App

New App Component which provide global style & static function replacement.

When To Use

Static function in React 18 concurrent mode will not well support. In v5, we recommend to use hooks for the static replacement. But it will make user manual work on define this.

Examples

Open message Open modal Open notification

basic

Static method for `message`, `notification`, `modal`.

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

// Sub page
const MyPage = () => {
  const { message, modal, notification } = App.useApp();

  const showMessage = () => {
    message.success('Success!');
  };

  const showModal = () => {
    modal.warning({
      title: 'This is a warning message',
      content: 'some messages...some messages...',
    });
  };

  const showNotification = () => {
    notification.info({
      message: `Notification topLeft`,
      description: 'Hello, Ant Design!!',
      placement: 'topLeft',
    });
  };

  return (
    <Space>
      <Button type="primary" onClick={showMessage}>
        Open message
      </Button>
      <Button type="primary" onClick={showModal}>
        Open modal
      </Button>
      <Button type="primary" onClick={showNotification}>
        Open notification
      </Button>
    </Space>
  );
};

// Entry component
export default () => (
  <App>
    <MyPage />
  </App>
);
```

How to use

Basic usage

App provides upstream and downstream method calls through `Context`, because useApp needs to be used as a subcomponent, we recommend encapsulating App at the top level in the application.

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

```
const MyPage: React.FC = () => {
  const { message, notification, modal } = App.useApp();
  message.success('Good!');
  notification.info({ message: 'Good' });
  modal.warning({ title: 'Good' });
  // ....
  // other message, notification, modal static function
  return <div>Hello word</div>;
};

const MyApp: React.FC = () => (
  <App>
    <MyPage />
  </App>
);

export default MyApp;
```

Note: App.useApp must be available under App.

Sequence with ConfigProvider

The App component can only use the token in the `ConfigProvider`, if you need to use the Token, the ConfigProvider and the App component must appear in pairs.

```
<ConfigProvider theme={{ ... }}>
  <App>
    ...
  </App>
</ConfigProvider>
```

Embedded usage scenarios (if not necessary, try not to do nesting)

```
<App>
  <Space>
    ...
    <App>...</App>
  </Space>
</App>
```

Global scene (redux scene)

```
// Entry component
import { App } from 'antd';
import type { MessageInstance } from 'antd/es/message/interface';
import type { ModalStaticFunctions } from 'antd/es/modal/confirm';
import type { NotificationInstance } from 'antd/es/notification/interface';

let message: MessageInstance;
let notification: NotificationInstance;
let modal: Omit<ModalStaticFunctions, 'warn'>;
```

```
export default () => {
  const staticFunction = App.useApp();
  message = staticFunction.message;
  modal = staticFunction.modal;
  notification = staticFunction.notification;
  return null;
};

export { message, notification, modal };


// sub page
import { Button, Space } from 'antd';
import React from 'react';
import { message } from './store';

export default () => {
  const showMessage = () => {
    message.success('Success!');
  };

  return (
    <Space>
      <Button type="primary" onClick={showMessage}>
        Open message
      </Button>
    </Space>
  );
};
```

API

App

Property	Description	Type	Default	Version
message	Global config for Message	[MessageConfig]()	-	5.3.0
notification	Global config for Notification	[NotificationConfig]()	-	5.3.0

Design Token

▾ Global Token

Token Name	Description	Type	Default Value
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)
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	`string`	-apple-system, BlinkMacSystemFont, 'Segoe UI', Roboto, 'Helvetica Neue', Arial, 'Noto Sans', sans-serif, 'Apple

Token Name	Description	Type	Default Value
	for screen display to maintain the readability and readability of the font under different platforms and browsers, reflecting the friendly, stable and professional characteristics.		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