

Material UI

Làm quen với thư viện Material UI



- ❖ Material UI là thư viện số 1 hiện tại kết hợp với react, tích hợp sẵn các component như : button, table, tab...
- ❖ Material UI khá hay vì nó cho phép ta custom và tạo ra một bộ theme cho chính trang web của mình
- ❖ Sau đây ta sẽ đi tìm hiểu về material UI và cách sử dụng
- ❖ Website: <https://material-ui.com/>
- ❖ Để sử dụng , tiến hành cài đặt material ui vào project : `npm i @material-ui/core`

Làm quen với thư viện Material UI



CYBERSOFT
ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH



- ❖ Sử dụng component có sẵn : mỗi component có một cách sử dụng khác nhau và được mô tả chi tiết trên trang của material ui, dưới đây là một ví dụ đơn giản sử dụng component Button

```
import React from "react";
import { Button } from "@material-ui/core";
const app = () => {
  return (
    <div>
      <h1>Button Demo</h1>
      <Button variant="contained" color="primary">
        Submit
      </Button>
    </div>
  );
};
export default app;
```

Button Demo

SUBMIT

Làm quen với thư viện Material UI



CYBERSOFT
ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH



- ❖ Tiếp theo, ta sẽ tìm hiểu về khái niệm JSS (CSS-in-JS)
 - ❖ Trước giờ, ta style component bằng sass hoặc css, có một số ưu điểm là dễ dùng, quen thuộc.. tuy nhiên nó mắc phải một số vấn đề: không linh hoạt(ví dụ , ta muốn truyền một biến vào để xét chiều rộng cho div) thì làm không được và không có tính đóng gói (encapsulation) trong component.
 - ❖ JSS được sinh ra để giải quyết vấn đề này, xây dựng css như một object javascript.
 - ❖ Các hệ thống lớn hiện tại sử dụng JSS rất nhiều, tất nhiên không ngoại trừ material-ui

Làm quen với thư viện Material UI



- ❖ Sử dụng JSS trong Material UI với withStyles
- ❖ Tạo một file style.js để định nghĩa jss cho component, mỗi component sẽ có một file style riêng.
- ❖ Ở đây theme đơn giản là một object để quy định nguyên bộ theme của material (màu, font chữ...)
- ❖ (2) là cách viết nếu muốn gọi lồng cấp, trong class content gọi ra span để chỉnh

```
import { makeStyles } from "@material-ui/core";

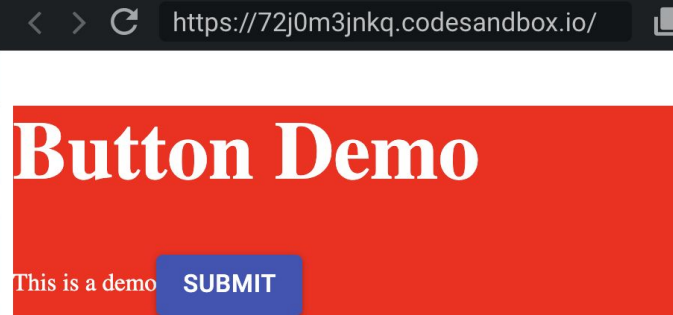
const useStyles = makeStyles(theme => ({
  content: {
    backgroundColor: "red",
    color: "#ffffff",
    "& span": {
      fontSize: 15
    }
  },
  title: {
    fontSize: 50
  }
}));

export default useStyles;
```

Làm quen với thư viện Material UI

❖ Ở component , ta sử dụng như sau

```
import React from "react";
import { Button } from "@material-ui/core";
import useStyles from "../style";
const App = () => {
  const classes = useStyles();
  return (
    <div className={classes.content}>
      <h1 className={classes.title}>Button Demo</h1>
      <span>This is a demo</span>
      <Button variant="contained" color="primary">
        Submit
      </Button>
    </div>
  );
};
export default App;
```



Làm quen với thư viện Material UI



- ❖ Tiếp theo , ta sẽ tìm hiểu về Typography trong material-ui, component này cho phép chúng ta thống nhất toàn bộ định dạng text trong trang web

```
pp.js x JS style.js JS index.js
1 import React from "react";
2 import { Button, Typography } from "@material-ui/core";
3 import useStyles from "../style";
4 const App = () => {
5   const classes = useStyles();
6   return (
7     <div className={classes.content}>
8       <Typography color="textPrimary" variant="h2" component="h1">
9         | Button Demo
10      </Typography>
11      <Typography variant="body1" component="span" color="primary">
12        | This is demo
13      </Typography>
14
15      <Button variant="contained" color="primary">
16        | Submit
17      </Button>
18    </div>
19  );
20 };
21 export default App;
22
```

Variant: định dạng text

color: màu text

component: tag html sẽ được render ra giao diện

=> toàn bộ định dạng text, màu, đều được định nghĩa sẵn trong theme của material. Ví dụ, nếu variant là h2, thì sẽ có font size bao nhiêu, font-weight bao nhiêu...color= "primary" là có màu xanh, còn color="textPrimary" là có màu đen, tất cả những cái này đều định nghĩa sẵn trong theme.

Làm quen với thư viện Material UI



- ❖ Trong trường hợp, ta không muốn sử dụng theme của material ui, mà muốn tự custome thì như thế nào ?
- ❖ Tạo ra một file theme để định nghĩa lại, trong đó, palette là màu, typography là định dạng text
- ❖ 2 thuộc tính fontSize và fontFamily nằm cuối và riêng biệt có nghĩa là, chỉ cần sử dụng <Typography> thì text sẽ mặc định có fontSize: 14 và fontFamily như v

```
import { createMuiTheme } from '@material-ui/core/styles';  
  
export const theme = createMuiTheme({  
  palette: {  
    common: {  
      black: 'rgb(47, 46, 46, 1)',  
      white: 'rgb(255, 255, 255, 1)'  
    },  
    background: {  
      paper: 'rgb(255, 255, 255, 1)',  
      default: 'rgb(251, 251, 251, 1)'  
    },  
    primary: {  
      light: 'rgb(255, 255, 255, 1)',  
      main: 'rgb(227, 141, 3, 1)',  
      dark: 'rgb(226, 112, 1, 1)',  
      contrastText: 'rgb(255, 255, 255, 1)'  
    },  
    secondary: {  
      light: 'rgb(47, 46, 46, 1)',  
      main: 'rgb(86, 85, 84, 1)',  
      dark: 'rgb(47, 46, 46, 1)',  
      contrastText: '#fff'  
    },  
    text: {  
      primary: 'rgb(47, 46, 46, 1)',  
      secondary: 'rgb(86, 85, 84, 1)',  
      disabled: 'rgb(155, 155, 155, 1)',  
      hint: 'rgb(184, 184, 184, 0.38)'  
    },  
  },  
  typography: {  
    h5: {  
      fontWeight: 600  
    },  
    h6: {  
      fontSize: 18,  
      fontWeight: 600  
    },  
    body1: {  
      fontSize: 14,  
      fontWeight: 600  
    },  
    body2: {  
      fontWeight: 400  
    },  
    fontSize: 14,  
    fontFamily: 'Open Sans', sans-serif  
  }  
});
```


Làm quen với thư viện Material UI



- ❖ Để bộ theme này có thể sử dụng trong toàn ứng dụng, ta sẽ bọc component lớn nhất lại, trường hợp này là app component

```
import useStyles from './style';
import { ThemeProvider } from '@material-ui/styles';
import { theme } from './theme'
const App = () => {
  const classes = useStyles();
  return (
    <ThemeProvider theme={theme}>
      <div className={classes.content}>
        <Typography color="textPrimary" variant="h2" component="h1">
          Button Demo
        </Typography>
        <Typography variant="body1" component="span" color="primary">
          This is demo
        </Typography>
        <Button variant="contained" color="primary">
          Submit
        </Button>
      </div>
    </ThemeProvider>
  );
};
export default App;
```