

When To Use

For displaying anchor hyperlinks on page and jumping between them.

Notes for developers

After version `4.24.0`, we rewrite `Anchor` use `FC`, Some methods of obtaining `ref` and calling internal instance methods will invalid.

Examples

Basic

```
import React from 'react';
import { Anchor, Col, Row } from 'antd';

const App: React.FC = () => (
  <Row>
    <Col span={16}>
      <div id="part-1" style={{ height: '100vh', background:
'rgba(255,0,0,0.02)' }} />
      <div id="part-2" style={{ height: '100vh', background:
'rgba(0,255,0,0.02)' }} />
      <div id="part-3" style={{ height: '100vh', background:
'rgba(0,0,255,0.02)' }} />
    </Col>
    <Col span={8}>
      <Anchor
        items={[
          {
            key: 'part-1',
            href: '#part-1',
            title: 'Part 1',
          },
          {
            key: 'part-2',
            href: '#part-2',
            title: 'Part 2',
          },
          {
            key: 'part-3',
            href: '#part-3',
            title: 'Part 3',
          },
        ]}
      />
    </Col>
  </Row>
);
```

```

    </Row>
  );

  export default App;

```

Horizontal Anchor

```

import React from 'react';
import { Anchor } from 'antd';

const App: React.FC = () => (
  <>
    <div style={{ padding: '20px' }}>
      <Anchor
        direction="horizontal"
        items={[
          {
            key: 'part-1',
            href: '#part-1',
            title: 'Part 1',
          },
          {
            key: 'part-2',
            href: '#part-2',
            title: 'Part 2',
          },
          {
            key: 'part-3',
            href: '#part-3',
            title: 'Part 3',
          },
        ]}
      />
    </div>
    <div>
      <div
        id="part-1"
        style={{
          width: '100vw',
          height: '100vh',
          textAlign: 'center',
          background: 'rgba(0,255,0,0.02)',
        }}
      />
      <div
        id="part-2"

```

```

        style={{
          width: '100vw',
          height: '100vh',
          textAlign: 'center',
          background: 'rgba(0,0,255,0.02)',
        }}
      />
      <div
        id="part-3"
        style={{ width: '100vw', height: '100vh', textAlign: 'center',
background: '#FFBE9' }}
      />
    </div>
  </>
);

export default App;

```

Static Anchor

```

import React from 'react';
import { Anchor } from 'antd';

const App: React.FC = () => (
  <Anchor
    affix={false}
    items={[
      {
        key: '1',
        href: '#anchor-demo-basic',
        title: 'Basic demo',
      },
      {
        key: '2',
        href: '#anchor-demo-static',
        title: 'Static demo',
      },
      {
        key: '3',
        href: '#api',
        title: 'API',
        children: [
          {
            key: '4',
            href: '#anchor-props',
            title: 'Anchor Props',

```

```

    },
    {
      key: '5',
      href: '#link-props',
      title: 'Link Props',
    },
  ],
},
]
  ]}
/>
);

export default App;

```

Customize the onClick event

```

import React from 'react';
import { Anchor } from 'antd';

const handleClick = (
  e: React.MouseEvent<HTMLElement>,
  link: {
    title: React.ReactNode;
    href: string;
  },
) => {
  e.preventDefault();
  console.log(link);
};

const App: React.FC = () => (
  <Anchor
    affix={false}
    onClick={handleClick}
    items={[
      {
        key: '1',
        href: '#anchor-demo-basic',
        title: 'Basic demo',
      },
      {
        key: '2',
        href: '#anchor-demo-static',
        title: 'Static demo',
      },
    ]}
  >

```

```

        key: '3',
        href: '#api',
        title: 'API',
        children: [
          {
            key: '4',
            href: '#anchor-props',
            title: 'Anchor Props',
          },
          {
            key: '5',
            href: '#link-props',
            title: 'Link Props',
          },
        ],
      },
    ],
  }
}
/>
);

export default App;

```

Customize the anchor highlight

```

import React from 'react';
import { Anchor } from 'antd';

const getCurrentAnchor = () => '#anchor-demo-static';

const App: React.FC = () => (
  <Anchor
    affix={false}
    getCurrentAnchor={getCurrentAnchor}
    items={[
      {
        key: '1',
        href: '#anchor-demo-basic',
        title: 'Basic demo',
      },
      {
        key: '2',
        href: '#anchor-demo-static',
        title: 'Static demo',
      },
      {
        key: '3',

```

```

      href: '#api',
      title: 'API',
      children: [
        {
          key: '4',
          href: '#anchor-props',
          title: 'Anchor Props',
        },
        {
          key: '5',
          href: '#link-props',
          title: 'Link Props',
        },
      ],
    },
  ]
}
/>
);

export default App;

```

Set Anchor scroll offset

```

import React, { useEffect, useState } from 'react';
import { Anchor, Col, Row } from 'antd';

const style: React.CSSProperties = {
  height: '30vh',
  backgroundColor: 'rgba(0, 0, 0, 0.85)',
  position: 'fixed',
  top: 0,
  insetInlineStart: 0,
  width: '75%',
  color: '#fff',
};

const App: React.FC = () => {
  const topRef = React.useRef<HTMLDivElement>(null);
  const [targetOffset, setTargetOffset] = useState<number>();

  useEffect(() => {
    setTargetOffset(topRef.current?.clientHeight);
  }, []);

  return (
    <div>

```

```

    <Row>
      <Col span={18}>
        <div
          id="part-1"
          style={{ height: '100vh', background: 'rgba(255,0,0,0.02)',
marginTop: '30vh' }}
        >
          Part 1
        </div>
        <div id="part-2" style={{ height: '100vh', background:
'rgba(0,255,0,0.02)' }}>
          Part 2
        </div>
        <div id="part-3" style={{ height: '100vh', background:
'rgba(0,0,255,0.02)' }}>
          Part 3
        </div>
      </Col>
      <Col span={6}>
        <Anchor
          targetOffset={targetOffset}
          items={[
            { key: 'part-1', href: '#part-1', title: 'Part 1' },
            { key: 'part-2', href: '#part-2', title: 'Part 2' },
            { key: 'part-3', href: '#part-3', title: 'Part 3' },
          ]}
        />
      </Col>
    </Row>
    <div style={style} ref={topRef}>
      <div>Fixed Top Block</div>
    </div>
  </div>
);
};

export default App;

```

Listening for anchor link change

```

import React from 'react';
import { Anchor } from 'antd';

const onChange = (link: string) => {
  console.log('Anchor:OnChange', link);
};

```

```

const App: React.FC = () => (
  <Anchor
    affix={false}
    onChange={onChange}
    items={[
      {
        key: '1',
        href: '#anchor-demo-basic',
        title: 'Basic demo',
      },
      {
        key: '2',
        href: '#anchor-demo-static',
        title: 'Static demo',
      },
      {
        key: '3',
        href: '#api',
        title: 'API',
        children: [
          {
            key: '4',
            href: '#anchor-props',
            title: 'Anchor Props',
          },
          {
            key: '5',
            href: '#link-props',
            title: 'Link Props',
          },
        ],
      },
    ]},
  </Anchor>
);

export default App;

```

Replace href in history

```

import React from 'react';
import { Anchor, Col, Row } from 'antd';

const App: React.FC = () => (
  <Row>

```



```

    <Col span={16}>
      <div id="part-1" style={{ height: '100vh', background:
'rgba(255,0,0,0.02)' }} />
      <div id="part-2" style={{ height: '100vh', background:
'rgba(0,255,0,0.02)' }} />
      <div id="part-3" style={{ height: '100vh', background:
'rgba(0,0,255,0.02)' }} />
    </Col>
    <Col span={8}>
      <Anchor
        replace
        items={[
          {
            key: 'part-1',
            href: '#part-1',
            title: 'Part 1',
          },
          {
            key: 'part-2',
            href: '#part-2',
            title: 'Part 2',
          },
          {
            key: 'part-3',
            href: '#part-3',
            title: 'Part 3',
          },
        ]}
      />
    </Col>
  </Row>
);

export default App;

```

Deprecated JSX demo

Debug

```

import React from 'react';
import { Anchor } from 'antd';

const { Link } = Anchor;

const App: React.FC = () => (
  <Anchor affix={false}>

```

```

    <Link href="#anchor-demo-basic" title="Basic demo" />
    <Link href="#anchor-demo-static" title="Static demo" />
    <Link href="#api" title="API">
      <Link href="#anchor-props" title="Anchor Props" />
      <Link href="#link-props" title="Link Props" />
    </Link>
  </Anchor>
);

export default App;

```

Component Token

Debug

```

import React from 'react';
import { Anchor, Col, ConfigProvider, Row } from 'antd';

/** Test usage. Do not use in your production. */

export default () => (
  <ConfigProvider
    theme={{
      components: {
        Anchor: {
          linkPaddingBlock: 100,
          linkPaddingInlineStart: 50,
        },
      },
    }}
  >
    <Row>
      <Col span={16}>
        <div id="part-1" style={{ height: '100vh', background:
'rgba(255,0,0,0.02)' }} />
        <div id="part-2" style={{ height: '100vh', background:
'rgba(0,255,0,0.02)' }} />
        <div id="part-3" style={{ height: '100vh', background:
'rgba(0,0,255,0.02)' }} />
      </Col>
      <Col span={8}>
        <Anchor
          items={[
            {
              key: 'part-1',
              href: '#part-1',

```

```

        title: 'Part 1',
      },
      {
        key: 'part-2',
        href: '#part-2',
        title: 'Part 2',
      },
      {
        key: 'part-3',
        href: '#part-3',
        title: 'Part 3',
      },
    ],
  ]}
  />
</Col>
</Row>
</ConfigProvider>
);

```

API

Common props ref: [Common props](#)

Anchor Props

Property	Description	Type	Default	Version
affix	Fixed mode of Anchor	boolean Omit<AffixProps, 'offsetTop' 'target' 'children'>	true	object: 5.19.0
bounds	Bounding distance of anchor area	number	5	
getContainer	Scrolling container	() => HTMLElement	() => window	
getCurrentAnchor	Customize the anchor highlight	(activeLink: string) => string	-	
offsetTop	Pixels to offset from top when calculating position of scroll	number	0	

showInkInFixed	Whether show ink-square when affix={false}	boolean	false	
targetOffset	Anchor scroll offset, default as offsetTop, example	number	-	
onChange	Listening for anchor link change	(currentActiveLink: string) => void		
onClick	Set the handler to handle click event	(e: MouseEvent, link: object) => void	-	
items	Data configuration option content, support nesting through children	{ key, href, title, target, children }[] see	-	5.1.0
direction	Set Anchor direction	vertical horizontal	vertical	5.2.0
replace	Replace items' href in browser history instead of pushing it	boolean	false	5.7.0

AnchorItem

Property	Description	Type	Default	Version
key	The unique identifier of the Anchor Link	string number	-	
href	The target of hyperlink	string		
target	Specifies where to display the linked URL	string		
title	The content of hyperlink	ReactNode		
children	Nested Anchor Link, Attention: This attribute does not support horizontal orientation	AnchorItem []	-	
replace	Replace item href in browser history instead of pushing it	boolean	false	5.7.0

Link Props

We recommend using the items form instead.

Property	Description	Type	Default	Version
href	The target of hyperlink	string		
target	Specifies where to display the linked URL	string		
title	The content of hyperlink	ReactNode		

Design Token