

A basic widget for getting the user input is a text field. Keyboard and mouse can be used for providing or changing data.

## When To Use

- A user input in a form field is needed.
- A search input is required.

## API

### Input

Property	Description	Type	Default	Version
addonAfter	The label text displayed after (on the right side of) the input field	ReactNode	-	
addonBefore	The label text displayed before (on the left side of) the input field	ReactNode	-	
allowClear	If allow to remove input content with clear icon	boolean   { clearIcon: ReactNode }	false	
bordered	Whether has border style	boolean	true	4.5.0
defaultValue	The initial input content	string	-	
disabled	Whether the input is disabled	boolean	false	
id	The ID for input	string	-	
maxLength	The max length	number	-	
showCount	Whether show text count	boolean   { formatter: ({ count: number, maxLength?: number }) => ReactNode }	false	4.18.0
status	Set validation status	'error'   'warning'	-	4.19.0
prefix	The prefix icon for the Input	ReactNode	-	
size	The size of the input box. Note: in the context of a form, the <code>middle</code> size is used	<code>large</code>   <code>middle</code>   <code>small</code>	-	
suffix	The suffix icon for the Input	ReactNode	-	
type	The type of input, see: <a href="#">MDN</a> ( use <code>Input.TextArea</code> instead of <code>type="textarea"</code> )	string	text	
value	The input content value	string	-	
onChange	Callback when user input	function(e)	-	
onPressEnter	The callback function that is	function(e)	-	

	triggered when Enter key is pressed			
--	-------------------------------------	--	--	--

When `Input` is used in a `Form.Item` context, if the `Form.Item` has the `id` and `options` props defined then `value`, `defaultValue`, and `id` props of `Input` are automatically set.

The rest of the props of `Input` are exactly the same as the original [input](#).

### Input.TextArea

Property	Description	Type	Default	Version
<code>allowClear</code>	If allow to remove input content with clear icon	boolean	false	
<code>autoSize</code>	Height autosize feature, can be set to true   false or an object { minRows: 2, maxRows: 6 }	boolean   object	false	
<code>bordered</code>	Whether has border style	boolean	true	4.5.0
<code>defaultValue</code>	The initial input content	string	-	
<code>maxLength</code>	The max length	number	-	4.7.0
<code>showCount</code>	Whether show text count	boolean   { formatter: ({ count: number, maxLength?: number }) => string }	false	4.7.0 (formatter: 4.10.0)
<code>value</code>	The input content value	string	-	
<code>onPressEnter</code>	The callback function that is triggered when Enter key is pressed	function(e)	-	
<code>onResize</code>	The callback function that is triggered when resize	function({ width, height })	-	

The rest of the props of `Input.TextArea` are the same as the original [textarea](#).

### Input.Search

Property	Description	Type	Default
<code>enterButton</code>	Whether to show an enter button after input. This property conflicts with the <code>addonAfter</code> property	boolean   ReactNode	false
<code>loading</code>	Search box with loading	boolean	false
<code>onSearch</code>	The callback function triggered when you click on the search-icon, the clear-icon or press the Enter key	function(value, event)	-

Supports all props of `Input`.

### Input.Group

--	--	--	--

Property	Description	Type	Default
compact	Whether use compact style	boolean	false
size	The size of <code>Input.Group</code> specifies the size of the included <code>Input</code> fields. Available: large default small	string	default

```

<Input.Group>
  <input />
  <input />
</Input.Group>

```

### Input.Password

Property	Description	Type	Default	Version
iconRender	Custom toggle button	(visible) => ReactNode	(visible) => (visible ? <EyeOutlined /> : <EyeInvisibleOutlined />)	4.3.0
visibilityToggle	Whether show toggle button	boolean	true	

### Input Methods

Name	Description	Parameters	Version
blur	Remove focus	-	
focus	Get focus	(option?: { preventScroll?: boolean, cursor?: 'start'   'end'   'all' })	option - 4.10.0

## FAQ

### Why Input lose focus when change `prefix/suffix`

When Input dynamic add or remove `prefix/suffix` will make React recreate the dom structure and new input will be not focused. You can set an empty `<span />` element to keep the dom structure:

```

const suffix = condition ? <Icon type="smile" /> : <span />;

<Input suffix={suffix} />;

```

### Why TextArea in control can make `value` exceed `maxLength` ?

When in control, component should show as what it set to avoid submit value not align with store value in Form.