

When To Use

- When want to highlight some data.
- When want to display statistic data with description.

Examples

Basic

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

const App: React.FC = () => (
  <Row gutter={16}>
    <Col span={12}>
      <Statistic title="Active Users" value={112893} />
    </Col>
    <Col span={12}>
      <Statistic title="Account Balance (CNY)" value={112893} precision={2} />
    </Col>
  </Row>
  <Button style={{ marginTop: 16 }} type="primary">
    Recharge
  </Button>
  <Col span={12}>
    <Statistic title="Active Users" value={112893} loading />
  </Col>
</Row>
);

export default App;
```

Unit

```
import React from 'react';
import { LikeOutlined } from '@ant-design/icons';
import { Col, Row, Statistic } from 'antd';

const App: React.FC = () => (
  <Row gutter={16}>
    <Col span={12}>
      <Statistic title="Feedback" value={1128} prefix={<LikeOutlined />} />
    </Col>
    <Col span={12}>
      <Statistic title="Unmerged" value={93} suffix="/ 100" />
    </Col>
  </Row>
);
```

```

    </Row>
  );

  export default App;

```

Animated number

```

import React from 'react';
import type { StatisticProps } from 'antd';
import { Col, Row, Statistic } from 'antd';
import CountUp from 'react-countup';

const formatter: StatisticProps['formatter'] = (value) => (
  <CountUp end={value as number} separator="," />
);

const App: React.FC = () => (
  <Row gutter={16}>
    <Col span={12}>
      <Statistic title="Active Users" value={112893} formatter={formatter} />
    </Col>
    <Col span={12}>
      <Statistic title="Account Balance (CNY)" value={112893} precision={2}
formatter={formatter} />
    </Col>
  </Row>
);

export default App;

```

In Card

```

import React from 'react';
import { ArrowDownOutlined, ArrowUpOutlined } from '@ant-design/icons';
import { Card, Col, Row, Statistic } from 'antd';

const App: React.FC = () => (
  <Row gutter={16}>
    <Col span={12}>
      <Card variant="borderless">
        <Statistic
          title="Active"
          value={11.28}
          precision={2}

```

```

        valueStyle={{ color: '#3f8600' }}
        prefix={<ArrowUpOutlined />}
        suffix="%"
      />
    </Card>
  </Col>
  <Col span={12}>
    <Card variant="borderless">
      <Statistic
        title="Idle"
        value={9.3}
        precision={2}
        valueStyle={{ color: '#cf1322' }}
        prefix={<ArrowDownOutlined />}
        suffix="%"
      />
    </Card>
  </Col>
</Row>
);

export default App;

```

Countdown

```

import React from 'react';
import type { CountdownProps } from 'antd';
import { Col, Row, Statistic } from 'antd';

const { Countdown } = Statistic;

const deadline = Date.now() + 1000 * 60 * 60 * 24 * 2 + 1000 * 30; // Dayjs
// is also OK

const onFinish: CountdownProps['onFinish'] = () => {
  console.log('finished!');
};

const onChange: CountdownProps['onChange'] = (val) => {
  if (typeof val === 'number' && 4.95 * 1000 < val && val < 5 * 1000) {
    console.log('changed!');
  }
};

const App: React.FC = () => (
  <Row gutter={16}>

```

```

    <Col span={12}>
      <Countdown title="Countdown" value={deadline} onFinish={onFinish} />
    </Col>
    <Col span={12}>
      <Countdown title="Million Seconds" value={deadline}
format="HH:mm:ss:SSS" />
    </Col>
    <Col span={24} style={{ marginTop: 32 }}>
      <Countdown title="Day Level" value={deadline} format="D 天 H 时 m 分 s
秒" />
    </Col>
    <Col span={12}>
      <Countdown title="Countdown" value={Date.now() + 10 * 1000} onChange=
{onChange} />
    </Col>
  </Row>
);

export default App;

```

Component Token

Debug

```

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

const App: React.FC = () => (
  <ConfigProvider
    theme={{
      components: {
        Statistic: {
          titleFontSize: 20,
          contentFontSize: 20,
        },
      },
    }}
  >
    <Row gutter={16}>
      <Col span={12}>
        <Statistic title="Active Users" value={112893} />
      </Col>
      <Col span={12}>
        <Statistic title="Account Balance (CNY)" value={112893} precision=
{2} />
      <Button style={{ marginTop: 16 }} type="primary">

```

```

        Recharge
      </Button>
    </Col>
    <Col span={12}>
      <Statistic title="Active Users" value={112893} loading />
    </Col>
  </Row>
</ConfigProvider>
);

export default App;

```

API

Common props ref: [Common props](#)

Statistic

Property	Description	Type	Default	Version
decimalSeparator	The decimal separator	string	.	
formatter	Customize value display logic	(value) => ReactNode	-	
groupSeparator	Group separator	string	,	
loading	Loading status of Statistic	boolean	false	4.8.0
precision	The precision of input value	number	-	
prefix	The prefix node of value	ReactNode	-	
suffix	The suffix node of value	ReactNode	-	
title	Display title	ReactNode	-	
value	Display value	string number	-	
valueStyle	Set value section style	CSSProperties	-	

Statistic.Countdown

Property	Description	Type	Default	Version
format	Format as dayjs	string	HH:mm:ss	
prefix	The prefix node of value	ReactNode	-	

suffix	The suffix node of value	ReactNode	-	
title	Display title	ReactNode	-	
value	Set target countdown time	number	-	
valueStyle	Set value section style	CSSProperties	-	
onFinish	Trigger when time's up	() => void	-	
onChange	Trigger when time's changing	(value: number) => void	-	4.16.0

Design Token