

## Specification

### Size

The first level navigation is left aligned near a logo, and the secondary menu is right aligned.

- Top Navigation: the height of the first level navigation is `64px` , and the second level navigation is `48px` .
- Top Navigation (for landing pages): the height of the first level navigation is `80px` , and the second level navigation is `56px` .
- Calculation formula of a top navigation:  $48+8n$  .
- Calculation formula of an aside navigation:  $200+8n$  .

### Interaction rules

- The first level navigation and the last level navigation should be distinguishable by visualization;
- The current item should have the highest priority of visualization;
- When the current navigation item is collapsed, the style of the current navigation item is applied to its parent level;
- The left side navigation bar has support for both the accordion and expanding styles; you can choose the one that fits your case the best.

## Visualization rules

Style of a navigation should conform to its level.

- **Emphasis by colorblock**

When the background color is a deep color, you can use this pattern for the parent level navigation item of the current page.

- **The highlight match stick**

When the background color is a light color, you can use this pattern for the current page navigation item; we recommend using it for the last item of the navigation path.

- **Highlighted font**

From the visualization aspect, a highlighted font is stronger than colorblock; this pattern is often used for the parent level of the current item.

- **Enlarge the size of the font**

`12px` , `14px` is a standard font size of navigation's, `14px` is used for the first and the second level of the navigation. You can choose an appropriate font size regarding the level of your navigation.

## Component Overview

- `Layout` : The layout wrapper, in which `Header` `Sider` `Content` `Footer` or `Layout` itself can be nested, and can be placed in any parent container.

- **Header** : The top layout with the default style, in which any element can be nested, and must be placed in **Layout** .
- **Sider** : The sidebar with default style and basic functions, in which any element can be nested, and must be placed in **Layout** .
- **Content** : The content layout with the default style, in which any element can be nested, and must be placed in **Layout** .
- **Footer** : The bottom layout with the default style, in which any element can be nested, and must be placed in **Layout** .

Based on **flex layout** , please pay attention to the [compatibility](#).

## Examples

### Basic Structure

```
import React from 'react';
import { Flex, Layout } from 'antd';

const { Header, Footer, Sider, Content } = Layout;

const headerStyle: React.CSSProperties = {
  textAlign: 'center',
  color: '#fff',
  height: 64,
  paddingInline: 48,
  lineHeight: '64px',
  backgroundColor: '#4096ff',
};

const contentStyle: React.CSSProperties = {
  textAlign: 'center',
  minHeight: 120,
  lineHeight: '120px',
  color: '#fff',
  backgroundColor: '#0958d9',
};

const siderStyle: React.CSSProperties = {
  textAlign: 'center',
  lineHeight: '120px',
  color: '#fff',
  backgroundColor: '#1677ff',
};

const footerStyle: React.CSSProperties = {
  textAlign: 'center',
```

```

    color: '#fff',
    backgroundColor: '#4096ff',
  };

const layoutStyle = {
  borderRadius: 8,
  overflow: 'hidden',
  width: 'calc(50% - 8px)',
  maxWidth: 'calc(50% - 8px)',
};

const App: React.FC = () => (
  <Flex gap="middle" wrap>
    <Layout style={layoutStyle}>
      <Header style={headerStyle}>Header</Header>
      <Content style={contentStyle}>Content</Content>
      <Footer style={footerStyle}>Footer</Footer>
    </Layout>

    <Layout style={layoutStyle}>
      <Header style={headerStyle}>Header</Header>
      <Layout>
        <Sider width="25%" style={siderStyle}>
          Sider
        </Sider>
        <Content style={contentStyle}>Content</Content>
      </Layout>
      <Footer style={footerStyle}>Footer</Footer>
    </Layout>

    <Layout style={layoutStyle}>
      <Header style={headerStyle}>Header</Header>
      <Layout>
        <Content style={contentStyle}>Content</Content>
        <Sider width="25%" style={siderStyle}>
          Sider
        </Sider>
      </Layout>
      <Footer style={footerStyle}>Footer</Footer>
    </Layout>

    <Layout style={layoutStyle}>
      <Sider width="25%" style={siderStyle}>
        Sider
      </Sider>
    </Layout>
  </Flex>
);

```

```

      <Header style={headerStyle}>Header</Header>
      <Content style={contentStyle}>Content</Content>
      <Footer style={footerStyle}>Footer</Footer>
    </Layout>
  </Layout>
</Flex>
);

export default App;

```

## Header-Content-Footer

```

import React from 'react';
import { Breadcrumb, Layout, Menu, theme } from 'antd';

const { Header, Content, Footer } = Layout;

const items = Array.from({ length: 15 }).map((_, index) => ({
  key: index + 1,
  label: `nav ${index + 1}`,
}));

const App: React.FC = () => {
  const {
    token: { colorBgContainer, borderRadiusLG },
  } = theme.useToken();

  return (
    <Layout>
      <Header style={{ display: 'flex', alignItems: 'center' }}>
        <div className="demo-logo" />
        <Menu
          theme="dark"
          mode="horizontal"
          defaultSelectedKeys={['2']}
          items={items}
          style={{ flex: 1, minWidth: 0 }}
        />
      </Header>
      <Content style={{ padding: '0 48px' }}>
        <Breadcrumb style={{ margin: '16px 0' }}>
          <Breadcrumb.Item>Home</Breadcrumb.Item>
          <Breadcrumb.Item>List</Breadcrumb.Item>
          <Breadcrumb.Item>App</Breadcrumb.Item>
        </Breadcrumb>
      </Content>
    </Layout>
  );
};

```

```

        style={{
          background: colorBgContainer,
          minHeight: 280,
          padding: 24,
          borderRadius: borderRadiusLG,
        }}
      >
        Content
      </div>
    </Content>
    <Footer style={{ textAlign: 'center' }}>
      Ant Design ©{new Date().getFullYear()} Created by Ant UED
    </Footer>
  </Layout>
);
};

export default App;

```

## Header-Sider

```

import React from 'react';
import { LaptopOutlined, NotificationOutlined, UserOutlined } from '@ant-
design/icons';
import type { MenuProps } from 'antd';
import { Breadcrumb, Layout, Menu, theme } from 'antd';

const { Header, Content, Footer, Sider } = Layout;

const items1: MenuProps['items'] = ['1', '2', '3'].map((key) => ({
  key,
  label: `nav ${key}`,
}));

const items2: MenuProps['items'] = [UserOutlined, LaptopOutlined,
NotificationOutlined].map(
  (icon, index) => {
    const key = String(index + 1);

    return {
      key: `sub${key}`,
      icon: React.createElement(icon),
      label: `subnav ${key}`,
      children: Array.from({ length: 4 }).map((_, j) => {
        const subKey = index * 4 + j + 1;
        return {

```

```

        key: subKey,
        label: `option${subKey}`,
      };
    }),
  };
},
);

const App: React.FC = () => {
  const {
    token: { colorBgContainer, borderRadiusLG },
  } = theme.useToken();

  return (
    <Layout>
      <Header style={{ display: 'flex', alignItems: 'center' }}>
        <div className="demo-logo" />
        <Menu
          theme="dark"
          mode="horizontal"
          defaultSelectedKeys={['2']}
          items={items1}
          style={{ flex: 1, minWidth: 0 }}
        />
      </Header>
      <div style={{ padding: '0 48px' }}>
        <Breadcrumb style={{ margin: '16px 0' }}>
          <Breadcrumb.Item>Home</Breadcrumb.Item>
          <Breadcrumb.Item>List</Breadcrumb.Item>
          <Breadcrumb.Item>App</Breadcrumb.Item>
        </Breadcrumb>
        <Layout
          style={{ padding: '24px 0', background: colorBgContainer,
borderRadius: borderRadiusLG }}
        >
          <Sider style={{ background: colorBgContainer }} width={200}>
            <Menu
              mode="inline"
              defaultSelectedKeys={['1']}
              defaultOpenKeys={['sub1']}
              style={{ height: '100%' }}
              items={items2}
            />
          </Sider>
          <Content style={{ padding: '0 24px', minHeight: 280
}}>Content</Content>

```

```

        </Layout>
      </div>
      <Footer style={{ textAlign: 'center' }}>
        Ant Design ©{new Date().getFullYear()} Created by Ant UED
      </Footer>
    </Layout>
  );
};

export default App;

```

## Header Sider 2

```

import React from 'react';
import { LaptopOutlined, NotificationOutlined, UserOutlined } from '@ant-
design/icons';
import type { MenuProps } from 'antd';
import { Breadcrumb, Layout, Menu, theme } from 'antd';

const { Header, Content, Sider } = Layout;

const items1: MenuProps['items'] = ['1', '2', '3'].map((key) => ({
  key,
  label: `nav ${key}`,
})));

const items2: MenuProps['items'] = [UserOutlined, LaptopOutlined,
NotificationOutlined].map(
  (icon, index) => {
    const key = String(index + 1);

    return {
      key: `sub${key}`,
      icon: React.createElement(icon),
      label: `subnav ${key}`,
      children: Array.from({ length: 4 }).map((_, j) => {
        const subKey = index * 4 + j + 1;
        return {
          key: subKey,
          label: `option${subKey}`,
        };
      }),
    };
  },
);


```

```

const App: React.FC = () => {
  const {
    token: { colorBgContainer, borderRadiusLG },
  } = theme.useToken();

  return (
    <Layout>
      <Header style={{ display: 'flex', alignItems: 'center' }}>
        <div className="demo-logo" />
        <Menu
          theme="dark"
          mode="horizontal"
          defaultSelectedKeys={['2']}
          items={items1}
          style={{ flex: 1, minWidth: 0 }}
        />
      </Header>
      <Layout>
        <Sider width={200} style={{ background: colorBgContainer }}>
          <Menu
            mode="inline"
            defaultSelectedKeys={['1']}
            defaultOpenKeys={['sub1']}
            style={{ height: '100%', borderRight: 0 }}
            items={items2}
          />
        </Sider>
        <Layout style={{ padding: '0 24px 24px' }}>
          <Breadcrumb
            items={[{ title: 'Home' }, { title: 'List' }, { title: 'App'
            style={{ margin: '16px 0' }}
          />
          <Content
            style={{
              padding: 24,
              margin: 0,
              minHeight: 280,
              background: colorBgContainer,
              borderRadius: borderRadiusLG,
            }}
          >
            Content
          </Content>
        </Layout>
      </Layout>
    )
  )
}

```



```

    </Layout>
  );
};

export default App;

```

## Sider

```

import React, { useState } from 'react';
import {
  DesktopOutlined,
  FileOutlined,
  PieChartOutlined,
  TeamOutlined,
  UserOutlined,
} from '@ant-design/icons';
import type { MenuProps } from 'antd';
import { Breadcrumb, Layout, Menu, theme } from 'antd';

const { Header, Content, Footer, Sider } = Layout;

type MenuItem = Required<MenuProps>['items'][number];

function getItem(
  label: React.ReactNode,
  key: React.Key,
  icon?: React.ReactNode,
  children?: MenuItem[],
): MenuItem {
  return {
    key,
    icon,
    children,
    label,
  } as MenuItem;
}

const items: MenuItem[] = [
  getItem('Option 1', '1', <PieChartOutlined />),
  getItem('Option 2', '2', <DesktopOutlined />),
  getItem('User', 'sub1', <UserOutlined />, [
    getItem('Tom', '3'),
    getItem('Bill', '4'),
    getItem('Alex', '5'),
  ]),
  getItem('Team', 'sub2', <TeamOutlined />, [getItem('Team 1', '6'),

```

```

getItem('Team 2', '8')]],
  getItem('Files', '9', <FileOutlined />),
];

const App: React.FC = () => {
  const [collapsed, setCollapsed] = useState(false);
  const {
    token: { colorBgContainer, borderRadiusLG },
  } = theme.useToken();

  return (
    <Layout style={{ minHeight: '100vh' }}>
      <Sider collapsible collapsed={collapsed} onCollapse={(value) =>
setCollapsed(value)}>
        <div className="demo-logo-vertical" />
        <Menu theme="dark" defaultSelectedKeys={['1']} mode="inline" items=
{items} />
      </Sider>
      <Layout>
        <Header style={{ padding: 0, background: colorBgContainer }} />
        <Content style={{ margin: '0 16px' }}>
          <Breadcrumb style={{ margin: '16px 0' }}>
            <Breadcrumb.Item>User</Breadcrumb.Item>
            <Breadcrumb.Item>Bill</Breadcrumb.Item>
          </Breadcrumb>
          <div
            style={{
              padding: 24,
              minHeight: 360,
              background: colorBgContainer,
              borderRadius: borderRadiusLG,
            }}
          >
            Bill is a cat.
          </div>
        </Content>
        <Footer style={{ textAlign: 'center' }}>
          Ant Design ©{new Date().getFullYear()} Created by Ant UED
        </Footer>
      </Layout>
    </Layout>
  );
};

export default App;

```

## Custom trigger

```
import React, { useState } from 'react';
import {
  MenuFoldOutlined,
  MenuUnfoldOutlined,
  UploadOutlined,
  UserOutlined,
  VideoCameraOutlined,
} from '@ant-design/icons';
import { Button, Layout, Menu, theme } from 'antd';

const { Header, Sider, Content } = Layout;

const App: React.FC = () => {
  const [collapsed, setCollapsed] = useState(false);
  const {
    token: { colorBgContainer, borderRadiusLG },
  } = theme.useToken();

  return (
    <Layout>
      <Sider trigger={null} collapsible collapsed={collapsed}>
        <div className="demo-logo-vertical" />
        <Menu
          theme="dark"
          mode="inline"
          defaultSelectedKeys={['1']}
          items={[
            {
              key: '1',
              icon: <UserOutlined />,
              label: 'nav 1',
            },
            {
              key: '2',
              icon: <VideoCameraOutlined />,
              label: 'nav 2',
            },
            {
              key: '3',
              icon: <UploadOutlined />,
              label: 'nav 3',
            },
          ]}
        />
      </Sider>
    </Layout>
  );
};
```

```

</Sider>
<Layout>
  <Header style={{ padding: 0, background: colorBgContainer }}>
    <Button
      type="text"
      icon={collapsed ? <MenuUnfoldOutlined /> : <MenuFoldOutlined
/>}
      onClick={() => setCollapsed(!collapsed)}
      style={{
        fontSize: '16px',
        width: 64,
        height: 64,
      }}
    />
  </Header>
  <Content
    style={{
      margin: '24px 16px',
      padding: 24,
      minHeight: 280,
      background: colorBgContainer,
      borderRadius: borderRadiusLG,
    }}
  >
    Content
  </Content>
</Layout>
</Layout>
);
};

export default App;

```

## Responsive

```

import React from 'react';
import { UploadOutlined, UserOutlined, VideoCameraOutlined } from '@ant-
design/icons';
import { Layout, Menu, theme } from 'antd';

const { Header, Content, Footer, Sider } = Layout;

const items = [UserOutlined, VideoCameraOutlined, UploadOutlined,
UserOutlined].map(
  (icon, index) => ({
    key: String(index + 1),

```

```

        icon: React.createElement(icon),
        label: `nav ${index + 1}`,
    })),
);

const App: React.FC = () => {
    const {
        token: { colorBgContainer, borderRadiusLG },
    } = theme.useToken();

    return (
        <Layout>
            <Sider
                breakpoint="lg"
                collapsedWidth="0"
                onBreakpoint={(broken) => {
                    console.log(broken);
                }}
                onCollapse={(collapsed, type) => {
                    console.log(collapsed, type);
                }}
            >
                <div className="demo-logo-vertical" />
                <Menu theme="dark" mode="inline" defaultSelectedKeys={['4']} items=
{items} />
            </Sider>
            <Layout>
                <Header style={{ padding: 0, background: colorBgContainer }} />
                <Content style={{ margin: '24px 16px 0' }}>
                    <div
                        style={{
                            padding: 24,
                            minHeight: 360,
                            background: colorBgContainer,
                            borderRadius: borderRadiusLG,
                        }}
                    >
                        content
                    </div>
                </Content>
                <Footer style={{ textAlign: 'center' }}>
                    Ant Design ©{new Date().getFullYear()} Created by Ant UED
                </Footer>
            </Layout>
        </Layout>
    );
};

```

```
};
```

```
export default App;
```

## Fixed Header

```
import React from 'react';
import { Breadcrumb, Layout, Menu, theme } from 'antd';

const { Header, Content, Footer } = Layout;

const items = Array.from({ length: 3 }).map((_, index) => ({
  key: String(index + 1),
  label: `nav ${index + 1}`,
}));

const App: React.FC = () => {
  const {
    token: { colorBgContainer, borderRadiusLG },
  } = theme.useToken();

  return (
    <Layout>
      <Header
        style={{
          position: 'sticky',
          top: 0,
          zIndex: 1,
          width: '100%',
          display: 'flex',
          alignItems: 'center',
        }}
      >
        <div className="demo-logo" />
        <Menu
          theme="dark"
          mode="horizontal"
          defaultSelectedKeys={['2']}
          items={items}
          style={{ flex: 1, minWidth: 0 }}
        />
      </Header>
      <Content style={{ padding: '0 48px' }}>
        <Breadcrumb style={{ margin: '16px 0' }}>
          <Breadcrumb.Item>Home</Breadcrumb.Item>
          <Breadcrumb.Item>List</Breadcrumb.Item>
        </Breadcrumb>
      </Content>
    </Layout>
  );
};
```

```

        <Breadcrumb.Item>App</Breadcrumb.Item>
      </Breadcrumb>
    <div
      style={{
        padding: 24,
        minHeight: 380,
        background: colorBgContainer,
        borderRadius: borderRadiusLG,
      }}
    >
      Content
    </div>
  </Content>
  <Footer style={{ textAlign: 'center' }}>
    Ant Design ©{new Date().getFullYear()} Created by Ant UED
  </Footer>
</Layout>
);
};

export default App;

```

## Fixed Sider

```

import React from 'react';
import {
  AppstoreOutlined,
  BarChartOutlined,
  CloudOutlined,
  ShopOutlined,
  TeamOutlined,
  UploadOutlined,
  UserOutlined,
  VideoCameraOutlined,
} from '@ant-design/icons';
import type { MenuProps } from 'antd';
import { Layout, Menu, theme } from 'antd';

const { Header, Content, Footer, Sider } = Layout;

const siderStyle: React.CSSProperties = {
  overflow: 'auto',
  height: '100vh',
  position: 'sticky',
  insetInlineStart: 0,
  top: 0,

```

```

    bottom: 0,
    scrollbarWidth: 'thin',
    scrollbarGutter: 'stable',
  };

const items: MenuProps['items'] = [
  UserOutlined,
  VideoCameraOutlined,
  UploadOutlined,
  BarChartOutlined,
  CloudOutlined,
  AppstoreOutlined,
  TeamOutlined,
  ShopOutlined,
].map((icon, index) => ({
  key: String(index + 1),
  icon: React.createElement(icon),
  label: `nav ${index + 1}`,
}));

const App: React.FC = () => {
  const {
    token: { colorBgContainer, borderRadiusLG },
  } = theme.useToken();

  return (
    <Layout hasSider>
      <Sider style={siderStyle}>
        <div className="demo-logo-vertical" />
        <Menu theme="dark" mode="inline" defaultSelectedKeys={['4']} items=
{items} />
      </Sider>
      <Layout>
        <Header style={{ padding: 0, background: colorBgContainer }} />
        <Content style={{ margin: '24px 16px 0', overflow: 'initial' }}>
          <div
            style={{
              padding: 24,
              textAlign: 'center',
              background: colorBgContainer,
              borderRadius: borderRadiusLG,
            }}
          >
            <p>long content</p>
            {
              // indicates very long content

```



```

        Array.from({ length: 100 }, (_, index) => (
          <React.Fragment key={index}>
            {index % 20 === 0 && index ? 'more' : '...'}
            <br />
          </React.Fragment>
        ))
      }
    </div>
  </Content>
  <Footer style={{ textAlign: 'center' }}>
    Ant Design ©{new Date().getFullYear()} Created by Ant UED
  </Footer>
</Layout>
</Layout>
);
};

export default App;

```

## Custom trigger debug

Debug

```

import React, { useState } from 'react';
import {
  DesktopOutlined,
  FileOutlined,
  MenuFoldOutlined,
  MenuUnfoldOutlined,
  PieChartOutlined,
  TeamOutlined,
  UserOutlined,
} from '@ant-design/icons';
import type { MenuProps } from 'antd';
import { Button, Layout, Menu, theme } from 'antd';

const { Header, Sider, Content } = Layout;

const items: MenuProps['items'] = [
  {
    key: '1',
    icon: <PieChartOutlined />,
    label: 'Option 1',
  },
  {
    key: '2',

```

```

    icon: <DesktopOutlined />,
    label: 'Option 2',
  },
  {
    key: 'sub1',
    icon: <UserOutlined />,
    label: 'User',
    children: [
      {
        key: '3',
        label: 'Tom',
      },
      {
        key: '4',
        label: 'Bill',
      },
      {
        key: '5',
        label: 'Alex',
      },
    ],
  },
],
},
{
  key: 'sub2',
  icon: <TeamOutlined />,
  label: 'Team',
  children: [
    {
      key: '6',
      label: 'Team 1',
    },
    {
      key: '7',
      label: 'Team 2',
    },
  ],
},
{
  key: '9',
  icon: <FileOutlined />,
},
];

```

```

const App: React.FC = () => {
  const [collapsed, setCollapsed] = useState(true);
  const {

```

```

    token: { colorBgContainer, borderRadiusLG },
  } = theme.useToken();

  return (
    <Layout>
      <Sider trigger={null} collapsible collapsed={collapsed}>
        <div className="demo-logo-vertical" />
        <Menu
          theme="dark"
          mode="inline"
          defaultSelectedKeys={['3']}
          defaultOpenKeys={['sub1']}
          items={items}
        />
      </Sider>
      <Layout>
        <Header style={{ padding: 0, background: colorBgContainer }}>
          <Button
            type="text"
            icon={collapsed ? <MenuUnfoldOutlined /> : <MenuFoldOutlined
/>}
            onClick={() => setCollapsed(!collapsed)}
            style={{
              fontSize: '16px',
              width: 64,
              height: 64,
            }}
          />
        </Header>
        <Content
          style={{
            margin: '24px 16px',
            padding: 24,
            minHeight: 280,
            background: colorBgContainer,
            borderRadius: borderRadiusLG,
          }}
        >
          Content
        </Content>
      </Layout>
    </Layout>
  );
};

export default App;

```

## Component Token

Debug

```
import React from 'react';
import { LaptopOutlined, NotificationOutlined, UserOutlined } from '@ant-
design/icons';
import type { MenuProps } from 'antd';
import { Breadcrumb, ConfigProvider, Layout, Menu, theme } from 'antd';

const { Header, Content, Sider } = Layout;

const items2: MenuProps['items'] = [UserOutlined, LaptopOutlined,
NotificationOutlined].map(
  (icon, index) => {
    const key = String(index + 1);

    return {
      key: `sub${key}`,
      icon: React.createElement(icon),
      label: `subnav ${key}`,
      children: Array.from({ length: 4 }).map((_, j) => {
        const subKey = index * 4 + j + 1;
        return {
          key: subKey,
          label: `option${subKey}`,
        };
      }),
    };
  },
);

const App: React.FC = () => {
  const {
    token: { colorBgContainer, colorBgLayout, borderRadiusLG },
  } = theme.useToken();

  return (
    <ConfigProvider
      theme={{
        components: {
          Layout: {
            bodyBg: '#fff',
            headerBg: '#1677ff',
            headerHeight: 64,
            headerPadding: `0 24px`,
```

```

        headerColor: colorBgContainer,
        siderBg: '#800080',
      },
    },
  }}
>
<Layout>
  <Header style={{ display: 'flex', alignItems: 'center' }}>
    <div className="demo-logo" />
    <div style={{ marginInlineStart: 24, fontSize: 24 }}>Ant
Design</div>
  </Header>
</Layout>
<ConfigProvider
  theme={{
    components: {
      Layout: {
        siderBg: 'red',
      },
    },
  }}
>
  <Sider width={32} />
</ConfigProvider>
<Sider width={200}>
  <Menu
    mode="inline"
    defaultSelectedKeys={['1']}
    defaultOpenKeys={['sub1']}
    style={{ borderRight: 0 }}
    items={items2}
  />
</Sider>
<Layout style={{ padding: '0 24px 24px' }}>
  <Breadcrumb style={{ margin: '16px 0' }}>
    <Breadcrumb.Item>Home</Breadcrumb.Item>
    <Breadcrumb.Item>List</Breadcrumb.Item>
    <Breadcrumb.Item>App</Breadcrumb.Item>
  </Breadcrumb>
  <Content
    style={{
      padding: 24,
      margin: 0,
      minHeight: 280,
      background: colorBgLayout,
      borderRadius: borderRadiusLG,

```

```

        }}
      >
        Content
      </Content>
    </Layout>
  </Layout>
</Layout>
</ConfigProvider>
);
};

export default App;

```

## API

```

<Layout>
  <Header>header</Header>
  <Layout>
    <Sider>left sidebar</Sider>
    <Content>main content</Content>
    <Sider>right sidebar</Sider>
  </Layout>
  <Footer>footer</Footer>
</Layout>

```

### Layout

Common props ref: [Common props](#)

The wrapper.

Property	Description	Type	Default
className	Container className	string	-
hasSider	Whether contain Sider in children, don't have to assign it normally. Useful in SSR avoid style flickering	boolean	-
style	To customize the styles	CSSProperties	-

### Layout.Sider

The sidebar.

Property	Description	Type	Default
----------	-------------	------	---------

breakpoint	<a href="#">Breakpoints</a> of the responsive layout	xs   sm   md   lg   xl   xxl	-
className	Container className	string	-
collapsed	To set the current status	boolean	-
collapsedWidth	Width of the collapsed sidebar, by setting to 0 a special trigger will appear	number	80
collapsible	Whether can be collapsed	boolean	false
defaultCollapsed	To set the initial status	boolean	false
reverseArrow	Reverse direction of arrow, for a sider that expands from the right	boolean	false
style	To customize the styles	CSSProperties	-
theme	Color theme of the sidebar	light   dark	dark
trigger	Specify the customized trigger, set to null to hide the trigger	ReactNode	-
width	Width of the sidebar	number   string	200
zeroWidthTriggerStyle	To customize the styles of the special trigger that appears when collapsedWidth is 0	object	-
onBreakpoint	The callback function, executed when <a href="#">breakpoints</a> changed	(broken) => {}	-
onCollapse	The callback function, executed by clicking the trigger or activating the responsive layout	(collapsed, type) => {}	-

### breakpoint width

```
{
  xs: '480px',
  sm: '576px',
  md: '768px',
  lg: '992px',
  xl: '1200px',
  xxl: '1600px',
}
```

**Design Token**