

Lab 19 + DSA 04

Sorting, Sass, React Testing, and Deployment

seattle-javascript-401n14

Vocab Review



What is a component?



How are components defined?



What is state?



What is props?



What is context?

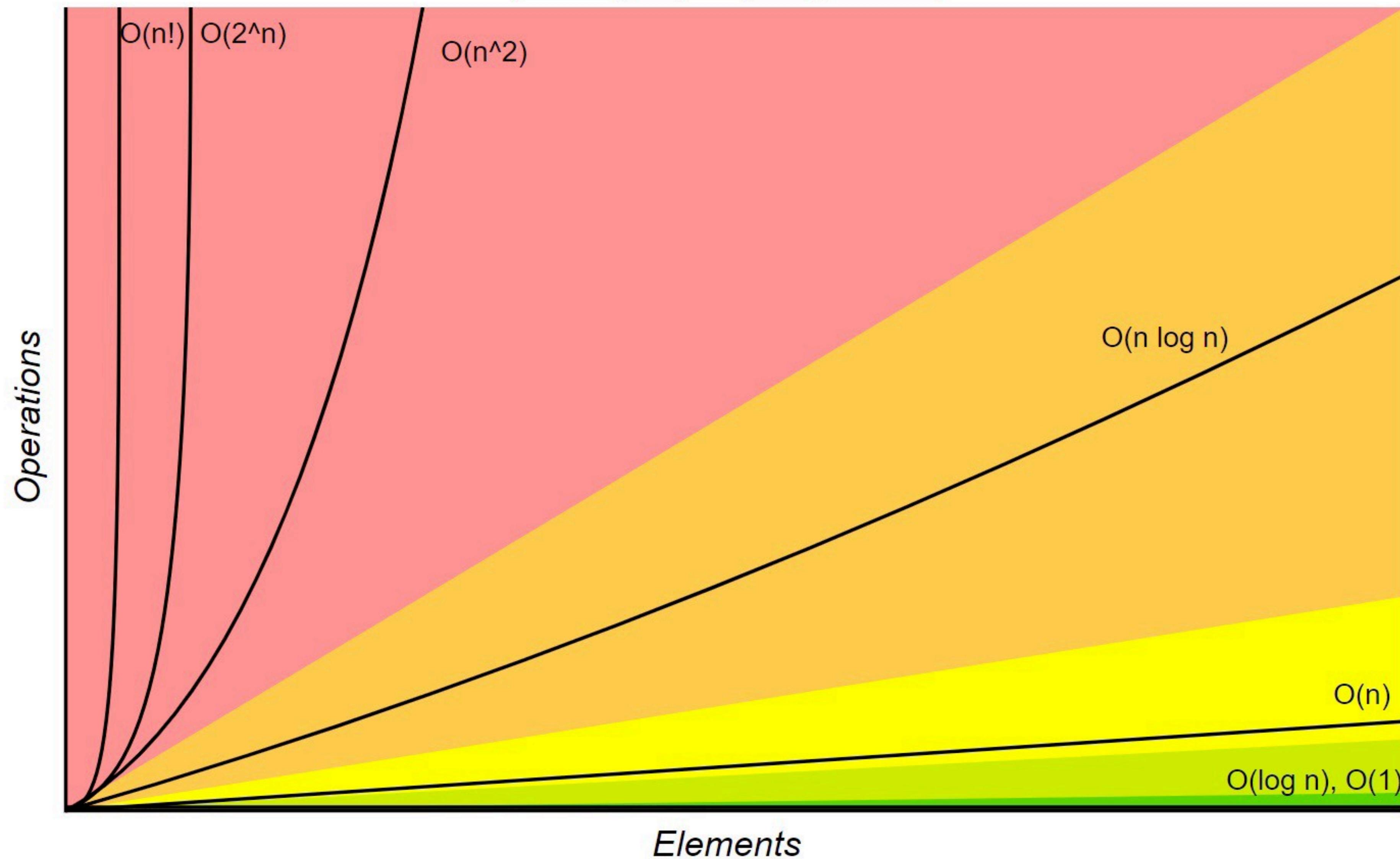


What is big-o?



Big-O Complexity Chart

Horrible Bad Fair Good Excellent



Sorting

- A very common coding need
- Think of every application that has filter and sort tables
- What happens when the size of the list gets larger and larger?
- Various algorithms that have different time complexities
- We'll cover the simplest one this class

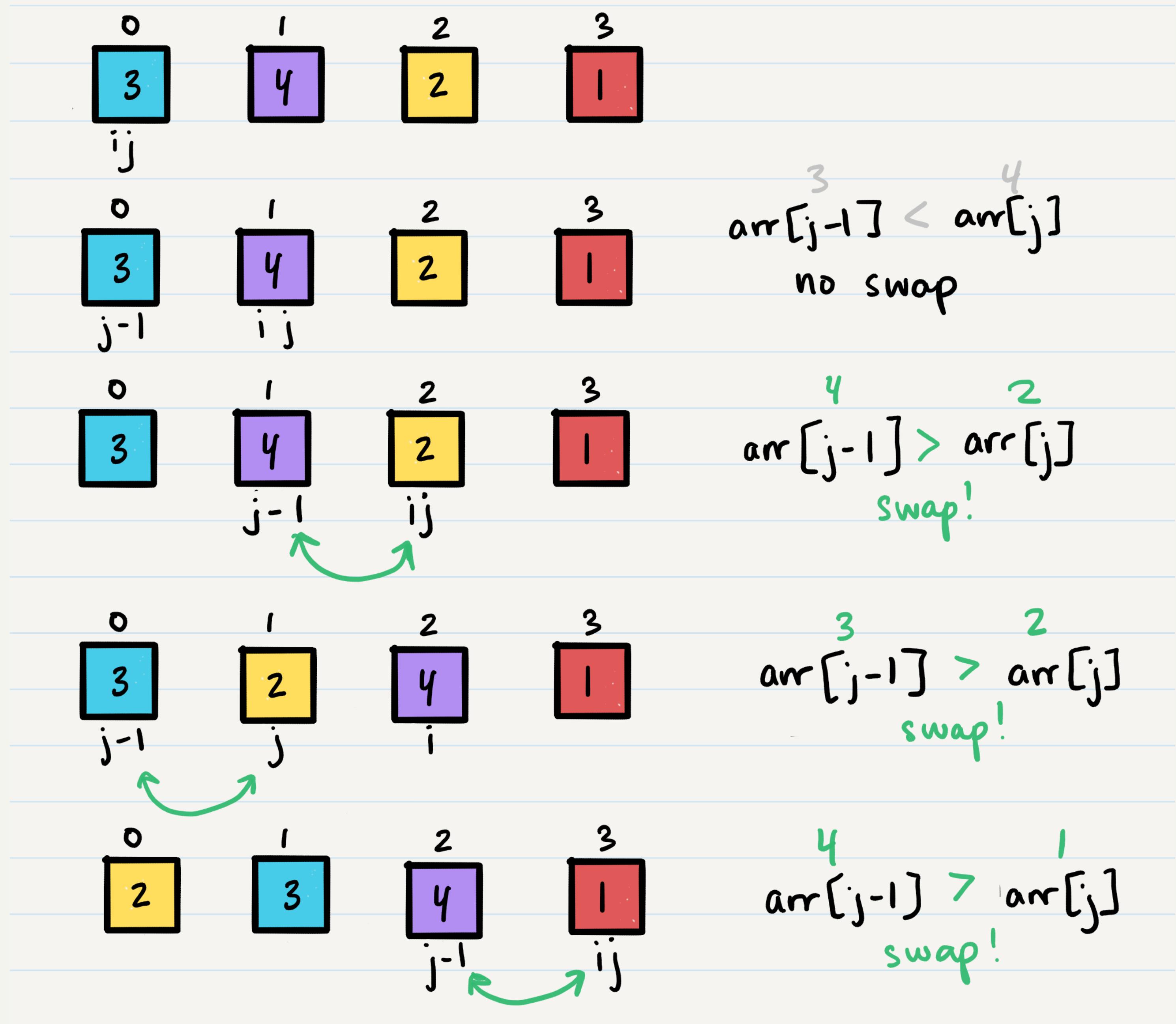


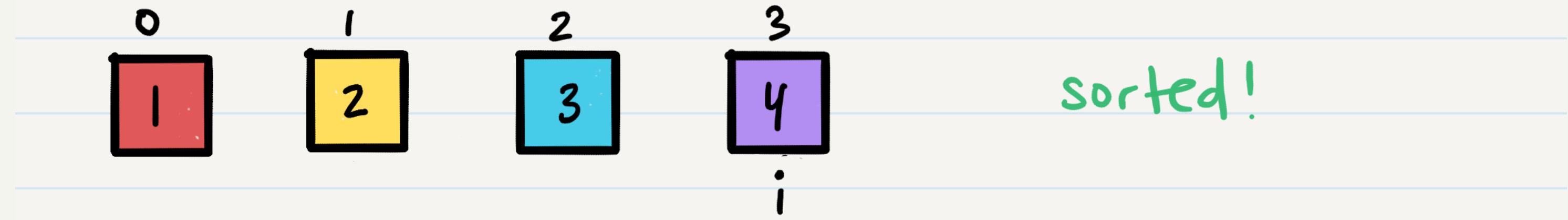
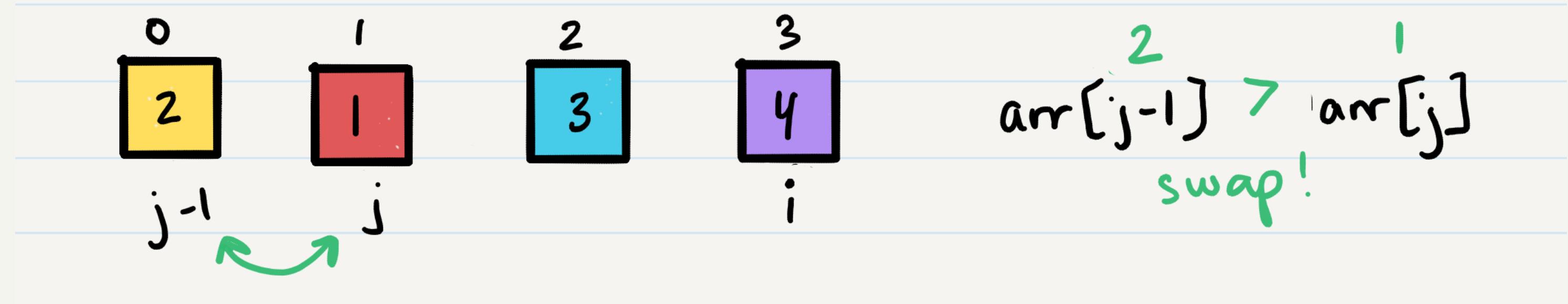
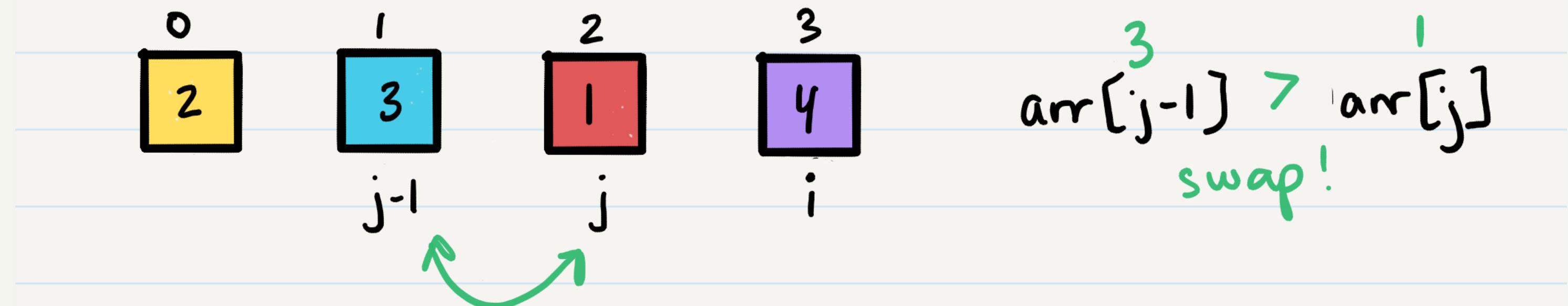
Insertion Sort

- Imagine sorting playing cards in your hands
 - Work left to right, index 0 to n-1
 - For each index, move the value left until it doesn't need to go further

```
while i < arr.length  
    j = i;  
  
    while j > 0 & arr[j - 1] > arr[j]  
        swap arr[j - 1] and arr[j]  
        j--  
  
    i++
```







Insertion Sort

- Two while loops, so $O(n^2)$
- Not very efficient!
- Avoid if you can, though it's the simplest sort to implement



We'll resume
sorting next class!

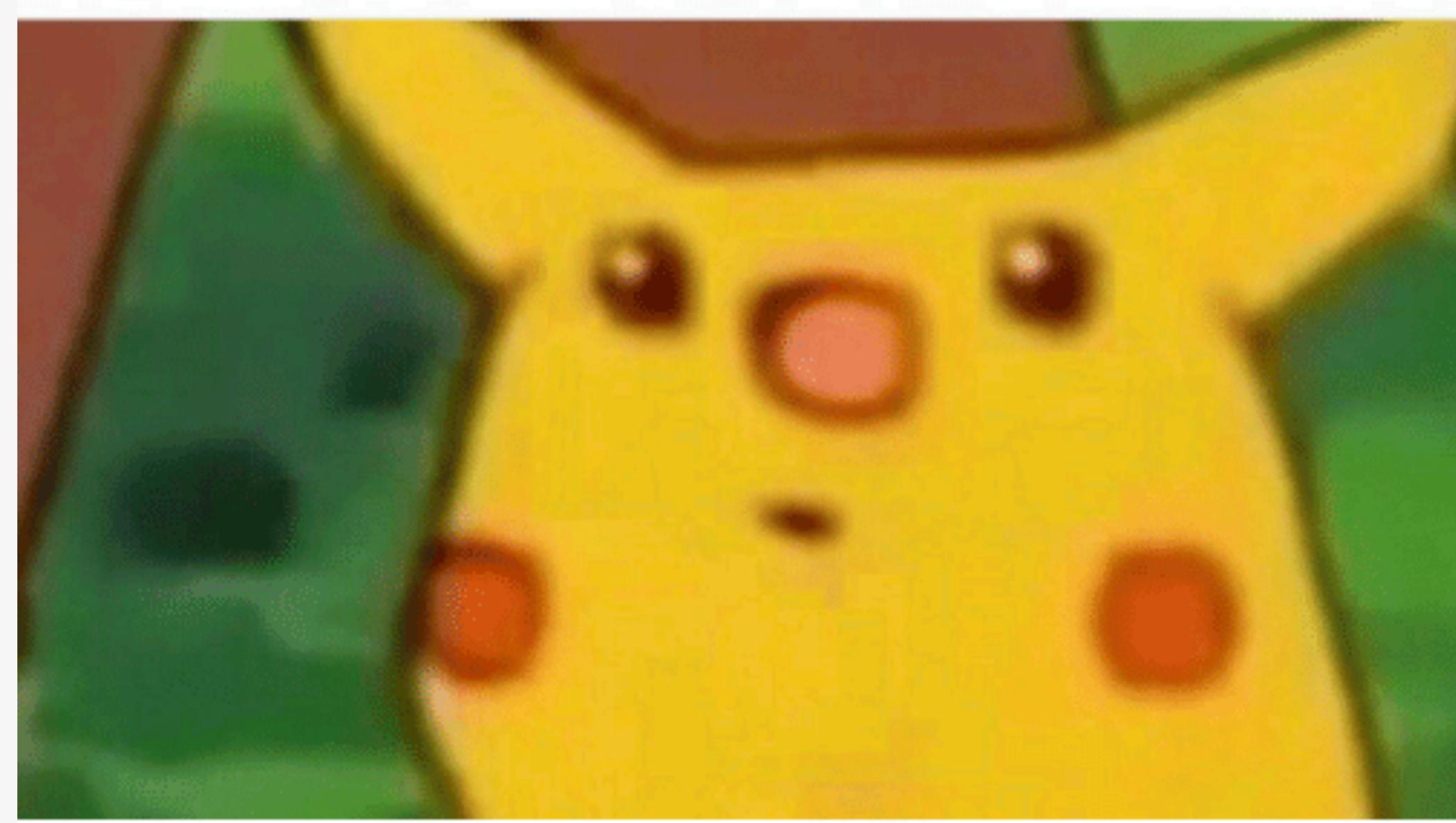


CSS Isn't Good

- Very restrictive
- Multiple sections of duplicated code
- Hard to change things
- Hard to separate pieces out / modularize
- No way to add code logic
- So...let's use a language that *compiles* into CSS!

Me: *makes small change to css*

My site:



CSS Preprocessors

- Languages with more code-like features
- **Variables!** Change colors, spacing, fonts, etc across your whole UI effortlessly
- **Mixins!** Generate snippets of style / CSS which you can include anywhere and as much as you like
- **Functions!** Add, subtract, calculate style values
- **Partials!** Modularize your styles
- **Nesting!** Better organization of styles based on nested selectors



Demo

demo/sass-test-deploy

Let's start styling something with Sass and check out all its cool features!

```
1 @import './variables';
2 @import './mixins';
3
4 h1,
5 h2 {
6   color: $theme-color;
7 }
8
9 .colored-box {
10  background-color: $theme-color;
11 }
12
13 .bordered-box {
14  border: 2px solid $theme-color;
15  @include border-radius(6px);
16  @include format-box();
17
18  h2 {
19    font-style: italic;
20    span {
21      color: $highlight-color;
22    }
23  }
24
25  button {
26    color: $theme-color;
27    margin: $spacing;
28    width: 100px;
29  }
30 }
```



Testing Your UIs

- Three major approaches
 - **Snapshot testing** - does the component match a previously saved “snapshot” of what it used to be?
 - Really handy when you’re refactoring, not useful if you’re still iterating on the UI
 - **Mount testing** - render the component and allow me to fake user action, while also being able to check state variables
 - **Render testing** - render the component and let me see what it looks like and search for HTML in the right places



Enzyme

- Another package we use in addition to Jest
- Allows us to search HTML using selectors (like JQuery!)
- We can fake user action such as button click or form input
- We can access component state

I WILL FIND YOU. I WILL TEST YOU.



AND I WILL REPORT YOUR
LATENCY

memegenerator.net

Demo

demo/sass-test-deploy

Time to add tests to our application!

```
1 import React from 'react';
2 import renderer from 'react-test-renderer';
3 import { configure, mount } from 'enzyme';
4 import Adapter from 'enzyme-adapter-react-16';
5
6 import App from '../components/App';
7 configure({ adapter: new Adapter() });
8
9 describe('App component', () => {
10   it('renders correctly', () => {
11     const page = renderer.create(<App />).toJSON();
12     expect(page).toMatchSnapshot();
13   });
14
15   it('has a Counter', () => {
16     let component = mount(<App />);
17     expect(component.find('div.bordered-box h2').text()).toBe(
18       'Your current count is: 0',
19     );
20   });
21
22   it('adds one to Counter', () => {
23     let component = mount(<App />);
24     let addBtn = component.find('div.bordered-box button#add');
25
26     addBtn.simulate('click');
27     expect(component.find('div.bordered-box h2').text()).toBe(
28       'Your current count is: 1',
29     );

```



What's Next:

- Due by Midnight Tonight:
 - **Learning Journal 19**
- Due by Midnight Sunday:
 - **Feedback Week 12**
 - **Career Coaching Assignments**
- Due by Midnight Monday:
 - **Code Challenge 19**
- Due by 6:30 PM Tuesday:
 - **Reading Class 20**
 - **Lab 19**
- Next Class on Tuesday: **Class 20 - Props and State + DSA 04**





DON'T LEAVE YET

THERE'S STILL .04

SECONDS LEFT IN

CLASS

A photograph of a young woman with dark hair, smiling warmly at the camera. She is wearing a light-colored, short-sleeved top. Behind her is a large world map showing various continents in different colors. To the left of the map, a green chalkboard is visible with some faint, partially legible text, including "ASIA" and "ME". The overall composition suggests a classroom setting.

Questions?