

# TimePicker

To select/input a time.

## When To Use

By clicking the input box, you can select a time from a popup panel.

## Examples



Select time 

### Basic

Click `TimePicker`, and then we could select or input a time in panel.

```
import React from 'react';
import { TimePicker } from 'antd';
import type { Dayjs } from 'dayjs';
import dayjs from 'dayjs';
import customParseFormat from 'dayjs/plugin/customParseFormat';

dayjs.extend(customParseFormat);

const onChange = (time: Dayjs, timeString: string) => {
  console.log(time, timeString);
};

const App: React.FC = () => (
  <TimePicker onChange={onChange} defaultOpen={true} />
);

export default App;
```

Select time 

### Under Control

`value` and `onChange` should be used together,

```
import React, { useState } from 'react';
import { TimePicker } from 'antd';
import type { Dayjs } from 'dayjs';

const App: React.FC = () => {
  const [value, setValue] = useState<Dayjs | null>(null);

  const onChange = (time: Dayjs) => {
    setValue(time);
  };

  return <TimePicker value={value} onChange={onChange} />;
};

export default App;
```

Select time 

Select time 

Select time 

### Three Sizes

The input box comes in three sizes. large is used in the form, while the medium size is the default.

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

const App: React.FC = () => (
  <Space wrap>
    <TimePicker defaultValue={dayjs('12:08:23')} size="large" />
    <TimePicker defaultValue={dayjs('12:08:23')} size="medium" />
    <TimePicker defaultValue={dayjs('12:08:23')} size="small" />
  </Space>
);

export default App;
```

Select time 

### disabled

A disabled state of the `TimePicker`.

```
import React from 'react';
import { TimePicker } from 'antd';
import dayjs from 'dayjs';
import customParseFormat from 'dayjs/plugin/customParseFormat';

dayjs.extend(customParseFormat);

const App: React.FC = () => <TimePicker defaultValue={dayjs('12:08:23')} disabled />;

export default App;
```

Select time 🕒

### Hour and minute [✎](#)

While part of `format` is omitted, the corresponding column in panel will disappear, too.

```
import React from 'react';
import { TimePicker } from 'antd';
import dayjs from 'dayjs';

const format = 'HH:mm';

const App: React.FC = () => <TimePicker default

export default App;
```

Select time 🕒

### interval option [✎](#)

Show stepped options by `hourStep`  
`minuteStep` `secondStep`.

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

const App: React.FC = () => <TimePicker minute

export default App;
```

Select time 🕒

### Addon [✎](#)

Render addon contents to time picker panel's bottom.

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

const App: React.FC = () => {
  const [open, setOpen] = useState(false);

  return (
    <TimePicker
      open={open}
      onOpenChange={setOpen}
      renderExtraFooter={() => (
        <Button size="small" type="primary" or
          OK
        </Button>
      )}
    />
  );
};

export default App;
```

Select time 🕒

Select time 🕒

Select time 🕒

### 12 hours [✎](#)

TimePicker of 12 hours format, with default format

`h:mm:ss a`.

```
import React from 'react';
import { Space, TimePicker } from 'antd';
import type { Dayjs } from 'dayjs';

const onChange = (time: Dayjs, timeString: str
  console.log(time, timeString);
};

const App: React.FC = () => (
  <Space wrap>
    <TimePicker use12Hours onChange={onChange}
      <TimePicker use12Hours format="h:mm:ss A"
      <TimePicker use12Hours format="h:mm a" onC
    </Space>
  );

export default App;
```

Start time → End time

### Time Range Picker

Use time range picker with

`TimePicker.RangePicker`.

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

const App: React.FC = () => <TimePicker.RangePicker />

export default App;
```

Select time

Select time

Start time → End time

Start time → End time

### Status

Add status to TimePicker with `status`, which could be `error` or `warning`.

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

const App: React.FC = () => (
  <Space direction="vertical">
    <TimePicker status="error" />
    <TimePicker status="warning" />
    <TimePicker.RangePicker status="error" />
    <TimePicker.RangePicker status="warning" />
  </Space>
);

export default App;
```

Select time

Start time → End time

### Bordered-less

Bordered-less style component.

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

const { RangePicker } = TimePicker;

const App: React.FC = () => (
  <>
    <TimePicker bordered={false} />
    <RangePicker bordered={false} />
  </>
);

export default App;
```

## API

```
import dayjs from 'dayjs';
import customParseFormat from 'dayjs/plugin/customParseFormat'
```

```
dayjs.extend(customParseFormat)
```

```
<TimePicker defaultValue={dayjs('13:30:56', 'HH:mm:ss')} />;
```

Property	Description	Type	Default	Version
<code>allowClear</code>	Whether allow	<code>boolean</code>	<code>true</code>	

Property	Description	Type	Default	Version
	clearing text			
autoFocus	If get focus when component mounted	boolean	false	
bordered	Whether has border style	boolean	true	
className	The className of picker	string	-	
clearIcon	The custom clear icon	ReactNode	-	
clearText	The clear tooltip of icon	string	clear	
defaultValue	To set default time	<a href="#">dayjs</a>	-	
disabled	Determine whether the TimePicker is disabled	boolean	false	
disabledTime	To specify the time that cannot be selected	<a href="#">DisabledTime</a>	-	4.19
format	To set the time format	string	<input type="text" value="HH:mm:ss"/>	
getPopupContainer	To set the container of the floating layer, while the default is to create a div element in body	function(trigger)	-	
hideDisabledOptions	Whether hide the options that can not be selected	boolean	false	
hourStep	Interval between hours in picker	number	1	
inputReadOnly	Set the <input type="text" value="readonly"/> attribute of the input tag (avoids virtual keyboard on touch devices)	boolean	false	
minuteStep	Interval between minutes in picker	number	1	
open	Whether to popup panel	boolean	false	
placeholder	Display when there's no value	string   [string, string]	<input type="text" value="Select a time"/>	
placement	The position where the selection box pops up	<input type="text" value="bottomLeft"/> <input type="text" value="bottomRight"/> <input type="text" value="topLeft"/> <input type="text" value="topRight"/>	bottomLeft	

Property	Description	Type	Default	Version
popupClassName	The className of panel	string	-	
popupStyle	The style of panel	CSSProperties	-	
renderExtraFooter	Called from time picker panel to render some addon to its bottom	() => ReactNode	-	
secondStep	Interval between seconds in picker	number	1	
showNow	Whether to show <code>Now</code> button on panel	boolean	-	4.4.0
size	To determine the size of the input box, the height of <code>large</code> and <code>small</code> , are 40px and 24px respectively, while default size is 32px	<code>large</code>   <code>middle</code>   <code>small</code>	-	
status	Set validation status	'error'   'warning'   'success'   'validating'	-	4.19.0
suffixIcon	The custom suffix icon	ReactNode	-	
use12Hours	Display as 12 hours format, with default format <code>h:mm:ss a</code>	boolean	false	
value	To set time	<a href="#">day.js</a>	-	
onChange	A callback function, can be executed when the selected time is changing	function(time: dayjs, timeString: string): void	-	
onOpenChange	A callback function which will be called while panel opening/closing	(open: boolean) => void	-	
onSelect	A callback function, executes when a value is selected	function(time: dayjs): void	-	

## DisabledTime

```

type DisabledTime = (now: Dayjs) => {
  disabledHours?: () => number[];
  disabledMinutes?: (selectedHour: number) => number[];
}

```

```
disabledSeconds?: (selectedHour: number, selectedMinute: number) => number[];
};
```

Methods

Name	Description	Version
blur()	Remove focus	
focus()	Get focus	

RangePicker

Same props from [RangePicker](#) of DatePicker. And includes additional props:

Property	Description	Type	Default	Version
disabledTime	To specify the time that cannot be selected	<a href="#">RangeDisabledTime</a>	-	4.19.0
order	Order start and end time	boolean	true	4.1.0

RangeDisabledTime

```
type RangeDisabledTime = (
  now: Dayjs,
  type = 'start' | 'end',
) => {
  disabledHours?: () => number[];
  disabledMinutes?: (selectedHour: number) => number[];
  disabledSeconds?: (selectedHour: number, selectedMinute: number) => number[];
};
```

Design Token

▼ Global Token

Token Name	Description	Type	Default Value
colorBgContainer	Container background color, e.g: default button, input box, etc. Be sure not to confuse this with `colorBgElevated`.	string	<input type="checkbox"/> #ffffff
colorBgContainerDisabled	Control the background color of container in disabled state.	string	<input type="checkbox"/> rgba(0, 0, 0, 0.04)
colorBgElevated	Container background color of the popup layer, in dark mode the color value of this token will be a little brighter than `colorBgContainer`. E.g: modal, pop-up, menu, etc.	string	<input type="checkbox"/> #ffffff
colorBorder	Default border color, used to separate different elements, such as: form	string	<input type="checkbox"/> #d9d9d9

Token Name	Description	Type	Default Value
	separator, card separator, etc.		
colorError	Used to represent the visual elements of the operation failure, such as the error Button, error Result component, etc.	string	<code>#ff4d4f</code>
colorErrorOutline	Control the outline color of input component in error state.	string	<code>rgba(255, 38, 5, 0.06)</code>
colorIcon	Weak action. Such as `allowClear` or Alert close button	string	<code>rgba(0, 0, 0, 0.45)</code>
colorIconHover	Weak action hover color. Such as `allowClear` or Alert close button	string	<code>rgba(0, 0, 0, 0.88)</code>
colorLink	Control the color of hyperlink.	string	<code>#1677ff</code>
colorLinkActive	Control the color of hyperlink when clicked.	string	<code>#0958d9</code>
colorLinkHover	Control the color of hyperlink when hovering.	string	<code>#69b1ff</code>
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	<code>#1677ff</code>
colorPrimaryBorder	The stroke color under the main color gradient, used on the stroke of components such as Slider.	string	<code>#91caff</code>
colorPrimaryHover	Hover state under the main color gradient.	string	<code>#4096ff</code>
colorSplit	Used as the color of separator, this color is the same as colorBorderSecondary but with transparency.	string	<code>rgba(5, 5, 5, 0.06)</code>
colorText	Default text color which comply with W3C standards, and this color is also the darkest neutral color.	string	<code>rgba(0, 0, 0, 0.88)</code>
colorTextDescription	Control the font color of text description.	string	<code>rgba(0, 0, 0, 0.45)</code>
colorTextDisabled	Control the color of text in disabled state.	string	<code>rgba(0, 0, 0, 0.25)</code>
colorTextHeading	Control the font color of heading.	string	<code>rgba(0, 0, 0, 0.88)</code>
colorTextLightSolid	Control the highlight color of text with background color, such as the text in Primary Button components.	string	<code>#fff</code>
colorTextPlaceholder	Control the color of placeholder text.	string	<code>rgba(0, 0, 0, 0.25)</code>

Token Name	Description	Type	Default Value
colorWarning	Used to represent the warning map token, such as Notification, Alert, etc. Alert or Control component(like Input) will use these map tokens.	string	<input type="checkbox"/> #faad14
colorWarningOutline	Control the outline color of input component in warning state.	string	<input type="checkbox"/> rgba(255, 215, 5, 0.1)
borderRadius	Border radius of base components	number	6
borderRadiusLG	LG size border radius, used in some large border radius components, such as Card, Modal and other components.	number	8
borderRadiusOuter	Outer border radius	number	4
borderRadiusSM	SM size border radius, used in small size components, such as Button, Input, Select and other input components in small size	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
controlHeightLG	LG component height	number	40
controlHeightSM	SM component height	number	24
controlItemBgActive	Control the background color of control component item when active.	string	<input type="checkbox"/> #e6f4ff
controlItemBgHover	Control the background color of control component item when hovering.	string	<input type="checkbox"/> rgba(0, 0, 0, 0.04)
controlOutline	Control the outline color of input component.	string	<input type="checkbox"/> rgba(5, 145, 255, 0.1)
controlOutlineWidth	Control the outline width of input component.	number	2
controlPaddingHorizontal	Control the horizontal padding of an element.	number	12
controlPaddingHorizontalSM	Control the horizontal padding of an element with a small-medium size.	number	8
fontFamily	The font family of Ant Design prioritizes the default interface font of the system, and provides a set of alternative font	string	-apple-system, BlinkMacSystemFont, 'Segoe UI', Roboto, 'Helvetica Neue',



Token Name	Description	Type	Default Value
	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.		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
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
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
motionEaseInOutCirc	Preset motion curve.	string	cubic-bezier(0.78, 0.14, 0.15, 0.86)
motionEaseInQuint	Preset motion curve.	string	cubic-bezier(0.755, 0.05, 0.855, 0.06)
motionEaseOutCirc	Preset motion curve.	string	cubic-bezier(0.08, 0.82, 0.17, 1)
motionEaseOutQuint	Preset motion curve.	string	cubic-bezier(0.23, 1, 0.32, 1)
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 element.	number	8

Token Name	Description	Type	Default Value
paddingXXS	Control the extra extra small padding of the element.	number	4
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

## FAQ

- [How to use TimePicker with customize date library like dayjs](#)