

Simple rectangular container.

## When To Use

A card can be used to display content related to a single subject. The content can consist of multiple elements of varying types and sizes.

## API

```
<Card title="Card title">Card content</Card>
```

### Card

Property	Description	Type	Default	Version
actions	The action list, shows at the bottom of the Card	Array<ReactNode>	-	
activeTabKey	Current TabPane's key	string	-	
bodyStyle	Inline style to apply to the card content	CSSProperties	-	
bordered	Toggles rendering of the border around the card	boolean	true	
cover	Card cover	ReactNode	-	
defaultActiveTabKey	Initial active TabPane's key, if <code>activeTabKey</code> is not set	string	-	
extra	Content to render in the top-right corner of the card	ReactNode	-	
headStyle	Inline style to apply to the card head	CSSProperties	-	
hoverable	Lift up when hovering card	boolean	false	
loading	Shows a loading indicator while the contents of the card are being fetched	boolean	false	
size	Size of card	default   small	default	
tabBarExtraContent	Extra content in tab bar	ReactNode	-	
tabList	List of TabPane's head	Array<{key: string, tab: ReactNode}>	-	
tabProps	<a href="#">Tabs</a>	-	-	
title	Card title	ReactNode	-	
type	Card style type, can be set to <code>inner</code>	string	-	

	or not set			
onTabChange	Callback when tab is switched	(key) => void	-	

### Card.Grid

Property	Description	Type	Default	Version
className	The className of container	string	-	
hoverable	Lift up when hovering card grid	boolean	true	
style	The style object of container	CSSProperties	-	

### Card.Meta

Property	Description	Type	Default	Version
avatar	Avatar or icon	ReactNode	-	
className	The className of container	string	-	
description	Description content	ReactNode	-	
style	The style object of container	CSSProperties	-	
title	Title content	ReactNode	-	