ThemeProvider

Multi-Platform Component

```
<ThemeProvider>{/* themed UDS Base components*/}</ThemeProvider>
```

Introduction

```
import { ThemeProvider } from '@telus-uds/components-base'
```

[Follow the appropriate instructions](#) to add this component in to your app.

Guidance

> (!) INFO
>
> ThemeProvider is automatically included in BaseProvider. **You don't need
> to include ThemeProvider separately in your app if you are already
> using BaseProvider**.

ThemeProvider makes a UDS theme available in React context for UDS Base
components to consume.

Theme switching

> (!) INFO
>
> coming soon!

Spacing scale

ThemeProvider exposes the (12 stop) spacing scale corresponding to the UDS
theme passed via the `useSizingScale` hook

```
0 / 0px
1 / 4px
2 / 8px
3 / 16px
4 / 24px
5 / 32px
6 / 40px
7 / 48px
8 / 64px
9 / 72px
10 / 96px
11 / 120px

function SizingScale() {
  const Scale = ({ index }) => {
    const value = useSpacingScale(index)
    return (
      <div>
        {index} / {value}px
      </div>
    )
  }

  return (
    <div>
      {Array.apply(null, Array(12)).map((_, i) => (
        <Scale index={i} key={i} />
      ))}
    </div>
  )
}
```

Props

Name	Type	Platform	Default	Description
themeOptions	shape	standard	{}	An object containing options allowing to customize the theming

Name	Type	Platform	Default	Description
				experience: - `forceAbsoluteFontSizing` available on web only; when set to true, allows using absolute font sizing (in pixels, doesn't scale) instead of the relative sizing (in `rem`, scales depending on the browser settings) - `contentMaxWidth`: allows configuration of the content max width to be used in components such as Footnote and Notification to avoid content to stretch its width more then the page's width - `forceZIndex`: available on web only, when set to false, sets zIndex of `View` to be `auto` and when true, sets zIndex to `0` (the default from `react-native-web`) - `enableHelmetSSR`: on Web SSR, allows React Helmet to run during server-side rendering. This should be disabled unless web app has been specifically configured to stop React Helmet accumulating instances (which may cause a memory leak). See React Helmet's docs: https://github.com/nfl/react-helmet

Name	Type	Platform	Default	Description
children*	node	standard		
defaultTheme*	shape	standard		