Tooltip ✎

A simple text popup tip.

When To Use

- The tip is shown on mouse enter, and is hidden on mouse leave. The Tooltip doesn't support complex text or operations.
- To provide an explanation of a `button/text/operation`. It's often used instead of the html `title` attribute.

Examples

Tooltip will show on mouse enter.

Basic ✎
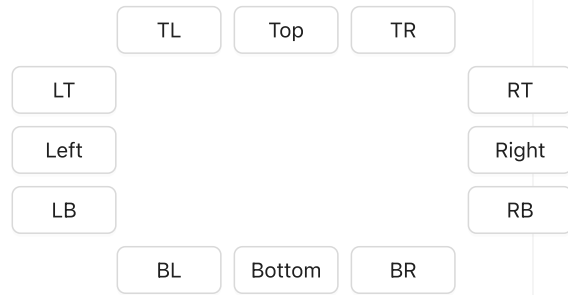
The simplest usage.

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

const App: React.FC = () => (
  <Tooltip title="prompt text">
    <span>Tooltip will show on mouse enter.</s
  </Tooltip>
);

export default App;
```

— — — — Content — — — —

TL	Top	TR
LT		RT
Left		Right
LB		RB
BL	Bottom	BR

TL	Top	TR
LT		RT
Left		Right
LB		RB
BL	Bottom	BR

Arrow ✎

Support show, hide or keep arrow in the center.

```
import { Button, Divider, Segmented, Tooltip }
import React, { useMemo, useState } from 'rea

const text = <span>prompt text</span>;

const buttonWidth = 70;
const gap = 8;

const btnProps = {
  style: {
    width: buttonWidth,
  },
};

const App: React.FC = () => {
  const options = ['Show', 'Hide', 'Center'];
  const [arrow, setArrow] = useState('Show');

  const mergedArrow = useMemo(() => {
    if (arrow === 'Hide') {
      return false;
    }

    if (arrow === 'Show') {
      return true;
    }

    return {
      pointAtCenter: true,
    };
  }, [arrow]);

  return (
    <div className="demo">
      <Segmented
        value={arrow}
        options={options}
        onChange={(val: string) => {
          setArrow(val);
        }}
      />
      <Divider orientation="center">Content</D
      <div style={{ marginLeft: buttonWidth, c
        <Tooltip placement="topLeft" title={te
          <Button {...btnProps}>TL</Button>
        </Tooltip>
        <Tooltip placement="top" title={text}
          <Button {...btnProps}>Top</Button>
        </Tooltip>
        <Tooltip placement="topRight" title={t
          <Button {...btnProps}>TR</Button>
        </Tooltip>
```

Placement ✎

There are 12 placement options available.

```
import { Button, Tooltip } from 'antd';
import React from 'react';

const text = <span>prompt text</span>;

const buttonWidth = 70;

const App: React.FC = () => (
  <div>
    <div style={{ marginLeft: buttonWidth, whi
      <Tooltip placement="topLeft" title={text
        <Button>TL</Button>
      </Tooltip>
      <Tooltip placement="top" title={text}>
        <Button>Top</Button>
      </Tooltip>
      <Tooltip placement="topRight" title={te
        <Button>TR</Button>
      </Tooltip>
    </div>
    <div style={{ width: buttonWidth, float: '
      <Tooltip placement="leftTop" title={text
        <Button>LT</Button>
      </Tooltip>
      <Tooltip placement="left" title={text}>
        <Button>Left</Button>
      </Tooltip>
      <Tooltip placement="leftBottom" title={t
        <Button>LB</Button>
      </Tooltip>
    </div>
    <div style={{ width: buttonWidth, marginLe
      <Tooltip placement="rightTop" title={te
        <Button>RT</Button>
      </Tooltip>
      <Tooltip placement="right" title={text}>
        <Button>Right</Button>
      </Tooltip>
      <Tooltip placement="rightBottom" title={
        <Button>RB</Button>
      </Tooltip>
    </div>
    <div style={{ marginLeft: buttonWidth, cle
      <Tooltip placement="bottomLeft" title={t
        <Button>BL</Button>
      </Tooltip>
      <Tooltip placement="bottom" title={text]
        <Button>Bottom</Button>
      </Tooltip>
      <Tooltip placement="bottomRight" title={
        <Button>BR</Button>
      </Tooltip>
    </div>
  </div>
);
```

Presets

pink   red   yellow   orange

cyan   green   blue   purple

geekblue   magenta   volcano

gold   lime

Custom

#f50   #2db7f5   #87d068

#108ee9

Colorful Tooltip ✎

We preset a series of colorful Tooltip styles for use in different situations.

```
import { Button, Divider, Space, Tooltip } fro
import React from 'react';

const colors = [
  'pink',
  'red',
  'yellow',
  'orange',
  'cyan',
  'green',
  'blue',
  'purple',
  'geekblue',
  'magenta',
  'volcano',
  'gold',
  'lime',
];

const customColors = ['#f50', '#2db7f5', '#87c

const App: React.FC = () => (
  <>
    <Divider orientation="left">Presets</Divic
    <Space wrap>
      {colors.map((color) => (
        <Tooltip title="prompt text" color={cc
          <Button>{color}</Button>
        </Tooltip>
      ))}
    </Space>
    <Divider orientation="left">Custom</Divide
    <Space wrap>
      {customColors.map((color) => (
        <Tooltip title="prompt text" color={cc
          <Button>{color}</Button>
        </Tooltip>
      ))}
    </Space>
  </>
);

export default App;
```

Auto Shift ✎

Auto adjust Popup and arrow position when Tooltip is close to the edge of the screen. Will be out of screen when exceed limitation.

```
import { Button, Tooltip } from 'antd';
import React from 'react';

const App: React.FC = () => {
  React.useEffect(() => {
    document.documentElement.scrollTop = docum
    document.documentElement.scrollLeft = docu
  }, []);

  return (
    <div>
      <div
        style={{
          width: '300vw',
          height: '300vh',
          display: 'flex',
          alignItems: 'center',
          justifyContent: 'center',
        }}
      >
        <Tooltip title="Thanks for using antd.
          <Button>Scroll The Window</Button>
        </Tooltip>
      </div>
    </div>
  );
};

export default App;
```

API

Property	Description	Type	Default
title	The text shown in the tooltip	ReactNode | () => ReactNode	-

Common API

The following APIs are shared by Tooltip, Popconfirm, Popover.

Property	Description	Type	Default	Ver
align	This value will be merged into placement's config, please refer to the settings rc-tooltip	object	-	
arrow	Change arrow's visible state and change whether the arrow is pointed at the center of target.	boolean | { pointAtCenter: boolean }	true	5.2
autoAdjustOverflow	Whether to adjust popup placement automatically when popup is off screen	boolean	true	
color	The background color	string	-	4.3
defaultOpen	Whether the floating tooltip card is open by default	boolean	false	4.2
destroyTooltipOnHide	Whether destroy tooltip when hidden	boolean	false	
getPopupContainer	The DOM container of the tip, the default behavior is to create a `div` element in `body`	(triggerNode: HTMLElement) => HTMLElement	() => document.body	
mouseEnterDelay	Delay in seconds, before tooltip is	number	0.1	

Property	Description	Type	Default	Ver
	shown on mouse enter			
mouseLeaveDelay	Delay in seconds, before tooltip is hidden on mouse leave	number	0.1	
overlayClassName	Class name of the tooltip card	string	–	
overlayStyle	Style of the tooltip card	object	–	
overlayInnerStyle	Style of the tooltip inner content	object	–	
placement	The position of the tooltip relative to the target, which can be one of `top` `left` `right` `bottom` `topLeft` `topRight` `bottomLeft` `bottomRight` `leftTop` `leftBottom` `rightTop` `rightBottom`	string	`top`	
trigger	Tooltip trigger mode. Could be multiple by passing an array	`hover` | `focus` | `click` | `contextMenu` | Array<string>	`hover`	
open	Whether the floating tooltip card is open or not. Use `visible` under 4.23.0 ([why?](#))	boolean	false	4.2
zIndex	Config `z-index` of Tooltip	number	–	
onOpenChange	Callback executed when visibility of the tooltip card is changed	(open: boolean) => void	–	4.2

Design Token

▼ Global Token

Token Name	Description	Type	Default Value
colorBgSpotlight	This color is used to draw the user's strong attention to the background color, and is currently only used in the background color of Tooltip.	string	☐ rgba(0, 0, 0, 0.85)
colorText	Default text color which comply with W3C standards, and this color is also the	string	☐ rgba(0, 0, 0, 0.88)

Token Name	Description	Type	Default Value
	darkest neutral color.		
colorTextLightSolid	Control the highlight color of text with background color, such as the text in Primary Button components.	`string`	☐ #fff
borderRadius	Border radius of base components	`number`	6
borderRadiusOuter	Outer border radius	`number`	4
borderRadiusXS	XS size border radius, used in some small border radius components, such as Segmented, Arrow and other components with small border radius.	`number`	2
boxShadowSecondary	Control the secondary box shadow style of an element.	`string`	0 6px 16px 0 rgba(0, 0, 0, 0.08), 0 3px 6px -4px rgba(0, 0, 0, 0.12), 0 9px 28px 8px rgba(0, 0, 0, 0.05)
controlHeight	The height of the basic controls such as buttons and input boxes in Ant Design	`number`	32
fontFamily	The font family of Ant Design prioritizes the default interface font of the system, and provides a set of alternative font libraries that are suitable for screen display to maintain the readability and readability of the font under different platforms and browsers, reflecting the friendly, stable and professional characteristics.	`string`	-apple-system, BlinkMacSystemFont, 'Segoe UI', Roboto, 'Helvetica Neue', Arial, 'Noto Sans', sans-serif, 'Apple Color Emoji', 'Segoe UI Emoji', 'Segoe UI Symbol', 'Noto Color Emoji'
fontSize	The most widely used font size in the design system, from which the text gradient will be derived.	`number`	14
lineHeight	Line height of text.	`number`	1.5714285714285714
motionDurationFast	Motion speed, fast speed. Used for small element animation interaction.	`string`	0.1s
motionEaseInOutCirc	Preset motion curve.	`string`	cubic-bezier(0.78, 0.14, 0.15, 0.86)
motionEaseOutCirc	Preset motion curve.	`string`	cubic-bezier(0.08, 0.82, 0.17, 1)
paddingSM	Control the small padding of the element.	`number`	12
paddingXS	Control the extra small padding of the element.	`number`	8
sizePopupArrow	The size of the component arrow	`number`	16
zIndexPopupBase	Base zIndex of component like FloatButton, Affix which can be cover by large popup	`number`	1000

Note

Please ensure that the child node of `Tooltip` accepts `onMouseEnter`, `onMouseLeave`, `onFocus`, `onClick` events.