

리액트 컴포넌트 스타일링

너무나 평범한
그냥 CSS

일단 고유한
클래스명 쓰자.
CSS Module

참 편리한
Sass, LESS,
Stylus ...

경계선을
허물자,
CSS in JS



너무나 평범한
그냥 CSS

기존에 CSS로만 작업하는게 익숙한 경우
디자이너가 CSS 작성하는 경우
다른 도구 고민하기 귀찮은 경우

```
.MyComponent { }  
.MyComponent .head { }  
.MyComponent .content { }  
.MyComponent .content .something-inside { }
```

일단 고유한
클래스명 쓰자.
CSS Module

컴포넌트마다 고유한 클래스명을 사용하고 싶다!

```
.MyComponent { }  
.head { }  
.content { }  
.something-inside { }
```

```
import React, { Component } from 'react';
import styles from './MyComponent.css';

class MyComponent extends Component {
  render() {
    return (
      <div className={styles.MyComponent}>
        <div className={styles.head}></div>
      </div>
    )
  }
}

export default MyComponent;
```

classnames 라이브러리는
굉장히 쓸만합니다!


```
import React from 'react';
import classNames from 'classnames/bind';
import styles from './Stylish.css';

const cx = classNames.bind(styles);

const Stylish = () => {
  return (
    <div className={cx('Stylish')}>
      <div className={cx('box')}></div>
    </div>
  );
};

export default Stylish;
```

개인적인 느낌:

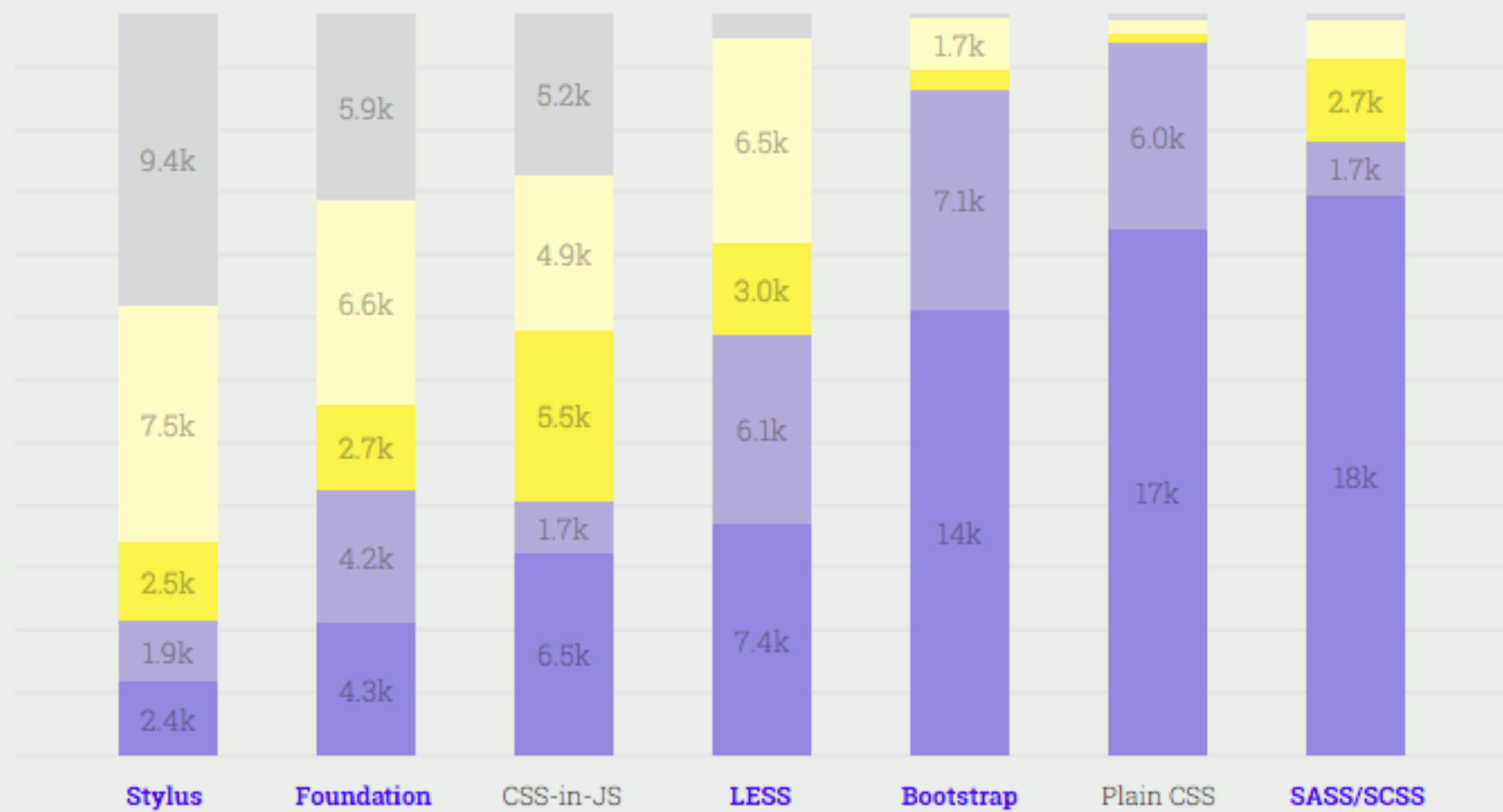
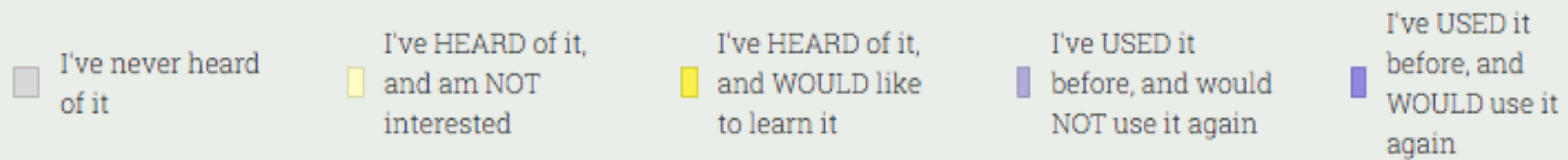
꽤 편리하긴 한데 계속 cx 쓰기 귀찮다.
그렇다고 styles._____ 는 더 귀찮고 ...

className="something" 의 형식이 가장 편한건 사실.



참 편리한
Sass, LESS,
Stylus ...


CSS를 더욱 쉽게 작성 할 수 있습니다.
함수 같은 것도 사용 할 수 있습니다.



```
$color: #d3d3d3;

.MyComponent {
  .head {
    @include media("<tablet") {
      width: 100%;
    }
    .content {
      background: $color;
      @include material-shadow(2, 0.5);
      .something-inside { }
    }
    .button {
      &:hover {}
    }
    .option {
      &.active { }
    }
  }
}
```

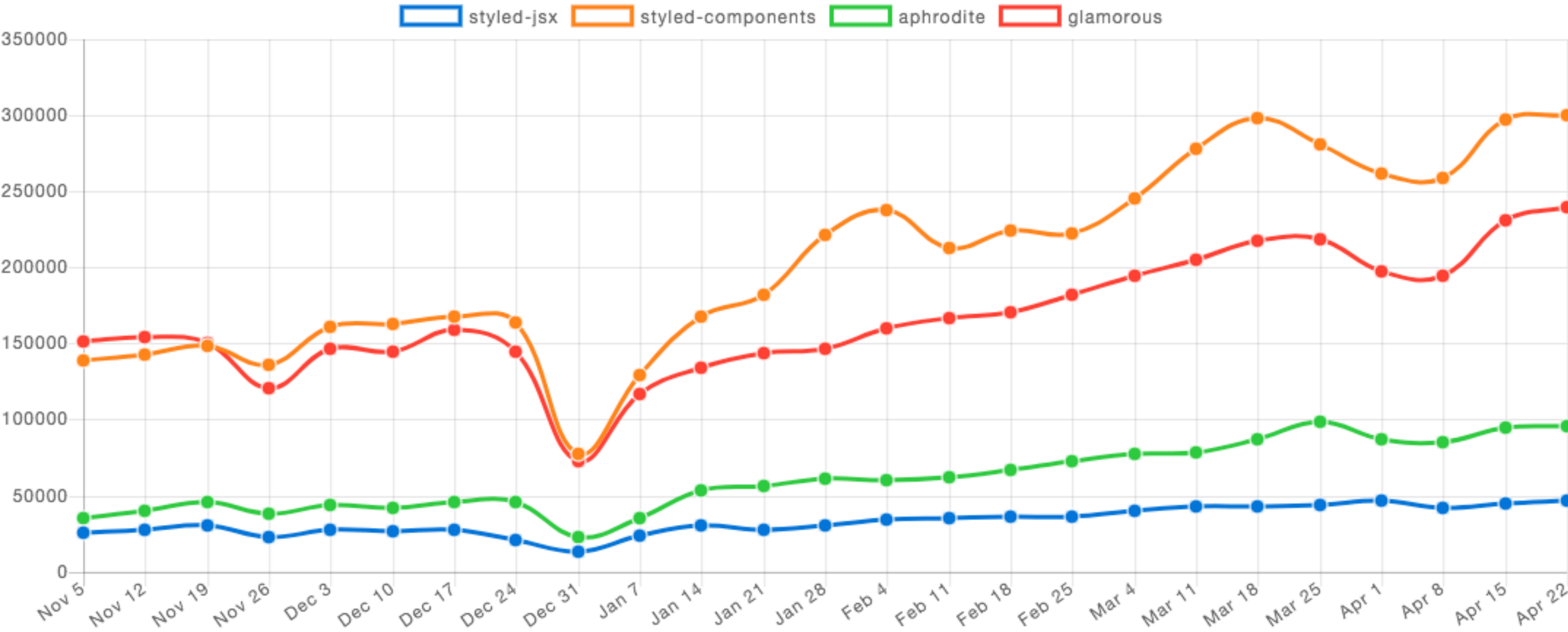
Sass + CSS Module 의 조합도 가능!

A green circle containing white text.

경계선을
허물자,
CSS in JS

그냥 자바스크립트 안에 CSS 적을래 ...
종류는 정말 많다! [링크]
파일 왔다 갔다 할 필요 없어서 좋음

Downloads in past 6 Months ▾



그래서 뭐 쓰지?

골고루 한번 짬은 사용해보자

사람마다 취향도 다르고
팀마다 원하는것이 다르다.