

# React: CSS in JS

Christopher “vjeux” Chedeau

# Problems with CSS at Scale

1. Global Namespace
2. Dependencies
3. Dead Code Elimination
4. Minification
5. Sharing Constants
6. Non-deterministic Resolution
7. Isolation

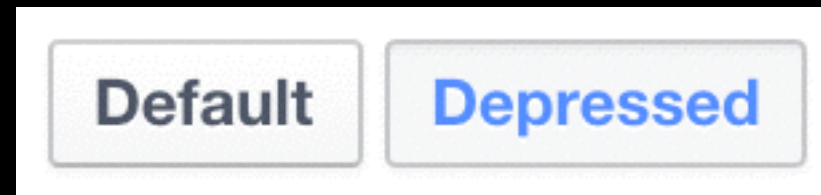
A cartoon illustration depicting a chaotic scene, likely a riot or protest. In the foreground, several people with exaggerated, angry facial expressions are shown. On the left, a man with a large red beard and a woman with a pink shirt and purple pants are visible. In the center, a man in a blue suit and tie is looking forward with a concerned expression. To his right, a man in a blue uniform with a police cap is also looking forward. On the far right, a man in a grey suit and a pink bow tie is holding a torch. The background is filled with a large, intense fire, and many other people are visible, some holding torches, suggesting a large-scale event. The overall style is a colorful, exaggerated cartoon.

# React: CSS in JS

Christopher “vjeux” Chedeau

[speakerdeck.com/vjeux/react-css-in-js](https://speakerdeck.com/vjeux/react-css-in-js)

# Let's build a button



```
/* button.css */

.button {
  background: #f6f7f8 url(/images/button/background.png) repeat-x;
  border: 1px solid #cdced0;
  border-radius: 2px;
  box-shadow: 0 1px 1px rgba(0, 0, 0, 0.05);
}

.button-depressed {
  background-color: #4e69a2;
  border-color: #c6c7ca;
  color: #5890ff;
}
```



# Let's build a button

```
/* button.js */  
  
var styles = {  
  container: {  
    background: '#f6f7f8 url(/images/button/background.png) repeat-x',  
    border: '1px solid #cdced0',  
    borderRadius: 2,  
    boxShadow: '0 1px 1px rgba(0, 0, 0, 0.05)',  
  },  
  depressed: {  
    backgroundColor: '#4e69a2',  
    borderColor: '#c6c7ca',  
    color: '#5890ff',  
  },  
};
```

# Differences

```
/* button.js */

var styles = {
  container: {
    background: '#f6f7f8 url(/images/button/background.png) repeat-x',
    border: '1px solid #cdced0',
    borderRadius: 2,
    boxShadow: '0 1px 1px rgba(0, 0, 0, 0.05)',
  },
  depressed: {
    backgroundColor: '#4e69a2',
    borderColor: '#c6c7ca',
    color: '#5890ff',
  },
};
```

# Inline Styles!!1!

```
/* button.js */
```

```
var styles = {  
  container: {  
    background: '#f6f7f8 url(/images/button/background.png) repeat-x',  
    border: '1px solid #cdced0',  
    borderRadius: 2,  
    boxShadow: '0 1px 1px rgba(0, 0, 0, 0.05)',  
  },  
  depressed: {  
    backgroundColor: '#4e69a2',  
    borderColor: '#c6c7ca',  
    color: '#5890ff',  
  },  
};
```

```
<div style={styles.container}>
```

# Conditionals

```
/* button.js */
```

```
propTypes: {  
  isDepressed: React.PropTypes.bool,  
},
```

```
<div style={m(  
  styles.container,  
  this.props.isDepressed && styles.depressed  
)} />
```



# Conditionals

```
function m(...args) {  
  return Object.assign({}, ...args);  
}
```

# Customization

```
/* button.js */

propTypes: {
  isDepressed: React.PropTypes.bool,
  style: React.PropTypes.object,
},

<div style={m(
  styles.container,
  this.props.isDepressed && styles.depressed,
  this.props.style
)} />
```

# Customization

```
/* button.js */

propTypes: {
  isDepressed: React.PropTypes.bool,
  style: React.PropTypes.object,
},

<div style={m(
  styles.container,
  this.props.style,
  this.props.isDepressed && styles.depressed
)} />
```

# Customization

```
/* button.js */
```

```
propTypes: {  
  isDepressed: React.PropTypes.bool,  
  color: React.PropTypes.string,  
},
```

```
<div style={m(  
  styles.container,  
  this.props.color && {color: this.props.color},  
  this.props.isDepressed && styles.depressed  
)} />
```

# Visibility

```
.hidden {  
  display: none;  
}
```



# Visibility

```
.hidden {  
  display: none;  
}
```

```
<div>  
  {visible && <div />}  
</div>
```

# ::after

```
.element::after {  
  content: '';  
}
```

# ::after

```
.element::after {  
  content: '';  
}
```

```
<div>  
  { /* ... */ }  
  <div />  
</div>
```

# :hover

```
.element {  
  border-color: transparent;  
}
```

```
.element:hover {  
  border-color: red;  
}
```

# :hover

```
.element {  
  border-color: transparent;  
}
```

```
.element:hover {  
  border-color: red;  
}
```

```
<div  
  onMouseEnter={() => this.setState({ hover: true })}  
  onMouseLeave={() => this.setState({ hover: false })}  
  style={{ borderColor: this.state.hover ? 'red' : 'transparent' }}  
</div>
```



# Media Queries

```
.element {  
  width: 600px;  
}
```

```
@media (max-device-width: 1024px) {  
  .element {  
    width: 300px;  
  }  
}
```

# Media Queries

```
.element {  
  width: 600px;  
}
```

```
@media (max-device-width: 1024px) {  
  .element {  
    width: 300px;  
  }  
}
```

```
<div style={{ width: window.innerWidth > 1024 ? 600 : 300 }} />
```

# Media Queries

```
var ResizeMixin = {  
  componentWillMount: function() {  
    this.resizeListener = window.addEventListener(  
      'resize',  
      this.forceUpdate.bind(this)  
    );  
  },  
  componentWillUnmount: function() {  
    window.removeEventListener(this.resizeListener);  
  },  
};
```

# Conclusion

Christopher “vjeux” Chedeau