

```
import { createContext, useState, useMemo } from "react";
import { createTheme } from "@mui/material/styles";
```

```
export const tokens = (mode) => ({
  ...(mode === "dark"
    ? {
        gray: {
          100: "#e0e0e0",
          200: "#c2c2c2",
          300: "#a3a3a3",
          400: "#858585",
          500: "#666666",
          600: "#525252",
          700: "#3d3d3d",
          800: "#292929",
          900: "#141414",
        },
        primary: {
          100: "#d0d1d5",
          200: "#a1a4ab",
          300: "#727681",
          400: "#1f2a40",
          500: "#141b2d",
          600: "#101624",
          700: "#0c101b",
          800: "#080b12",
          900: "#040509",
        },
        orangeAccent: {
          100: "#fdefde",
          200: "#fadbfe",
          300: "#f8d09d",
          400: "#f5c07d",
          500: "#f3b05c",
          600: "#c28d4a",
          700: "#926a37",
          800: "#614625",
          900: "#312312",
        },
        redAccent: {
          100: "#f8dcdb",
          200: "#f1b9b7",
          300: "#e99592",
          400: "#e2726e",
          500: "#db4f4a",
          600: "#af3f3b",

```

```
    700: "#832f2c",
    800: "#58201e",
    900: "#2c100f",
  },
  blueAccent: {
    100: "#e1e2fe",
    200: "#c3c6fd",
    300: "#a4a9fc",
    400: "#868dfb",
    500: "#6870fa",
    600: "#535ac8",
    700: "#3e4396",
    800: "#2a2d64",
    900: "#151632",
  },
}
: {
  gray: {
    100: "#141414",
    200: "#292929",
    300: "#3d3d3d",
    400: "#525252",
    500: "#666666",
    600: "#858585",
    700: "#a3a3a3",
    800: "#c2c2c2",
    900: "#e0e0e0",
  },
  primary: {
    100: "#040509",
    200: "#949494",
    300: "#0c101b",
    400: "#ffffff",
    500: "#141b2d",
    600: "#434957",
    700: "#727681",
    800: "#a1a4ab",
    900: "#ffffff",
  },
  orangeAccent: {
    100: "#312312",
    200: "#614625",
    300: "#926a37",
    400: "#c28d4a",
    500: "#f3b05c",
    600: "#f5c07d",
```

```

    700: "#f8d09d",
    800: "#fadbfe",
    900: "#fdefde",
  },
  redAccent: {
    100: "#2c100f",
    200: "#58201e",
    300: "#832f2c",
    400: "#af3f3b",
    500: "#db4f4a",
    600: "#e2726e",
    700: "#e99592",
    800: "#f1b9b7",
    900: "#f8dcdb",
  },
  blueAccent: {
    100: "#151632",
    200: "#2a2d64",
    300: "#3e4396",
    400: "#535ac8",
    500: "#6870fa",
    600: "#868dfb",
    700: "#a4a9fc",
    800: "#c3c6fd",
    900: "#e1e2fe",
  },
  }),
});

export const themeSettings = (mode) => {
  const colors = tokens(mode);

  return {
    palette: {
      mode: mode,
      ...(mode === "dark"
        ? {
            primary: {
              main: colors.primary[500],
            },
            secondary: {
              main: colors.orangeAccent[500],
            },
            neutral: {
              dark: colors.gray[700],
              main: colors.gray[500],
            },
          }
        : {}
      ),
    },
  };
};

```

```
    light: colors.gray[100],
  },
  background: {
    default: colors.primary[500],
  },
}
: {
  primary: {
    main: colors.primary[100],
  },
  secondary: {
    main: colors.orangeAccent[500],
  },
  neutral: {
    dark: colors.gray[700],
    main: colors.gray[500],
    light: colors.gray[100],
  },
  background: "#fcfcfc",
}),
},
typography: {
  fontFamily: ["Poppins", "sans-serif"].join(","),
  fontSize: 12,
  h1: {
    fontFamily: ["Poppins", "sans-serif"].join(","),
    fontSize: 40,
  },
  h2: {
    fontFamily: ["Poppins", "sans-serif"].join(","),
    fontSize: 32,
  },
  h3: {
    fontFamily: ["Poppins", "sans-serif"].join(","),
    fontSize: 24,
  },
  h4: {
    fontFamily: ["Poppins", "sans-serif"].join(","),
    fontSize: 20,
  },
  h5: {
    fontFamily: ["Poppins", "sans-serif"].join(","),
    fontSize: 16,
  },
  h6: {
    fontFamily: ["Poppins", "sans-serif"].join(","),
```

```
    fontSize: 14,  
  },  
},  
};  
};
```

```
export const ColorModeContext = createContext({  
  toggleColorMode: () => {},  
});
```

```
export const useMode = () => {  
  const [mode, setMode] = useState("light");
```

```
  const colorMode = useMemo(  
    () => ({  
      toggleColorMode: () =>  
        setMode((prev) => (prev === "light" ? "dark" : "light")),  
    }),  
    [],  
  );
```

```
  const theme = useMemo(() => createTheme(themeSettings(mode)), [mode]);
```

```
  return [theme, colorMode];  
};
```