StyleSheet

Work with strict styles that provide deterministic rendering and automatically adapt to localized writing direction.

The StyleSheet abstraction converts predefined styles to (vendor-prefixed) CSS without requiring a compile-time step. Styles that cannot be resolved outside of the render loop (e.g., dynamic positioning) are usually applied as inline styles.

```
import { StyleSheet } from 'react-native';
```

> **Did you know?** StyleSheet automatically merges styles and produces “utility” CSS for lightweight, reliable, and performant styling. Read more in the styling guide.

API

Static properties

> **absoluteFill**: ?number
> A very common pattern is to create overlays with position absolute and zero positioning, so `absoluteFill` can be used for convenience and to reduce duplication of these repeated styles.

> **absoluteFillObject**: ?Object
> Sometimes you may want `absoluteFill` but with a couple tweaks - `absoluteFillObject` can be used to create a customized entry in a StyleSheet.

> **hairlineWidth**: ?Object
> Equal to 1px. This is not implemented using screen density as browsers may round sub-pi es down to `0`, causing the line not to be rendered.

Static methods

> **create**: ({ [key]: ruleset }) => ({ [key]: ruleset })
> Define style objects. Each key of the object passed to `create` must define a style object. These values should not be introspected at runtime.

> **flatten**: (styles: Style) => Object
> Flatten an array of styles into a single style object. **This is not recommended as it is not compatible with static extraction of styles to CSS.**

Updated July 20, 2023 Edit