;
```

**Debug Routes**

```jsx
import React from 'react';
import { Breadcrumb } from 'antd';

export default () => (
  <Breadcrumb
    routes={[
      {
        path: '/home',
        breadcrumbName: 'Home',
      },
      {
        path: '/user',
        breadcrumbName: 'User',
        children: [
          {
            path: '/user1',
            breadcrumbName: 'User1',
          },
          {
            path: '/user2',
            breadcrumbName: 'User2',
          },
        ],
      },
    ]}
  />
);
```

**Component Token**

Debug

```jsx
import React from 'react';
import { HomeOutlined, UserOutlined } from '@ant-design/icons';
import { Breadcrumb, ConfigProvider } from 'antd';

const menuItems = [
  {
    key: '1',
    label: (
      <a target="_blank" rel="noopener noreferrer" href="http://www.alipay.com/">
        General
      </a>
```

```
      ),
    },
    {
      key: '2',
      label: (
        <a target="_blank" rel="noopener noreferrer"
href="http://www.taobao.com/">
          Layout
        </a>
      ),
    },
    {
      key: '3',
      label: (
        <a target="_blank" rel="noopener noreferrer"
href="http://www.tmall.com/">
          Navigation
        </a>
      ),
    },
];
export default () => (
  <ConfigProvider
    theme={{
      components: {
        Breadcrumb: {
          itemColor: '#b02121',
          lastItemColor: '#0f3a88',
          iconFontSize: 28,
          linkColor: '#979a42',
          linkHoverColor: '#9450c0',
          separatorColor: '#b41b60',
          separatorMargin: 22,
        },
      },
    }}
  >
    <Breadcrumb
      separator=">"
      items={[
        {
          title: 'Home',
        },
        {
          title: <a href="">Application Center</a>,
        },
```

```
      {
        title: <a href="">General</a>,
        menu: { items: menuItems },
      },
      {
        title: 'Application Center',
        href: '',
      },
      {
        href: '',
        title: <HomeOutlined />,
      },
      {
        href: '',
        title: (
          <>
            <UserOutlined />
            <span>Application List</span>
          </>
        ),
      },
    ]}
  />
  </ConfigProvider>
);
```

## API

Common props ref： [Common props](#)

### Breadcrumb

| Property | Description | Type | Default | Version |
|----------|-------------|------|---------|---------|
| itemRender | Custom item renderer | (route, params, routes, paths) => ReactNode | - | |
| params | Routing parameters | object | - | |
| items | The routing stack information of router | [ItemType[]](#) | - | 5.3.0 |
| separator | Custom separator | ReactNode | / | |

### ItemType

*type ItemType = Omit<[RouteItemType](#), 'title' | 'path'> | [SeparatorType](#)*

**RouteItemType**

| Property | Description | Type | Default | Version |
|---|---|---|---|---|
| className | The additional css class | string | - | |
| dropdownProps | The dropdown props | [Dropdown](#) | - | |
| href | Target of hyperlink. Can not work with `path` | string | - | |
| path | Connected path. Each path will connect with prev one. Can not work with `href` | string | - | |
| menu | The menu props | [MenuProps](#) | - | 4.24.0 |
| onClick | Set the handler to handle click event | (e:MouseEvent) => void | - | |
| title | item name | ReactNode | - | |

**SeparatorType**

```
const item = {
  type: 'separator', // Must have
  separator: '/',
};
```

| Property | Description | Type | Default | Version |
|---|---|---|---|---|
| type | Mark as separator | `separator` | | 5.3.0 |
| separator | Custom separator | ReactNode | / | 5.3.0 |

**Use with browserHistory**

The link of Breadcrumb item targets `#` by default, you can use `itemRender` to make a `browserHistory` Link.

```
import { Link } from 'react-router';

const items = [
  {
    path: '/index',
    title: 'home',
  },
  {
    path: '/first',
```

```
      title: 'first',
      children: [
        {
          path: '/general',
          title: 'General',
        },
        {
          path: '/layout',
          title: 'Layout',
        },
        {
          path: '/navigation',
          title: 'Navigation',
        },
      ],
    },
    {
      path: '/second',
      title: 'second',
    },
];

function itemRender(currentRoute, params, items, paths) {
  const isLast = currentRoute?.path === items[items.length - 1]?.path;

  return isLast ? (
    <span>{currentRoute.title}</span>
  ) : (
    <Link to={`/${paths.join("/")}`}>{currentRoute.title}</Link>
  );
}

return <Breadcrumb itemRender={itemRender} items={items} />;
```

## Design Token