

스타일링 라이브러리 정리

1 지금까지 알게 된 것

정리

styled-components

CSS in JS

CSS의 문제점을 해소

해결책

스타일을 style 태그로 분리

사용법(Template literals)

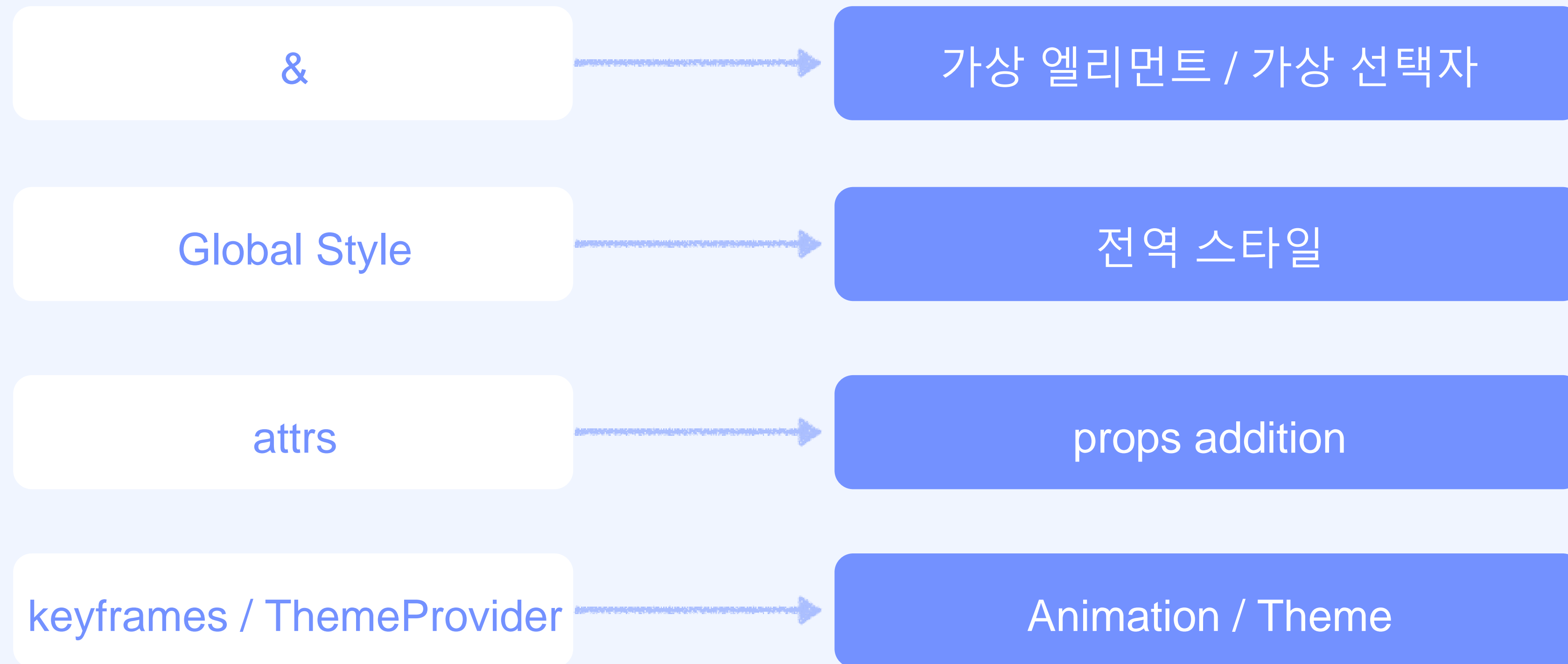
`styled.{element}``

styled(스타일드컴포넌트)

상속

정리

styled-components 2



정리

emotion

react에 특화

@emotion/react

css props

jsx를 대체함

styled components

styled-component + @

composition

css 안에서 css 사용

정리

emotion 2

기능

Fallbacks, &, Global, keyframes

styled-components

서로 점점 유사해짐

trend

emotion이 우세한 듯

사이즈 / 속도

emotions이 우세한 듯

정리

sass

전처리기

CSS의 확장

Sass / Scss

보다 CSS와 유사한 SCSS

사용

variables / modules / mixin / extend

정리

sass 2

syntax

언어처럼 자체 syntax가 있음

Interpolation

`#{ }` 값을 주입(마치 ``${ }``)

values / functions

프로그래밍 언어스러움

라이브러리를 고르는 기준

2 스타일링 라이브러리

npm trend

(<https://www.npmtrends.com/@emotion/css-vs-@emotion/react-vs-styled-components-vs-sass>)

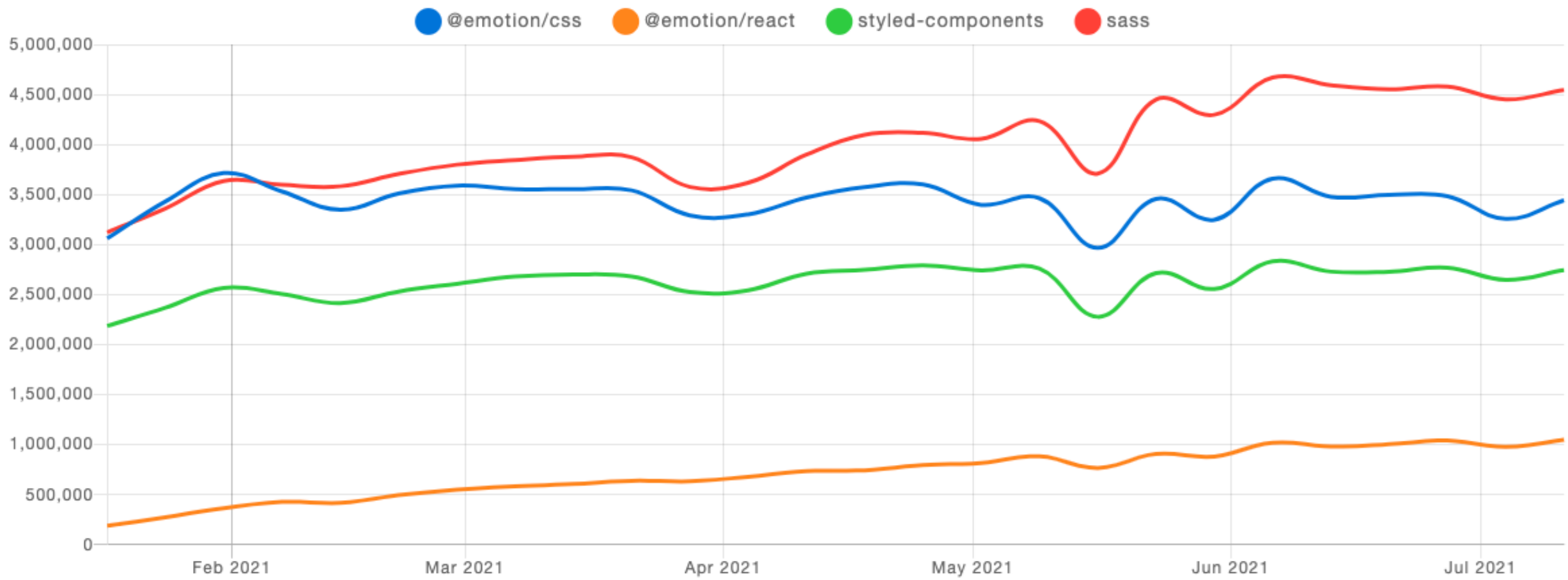
npm trends

@emotion/css vs @emotion/react vs sass vs styled-components

Enter an npm package...

@emotion/css @emotion/react styled-components sass + emotion + glamorous + aphrodite + jss + radium

Downloads in past 6 Months



Stats

	stars 🌟	issues 🐛	updated 🔧	created 🗓️	size 📦
@emotion/css					minzipped size 5.7 KB
@emotion/react					minzipped size 7.6 KB
styled-components	34,183	266	Jul 17, 2021	Aug 16, 2016	minzipped size 12.6 KB
sass	2,440	82	Jul 15, 2021	Nov 1, 2016	minzipped size 482.1 KB

CSS의 문제점 (React를 쓰면서...)

Global Namespace: 글로벌 변수를 지양해야하는 JS와 대치

Dependencies: css 간의 의존 관리

Dead Code Elimination: 안쓰는 css 인지 어려움

Minification: 클래스 이름 최소화

Sharing Constants: JS의 코드와 값을 공유하고 싶음

Non-deterministic Resolution: css 파일 로드 타이밍 이슈

Isolation: 격리

어떤 라이브러리를 선택할 지는 개발자(본인)에게 달려있음

경험

반응형 웹을 대응하기 위해 media query를 많이 쓰고

딱히 design system이 없던 곳에서 sass를 썼음

지금은 개발 편의성을 생각하면 CSS in JS를 쓸듯 emotion