Advanced 进阶

在这里，你可以找到如何在你的自定义组件中使用系统（system）的例子。

将 `sx` 属性添加到你的自定义组件

The `unstable_styleFunctionSx` utility adds the support for the [`sx` prop](#) to your own components. Normally you would use the `Box` component from `@material-ui/core` at the root of your component tree. If you would like to use the system independently from Material-UI, the `unstable_styleFunctionSx` utility will give you the same capabilities, while having a smaller bundle size. Normally you would use the `Box` component from `@mui/material` at the root of your component tree. If you would like to use the system independently from MUI, the `unstable_styleFunctionSx` utility will give you the same capabilities, while having a smaller bundle size. Normally you would use the `Box` component from `@mui/material` at the root of your component tree. If you would like to use the system independently from MUI, the `unstable_styleFunctionSx` utility will give you the same capabilities, while having a smaller bundle size.

{{"demo": "StyleFunctionSxDemo.js"}}

使用独立的系统工具集

如果你在自定义组件中只需要系统中的一些元素，你可以直接使用和组合不同的风格功能，并将其作为组件属性访问。 You might use this approach if you need smaller bundle size and better performance than using Box, for the price of using a subset of what the [`sx` prop](#) supports, and a different API.

{{"demo": "CombiningStyleFunctionsDemo.js", "defaultCodeOpen": true}}