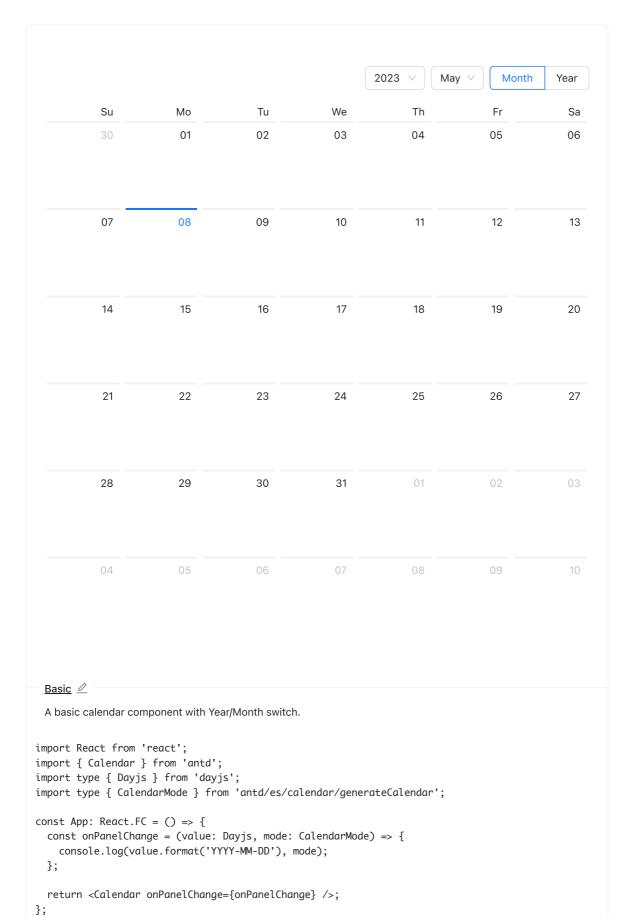
Calendar 🗸

Container for displaying data in calendar form.

When To Use

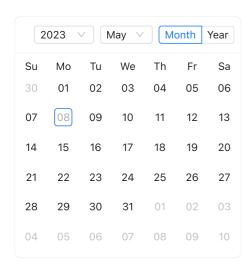
When data is in the form of dates, such as schedules, timetables, prices calendar, lunar calendar. This component also supports Year/Month switch.

Examples



export default App;

				2023 V N	May ∨ Mont	th Year
Su	Мо	Tu	We	Th	Fr	Sa
30	01	02	03	04	05	06
07	08 This is w This is u	09	10 This is w This is u This is e	11	12	13
14	15 This is w This is v This is e	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31	01	02	03
04	05	06	07	08 This is w This is u	09	This is w This is u This is e
port React fro port type { Bo port { Badge, port type { Do	can be rendered b	ı 'antd'; ı 'antd'; rjs';			er with the data	ı you need.
<pre>let listData; switch (value. case 8: listData =</pre>	= ['warning', cont 'success', cont	eent: 'This is eent: 'This is eent: 'This is	usual event	t.'},		



Card 🖉

Nested inside a container element for rendering in limited space.

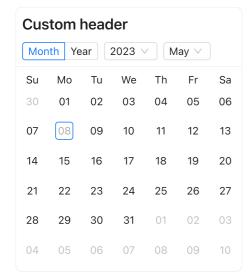
```
import { Calendar, theme } from 'antd';
import type { CalendarMode } from 'antd/es/calendar/generateCalendar';
import type { Dayjs } from 'dayjs';
import React from 'react';
const onPanelChange = (value: Dayjs, mode: CalendarMode) => {
  console.log(value.format('YYYY-MM-DD'), mode);
};
const App: React.FC = () \Rightarrow {
  const { token } = theme.useToken();
  const wrapperStyle: React.CSSProperties = {
    width: 300,
    border: `1px solid ${token.colorBorderSecondary}`,
    borderRadius: token.borderRadiusLG,
  };
  return (
    <div style={wrapperStyle}>
      <Calendar fullscreen={false} onPanelChange={onPanelChange} />
  );
};
export default App;
```

Мо 02	Tu 03	We	Th	Fr	S
02	03	04			
			05	06	0
09	10	11	12	13	1
16	17	18	19	20	2
23	24	25	26	27	2
30	31	01	02	03	0
	23	23 24	16 17 18 23 24 25	16 17 18 19 23 24 25 26	16 17 18 19 20 23 24 25 26 27

Selectable Calendar 🖉

A basic calendar component with Year/Month switch.

```
import React, { useState } from 'react';
import { Alert, Calendar } from 'antd';
import type { Dayjs } from 'dayjs';
import dayjs from 'dayjs';
const App: React.FC = () => {
  const [value, setValue] = useState(() => dayjs('2017-01-25'));
  const [selectedValue, setSelectedValue] = useState(() => dayjs('2017-01-25'));
  const onSelect = (newValue: Dayjs) => {
    setValue(newValue);
    setSelectedValue(newValue);
  const onPanelChange = (newValue: Dayjs) => {
    setValue(newValue);
  };
  return (
      <Alert message={`You selected date: ${selectedValue?.format('YYYY-MM-DD')}`} />
      <\!\!\text{Calendar value}\!=\!\!\{\text{value}\}\ \ \text{onSelect}\!=\!\!\{\text{onSelect}\}\ \ \text{onPanelChange}\!=\!\!\{\text{onPanelChange}\}\ \ /\!\!>
```



Customize Header 🖉

Customize Calendar header content.

```
import React from 'react';
import dayjs from 'dayjs';
import 'dayjs/locale/zh-cn';
import type { Dayjs } from 'dayjs';
import dayLocaleData from 'dayjs/plugin/localeData';
import { Calendar, Col, Radio, Row, Select, Typography, theme } from 'antd';
import type { CalendarMode } from 'antd/es/calendar/generateCalendar';
dayjs.extend(dayLocaleData);
const App: React.FC = () => {
 const { token } = theme.useToken();
 const onPanelChange = (value: Dayjs, mode: CalendarMode) => {
   console.log(value.format('YYYY-MM-DD'), mode);
 };
 const wrapperStyle = {
   width: 300,
   border: `1px solid ${token.colorBorderSecondary}`,
   borderRadius: token.borderRadiusLG,
 };
 return (
   <div style={wrapperStyle}>
     <Calendar
        fullscreen={false}
        headerRender=\{(\{ value, type, onChange, onTypeChange \}) => \{
         const start = 0;
          const end = 12;
          const monthOptions = [];
          let current = value.clone();
          const localeData = value.localeData();
          const months = [];
          for (let i = 0; i < 12; i++) {
           current = current.month(i);
            months.push(localeData.monthsShort(current));
         }
          for (let i = start; i < end; i++) {
            monthOptions.push(
              <Select.Option key={i} value={i} className="month-item">
                {months[i]}
              </Select.Option>,
         }
```

```
const year = value.year();
          const month = value.month();
          const options = [];
          for (let i = year - 10; i < year + 10; i += 1) {
            options.push(
              <Select.Option key={i} value={i} className="year-item">
              </Select.Option>,
            );
          }
          return (
            <div style={{ padding: 8 }}>
              <Typography.Title level={4}>Custom header</Typography.Title>
              <Row gutter={8}>
                <Col>
                  <Radio.Group
                    size="small"
                    onChange={(e) => onTypeChange(e.target.value)}
                    value={type}
                    <Radio.Button value="month">Month/Radio.Button>
                    <Radio.Button value="year">Year</Radio.Button>
                  </Radio.Group>
                </Col>
                <Col>
                  <Select
                    size="small"
                    dropdownMatchSelectWidth={false}
                    className="my-year-select"
                    value={year}
                    onChange={(newYear) => {
                      const now = value.clone().year(newYear);
                      onChange(now);
                    }}
 // The default locale is en-US, if you want to use other locale, just set locale in entry file gl
 // import dayjs from 'dayjs';
 // import 'dayjs/locale/zh-cn';
 // dayjs.locale('zh-cn');
 <Calendar
  dateCellRender={dateCellRender}
  monthCellRender={monthCellRender}
  onPanelChange={onPanelChange}
  onSelect={onSelect}
              </Row>
                                                                                   Default
 Property
                            Description
                                                           Type
        }}
        on Panel Change = \{on Panels Change\} the display of
                                                           function(date:
                            the date cell, the
 dateCellRender
                                                           Dayjs):
                            returned content will be
    </div>
                                                           ReactNode
 );
                            appended to the cell
};
                            Customize the display of
                                                            function(date:
export default App;
                            the date cell, the
 dateFullCellRender
                                                           Dayjs):
                            returned content will
                                                           ReactNode
                            override the cell
                            The date selected by
 defaultValue
                                                           <u>dayjs</u>
                            default
 disabledDate
                            Function that specifies
                                                           (currentDate:
                            the dates that cannot be
                                                           Dayjs) =>
                            selected, currentDate
                                                           boolean
                            is same dayjs object as
```

Property	Description	Туре	Default	V
	<pre>value prop which you shouldn't mutate it] (https://github.com/ant- design/ant- design/issues/30987)</pre>			
fullscreen	Whether to display in full-screen	boolean	true	
headerRender	Render custom header in panel	<pre>function(object: {value: Dayjs, type: string, onChange: f(), onTypeChange: f()})</pre>	-	
locale	The calendar's locale	object	<u>(default)</u>	
mode	The display mode of the calendar	month year	month	
monthCellRender	Customize the display of the month cell, the returned content will be appended to the cell	<pre>function(date: Dayjs): ReactNode</pre>	-	
monthFullCellRender	Customize the display of the month cell, the returned content will override the cell	<pre>function(date: Dayjs): ReactNode</pre>	-	
validRange	To set valid range	[<u>dayjs</u> , <u>dayjs</u>]	_	
value	The current selected date	<u>dayjs</u>	_	
onChange	Callback for when date changes	function(date: Dayjs)	_	
onPane1Change	Callback for when panel changes	<pre>function(date: Dayjs, mode: string)</pre>	-	
onSelect	Callback for when a date is selected	<pre>function(date: Dayjs)</pre>	-	

Design Token

▼ Global Token

Token Name	Description	Туре	Default Value
colorBgContainer	Container background color, e.g. default button, input box, etc. Be sure not to confuse this with `colorBgElevated`.	string	□ #ffffff
colorBgContainerDisabled	Control the background color of container in disabled state.	string	□rgba(0, 0, 0, 0.04)
coloricon	Weak action. Such as `allowClear` or Alert close button	string	□rgba(0, 0, 0, 0.45)

Token Name	Description	Туре	Default Value
colorIconHover	Weak action hover color. Such as `allowClear` or Alert close button	string	□rgba(0, 0, 0, 0.88)
colorLink	Control the color of hyperlink.	string	□ #1677ff
colorLinkActive	Control the color of hyperlink when clicked.	string	□ #0958d9
colorLinkHover	Control the color of hyperlink when hovering.	string	□ #69b1ff
colorPrimary	Brand color is one of the most direct visual elements to reflect the characteristics and communication of the product. After you have selected the brand color, we will automatically generate a complete color palette and assign it effective design semantics.	string	□ #1677ff
colorPrimaryHover	Hover state under the main color gradient.	string	□ #4096ff
colorSplit	Used as the color of separator, this color is the same as colorBorderSecondary but with transparency.	string	□rgba(5, 5, 5, 0.06)
colorText	Default text color which comply with W3C standards, and this color is also the darkest neutral color.	string	□rgba(0, 0, 0, 0.88)
colorTextDisabled	Control the color of text in disabled state.	string	□rgba(0, 0, 0, 0.25)
colorTextHeading	Control the font color of heading.	string	gba(0, 0, 0, 0.88)
colorTextLightSolid	Control the highlight color of text with background color, such as the text in Primary Button components.	string	#fff
borderRadiusLG	LG size border radius, used in some large border radius components, such as Card, Modal and other components.	number	8
borderRadiusSM	SM size border radius, used in small size components, such as Button, Input, Select and other input components in small size	number	4
controlHeight	The height of the basic controls such as buttons and input boxes in Ant Design	number	32
controlHeightLG	LG component height	number	40
controlHeightSM	SM component height	number	24
controlltemBgActive	Control the background color of control component item when active.	string	#e6f4ff
controlltemBgHover	Control the background color of control	string	□rgba(0, 0, 0, 0.04)

Token Name	Description	Туре	Default Value
	component item when hovering.		
controlPaddingHorizontal	Control the horizontal padding of an element.	number	12
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
fontSizeLG	Large font size	number	16
fontSizeSM	Small font size	number	12
fontWeightStrong	Control the font weight of heading components (such as h1, h2, h3) or selected item.	number	600
lineHeight	Line height of text.	number	1.5714285714285714
lineHeightLG	Line height of large text.	number	1.5
lineHeightSM	Line height of small text.	number	1.6666666666666667
lineType	Border style of base components	string	solid
lineWidth	Border width of base components	number	1
lineWidthBold	The default line width of the outline class components, such as Button, Input, Select, etc.	number	2
marginXS	Control the margin of an element, with a small size.	number	8
marginXXS	Control the margin of an element, with the smallest size.	number	4
motionDurationMid	Motion speed, medium speed. Used for medium element animation interaction.	string	0.2s
motionDurationSlow	Motion speed, slow speed. Used for large element animation interaction.	string	0.3s
padding	Control the padding of the element.	number	16
paddingSM	Control the small padding of the element.	number	12
paddingXS	Control the extra small padding of the	number	8

Token Name	Description	Туре	Default Value
	element.		
paddingXXS	Control the extra extra small padding of the element.	number	4
screenXS	Control the screen width of extra small screens.	number	480

FAQ

How to use Calendar with customize date library?

See <u>Use custom date library</u>

How to set locale for date-related components?

See <u>How to set locale for date-related components</u>

Date-related components locale is not working?

See FAQ <u>Date-related-components-locale-is-not-working?</u>