Guides Themes

Version: 6.x

Themes

Themes allow you to change the colors of various components provided by React Navigation. You can use themes to:

- Customize the colors match your brand
- Provide light and dark themes based on the time of the day or user preference

Basic usage

To pass a custom theme, you can pass the `theme` prop to the navigation container.

```
import * as React from 'react';
import { NavigationContainer, DefaultTheme } from '@react-navigation/native';

const MyTheme = {
  ...DefaultTheme,
  colors: {
    ...DefaultTheme.colors,
    primary: 'rgb(255, 45, 85)',
  },
};

export default function App() {
  return (
    <NavigationContainer theme={MyTheme}>{/* content */}</NavigationContainer>
  );
}
```

Try this example on Snack

You can change the theme prop dynamically and all the components will automatically update to reflect the new theme. If you haven't provided a `theme` prop, the default theme will be used.

A theme is a JS object containing a list of colors to use. It contains the following properties:

- `dark` (`boolean`): Whether this is a dark theme or a light theme
- `colors` (`object`): Various colors used by react navigation components:
  - `primary` (`string`): The primary color of the app used to tint various elements. Usually you'll want to use your brand color for this.
  - `background` (`string`): The color of various backgrounds, such as background color for the screens.
  - `card` (`string`): The background color of card-like elements, such as headers, tab bars etc.
  - `text` (`string`): The text color of various elements.
  - `border` (`string`): The color of borders, e.g. header border, tab bar border etc.
  - `notification` (`string`): The color of Tab Navigator badge.

When creating a custom theme, you will need to provide all of these properties.

Example theme:

```
const MyTheme = {
  dark: false,
  colors: {
    primary: 'rgb(255, 45, 85)',
    background: 'rgb(242, 242, 242)',
    card: 'rgb(255, 255, 255)',
    text: 'rgb(28, 28, 30)',
    border: 'rgb(199, 199, 204)',
    notification: 'rgb(255, 69, 58)',
  },
};
```

Providing a theme will take care of styling of all the official navigators. React Navigation also provides several tools to help you make your customizations of those navigators and the screens within the navigators can use the theme too.

Built-in themes

As operating systems add built-in support for light and dark modes, supporting dark mode is less about keeping hip to trends and more about conforming to the average user expectations for how apps should work. In order to provide support for light and dark mode in a way that is reasonably consistent with the OS defaults, these themes are built in to React Navigation.

You can import the default and dark themes like so:

```
import { DefaultTheme, DarkTheme } from '@react-navigation/native';
```

Using the operating system preferences

On iOS 13+ and Android 10+, you can get user's preferred color scheme (`'dark'` or `'light'`) with the (Appearance API).

```
import { useColorScheme } from 'react-native';
import {
  NavigationContainer,
  DefaultTheme,
  DarkTheme,
} from '@react-navigation/native';

export default () => {
  const scheme = useColorScheme();

  return (
    <NavigationContainer theme={scheme === 'dark' ? DarkTheme : DefaultTheme}>
      {/* content */}
    </NavigationContainer>
  );
};
```

Try this example on Snack

Using the current theme in your own components

To gain access to the theme in any component that is rendered inside the navigation container:, you can use the `useTheme` hook. It returns the theme object:

```
import * as React from 'react';
import { TouchableOpacity, Text } from 'react-native';
import { useTheme } from '@react-navigation/native';

// Black background and white text in light theme, inverted on dark theme
```

```
function MyButton() {
  const { colors } = useTheme();

  return (
    <TouchableOpacity style={{ backgroundColor: colors.card }}>
      <Text style={{ color: colors.text }}>Button!</Text>
    </TouchableOpacity>
  );
}
```

Try this example on Snack

Edit this page