

何时使用 {#when-to-use}

最基础的卡片容器，可承载文字、列表、图片、段落，常用于后台概览页面。

代码演示

典型卡片

```
import React from 'react';
import { Card, Space } from 'antd';

const App: React.FC = () => (
  <Space direction="vertical" size={16}>
    <Card title="Default size card" extra={<a href="#">More</a>} style={{
width: 300 }}>
      <p>Card content</p>
      <p>Card content</p>
      <p>Card content</p>
    </Card>
    <Card size="small" title="Small size card" extra={<a href="#">More</a>}
style={{ width: 300 }}>
      <p>Card content</p>
      <p>Card content</p>
      <p>Card content</p>
    </Card>
  </Space>
);

export default App;
```

无边框

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

const App: React.FC = () => (
  <Card title="Card title" variant="borderless" style={{ width: 300 }}>
    <p>Card content</p>
    <p>Card content</p>
    <p>Card content</p>
  </Card>
);

export default App;
```

简洁卡片

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

const App: React.FC = () => (
  <Card style={{ width: 300 }}>
    <p>Card content</p>
    <p>Card content</p>
    <p>Card content</p>
  </Card>
);

export default App;
```

更灵活的内容展示

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

const { Meta } = Card;

const App: React.FC = () => (
  <Card
    hoverable
    style={{ width: 240 }}
    cover=
    >
    <Meta title="Europe Street beat" description="www.instagram.com" />
  </Card>
);

export default App;
```

栅格卡片

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

const App: React.FC = () => (
  <Row gutter={16}>
    <Col span={8}>
      <Card title="Card title" variant="borderless">
```

```

        Card content
      </Card>
    </Col>
    <Col span={8}>
      <Card title="Card title" variant="borderless">
        Card content
      </Card>
    </Col>
    <Col span={8}>
      <Card title="Card title" variant="borderless">
        Card content
      </Card>
    </Col>
  </Row>
);

export default App;

```

预加载的卡片

```

import React, { useState } from 'react';
import { EditOutlined, EllipsisOutlined, SettingOutlined } from '@ant-
design/icons';
import { Avatar, Card, Flex, Switch } from 'antd';

const actions: React.ReactNode[] = [
  <EditOutlined key="edit" />,
  <SettingOutlined key="setting" />,
  <EllipsisOutlined key="ellipsis" />,
];

const App: React.FC = () => {
  const [loading, setLoading] = useState<boolean>(true);
  return (
    <Flex gap="middle" align="start" vertical>
      <Switch checked={!loading} onChange={(checked) =>
setLoading(!checked)} />
      <Card loading={loading} actions={actions} style={{ minWidth: 300 }}>
        <Card.Meta
          avatar={<Avatar src="https://api.dicebear.com/7.x/miniavs/svg?
seed=1" />}
          title="Card title"
          description={
            <>
              <p>This is the description</p>
              <p>This is the description</p>
            </>

```

```

        </>
    }
    />
</Card>
<Card loading={loading} actions={actions} style={{ minWidth: 300 }}>
    <Card.Meta
        avatar={<Avatar src="https://api.dicebear.com/7.x/miniavs/svg?
seed=2" />}
        title="Card title"
        description={
            <>
                <p>This is the description</p>
                <p>This is the description</p>
            </>
        }
    />
</Card>
</Flex>
);
};

export default App;

```

网格型内嵌卡片

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

const gridStyle: React.CSSProperties = {
  width: '25%',
  textAlign: 'center',
};

const App: React.FC = () => (
  <Card title="Card Title">
    <Card.Grid style={gridStyle}>Content</Card.Grid>
    <Card.Grid hoverable={false} style={gridStyle}>
      Content
    </Card.Grid>
    <Card.Grid style={gridStyle}>Content</Card.Grid>
    <Card.Grid style={gridStyle}>Content</Card.Grid>
    <Card.Grid style={gridStyle}>Content</Card.Grid>
    <Card.Grid style={gridStyle}>Content</Card.Grid>
  </Card>
);
```

```
export default App;
```

内部卡片

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

const App: React.FC = () => (
  <Card title="Card title">
    <Card type="inner" title="Inner Card title" extra={<a
href="#">More</a>}>
      Inner Card content
    </Card>
    <Card
      style={{ marginTop: 16 }}
      type="inner"
      title="Inner Card title"
      extra={<a href="#">More</a>}
    >
      Inner Card content
    </Card>
  </Card>
);

export default App;
```

带页签的卡片

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

const tabList = [
  {
    key: 'tab1',
    tab: 'tab1',
  },
  {
    key: 'tab2',
    tab: 'tab2',
  },
];

const contentList: Record<string, React.ReactNode> = {
  tab1: <p>content1</p>,

```

```

    tab2: <p>content2</p>,
};

const tabListNoTitle = [
  {
    key: 'article',
    label: 'article',
  },
  {
    key: 'app',
    label: 'app',
  },
  {
    key: 'project',
    label: 'project',
  },
];

const contentListNoTitle: Record<string, React.ReactNode> = {
  article: <p>article content</p>,
  app: <p>app content</p>,
  project: <p>project content</p>,
};

const App: React.FC = () => {
  const [activeTabKey1, setActiveTabKey1] = useState<string>('tab1');
  const [activeTabKey2, setActiveTabKey2] = useState<string>('app');

  const onTab1Change = (key: string) => {
    setActiveTabKey1(key);
  };
  const onTab2Change = (key: string) => {
    setActiveTabKey2(key);
  };

  return (
    <>
      <Card
        style={{ width: '100%' }}
        title="Card title"
        extra={<a href="#">More</a>}
        tabList={tabList}
        activeTabKey={activeTabKey1}
        onTabChange={onTab1Change}
      >
        {contentList[activeTabKey1]}
      </Card>
    </>
  );
};

```

```

    </Card>
    <br />
    <br />
    <Card
      style={{ width: '100%' }}
      tabList={tabListNoTitle}
      activeTabKey={activeTabKey2}
      tabBarExtraContent={<a href="#">More</a>}
      onTabChange={onTab2Change}
      tabProps={{
        size: 'middle',
      }}
    >
      {contentListNoTitle[activeTabKey2]}
    </Card>
  </>
);
};

export default App;

```

支持更多内容配置

```

import React from 'react';
import { EditOutlined, EllipsisOutlined, SettingOutlined } from '@ant-
design/icons';
import { Avatar, Card } from 'antd';

const { Meta } = Card;

const App: React.FC = () => (
  <Card
    style={{ width: 300 }}
    cover={
      
    >
    actions=[
      <SettingOutlined key="setting" />,
      <EditOutlined key="edit" />,
      <EllipsisOutlined key="ellipsis" />,
    ]
  >

```

```

    <Meta
      avatar={<Avatar src="https://api.dicebear.com/7.x/miniavs/svg?seed=8"
/>}
      title="Card title"
      description="This is the description"
    />
  </Card>
);

export default App;

```

组件 Token

Debug

```

import React from 'react';
import { EditOutlined, EllipsisOutlined, SettingOutlined } from '@ant-
design/icons';
import { Card, ConfigProvider } from 'antd';

export default () => (
  <ConfigProvider
    theme={{
      components: {
        Card: {
          headerBg: '#e6f4ff',
          bodyPaddingSM: 22,
          headerPaddingSM: 20,
          headerPadding: 18,
          bodyPadding: 26,
          headerFontSize: 20,
          headerFontSizeSM: 20,
          headerHeight: 60,
          headerHeightSM: 60,
          actionsBg: '#e6f4ff',
          actionsLiMargin: `2px 0`,
          tabsMarginBottom: 0,
          extraColor: 'rgba(0,0,0,0.25)',
        },
      },
    }}
  >
    <Card
      title="Card title"
      actions={[
        <SettingOutlined key="setting" />,

```



```
        <EditOutlined key="edit" />,
        <EllipsisOutlined key="ellipsis" />,
    ]}
    extra="More"
    tabList={[
      {
        key: 'tab1',
        label: 'tab1',
      },
      {
        key: 'tab2',
        label: 'tab2',
      },
    ]}
  >
    <p>Card content</p>
    <p>Card content</p>
    <p>Card content</p>
  </Card>
  <Card size="small" title="Small size card" extra={<a href="#">More</a>}
style={{ width: 300 }}>
    <p>Card content</p>
    <p>Card content</p>
    <p>Card content</p>
  </Card>
</ConfigProvider>
);
```

API

通用属性参考: [通用属性](#)

```
<Card title="卡片标题">卡片内容</Card>
```

参数	说明	类型	默认值	版本
actions	卡片操作组, 位置在卡片底部	Array<ReactNode>	-	
activeTabKey	当前激活页签的key	string	-	
bordered	是否有边框, 请使用 variant 替换	boolean	true	

variant	形态变体	outlined borderless	outlined	5.24.0
cover	卡片封面	ReactNode	-	
defaultActiveTabKey	初始化选中页签的 key，如果没有设置 activeTabKey	string	第一个页签的 key	
extra	卡片右上角的操作区域	ReactNode	-	
hoverable	鼠标移过时可浮起	boolean	false	
loading	当卡片内容还在加载中时，可以用 loading 展示一个占位	boolean	false	
size	card 的尺寸	default small	default	
tabBarExtraContent	tab bar 上额外的元素	ReactNode	-	
tabList	页签标题列表	TabItem.Type []	-	
tabProps	Tabs	-	-	
title	卡片标题	ReactNode	-	
type	卡片类型，可设置为 inner 或不设置	string	-	
classNames	配置卡片内置模块的 className	Record<SemanticDOM, string>	-	5.14.0
styles	配置卡片内置模块的 style	Record<SemanticDOM, CSSProperties>	-	5.14.0
onTabChange	页签切换的回调	(key) => void	-	

Card.Grid

参数	说明	类型	默认值	版本
className	网格容器类名	string	-	
hoverable	鼠标移过时可浮起	boolean	true	
style	定义网格容器类名的样式	CSSProperties	-	

Card.Meta

参数	说明	类型	默认值	版本
avatar	头像/图标	ReactNode	-	
className	容器类名	string	-	
description	描述内容	ReactNode	-	
style	定义容器类名的样式	CSSProperties	-	
title	标题内容	ReactNode	-	

Semantic DOM

演示

```
import React from 'react';

import SemanticPreview from '../../../dumi/components/SemanticPreview';
import useLocale from '../../../dumi/hooks/useLocale';

import { EditOutlined, EllipsisOutlined, SettingOutlined } from '@ant-
design/icons';
import { Avatar, Card } from 'antd';

const { Meta } = Card;

const locales = {
  cn: {
    header: '设置卡片头部区域',
    body: '设置卡片内容区域',
    extra: '设置卡片右上角的操作区域',
    title: '设置卡片标题',
    actions: '设置卡片底部操作组',
    cover: '设置标题封面',
  },
  en: {
    header: 'set `header` of card',
    body: 'set `body` of card',
    extra: 'set `extra` of card',
    title: 'set `title` of card',
    actions: 'set `actions` of card',
    cover: 'set `cover` of card',
  },
};
```

```

const BlockCard: React.FC<React.PropsWithChildren> = (props) => {
  const divRef = React.useRef<HTMLDivElement>(null);

  return (
    <div ref={divRef} style={{ position: 'absolute', inset: 0 }}>
      <Card
        {...props}
        title="Card title"
        extra="More"
        style={{ width: 300 }}
        cover={
          
          </div>
          actions=[
            <SettingOutlined key="setting" />,
            <EditOutlined key="edit" />,
            <EllipsisOutlined key="ellipsis" />,
          ]
        />
      </div>
    );
};

const App: React.FC = () => {
  const [locale] = useLocale(locales);
  return (
    <SemanticPreview
      componentName="Card"
      semantics=[
        { name: 'header', desc: locale.header, version: '5.14.0' },
        { name: 'title', desc: locale.title, version: '5.14.0' },
        { name: 'extra', desc: locale.extra, version: '5.14.0' },
        { name: 'cover', desc: locale.cover, version: '5.14.0' },
        { name: 'body', desc: locale.body, version: '5.14.0' },
        { name: 'actions', desc: locale.actions, version: '5.14.0' },
      ]
    >
      <BlockCard>
        <Meta
          avatar={<Avatar src="https://api.dicebear.com/7.x/miniavs/svg?seed=8" />}
          title="Card Meta title"

```

```
        description="This is the description"
      />
    </BlockCard>
  </SemanticPreview>
);
};

export default App;
```

主题变量 (Design Token)