How to style with inline styles

Throughout the sample files, we've provided lots of tags with "style={styles.foo}". These are all present to make it easier on you to populate styles.

Styling and CSS is a very time-consuming task. Given that we have limited time this week, it is much better to spend it on learning React rather than fine-tuning styles.

Therefore we've provided a bunch of styles in the lab steps below. We'll ask you to create your own styles on a couple. But after that we'll simply give you the styles to type in. It should go much quicker and we can focus on React.

App.js

1. Our company name, "DINNER AND A MOVIE" appears in two places in App.js. Both of them are in links (<a href="/">). Locate those two links.
2. We want them both to be styled in the same way. So add a style attribute to both. Set it to any object name you like. "styles.navlink" would be a fine candidate but you get to choose.
3. Add a const object in App.js with the name you chose above. It should have properties that will make these links appear …
   - to be in all caps (hint: in CSS, this is text-transform),
   - not underlined (hint: in CSS, it would be text-decoration),
   - and have a padding of 10px.
4. Run and test. Adjust your styles until they meet the above requirements.

LandingPage.js

5. The poster is way too big. It already has a style of styles.poster. So add an object called "poster" to styles and give it a maxWidth of 95% and an objectFit of 'contain'.
6. Run and test. When your poster is smaller and fits in its container you got it right.

7. Now for the rest of LandingPage, do something like the below. As you create each new style, feel free to refresh so you can see what the styles are doing if you like.

```
header: {
  width: "95vw",
  margin: "10px auto",
  padding: "10px",
},
filmsWrapper: {
  display: 'flex',
  flexWrap: 'wrap',
},
wrapper: {
  width: '300px',
  margin: '30px',
  cursor: 'pointer',
},
innerWrapper: {
```

```
    display: 'flex',
    flexDirection: 'row',
  },
  poster: {
    maxWidth: '95%',
    objectFit: 'contain',
  },
  posterDiv: {
    flex: '1 1 25%',
    padding: "5px",
  },
  textDiv: {
    flex: '1 1 25%',
    padding: "10px 5px 10px 5px",
  },
  title: {
    fontSize: '1.75rem',
    fontWeight: '900',
    textTransform: 'uppercase',
    margin: '2px',
  },
  tagline: {
    margin: '2px',
    lineHeight: '1em',
  },
  runtimeP: {
    fontSize: "0.75rem",
    margin: '2px',
  },
```

FilmDetails

8. Here are some styles for FilmDetails. Add them in to the styles. Once again, make an effort to understand what they do, but don't get stuck analyzing; our focus needs to be on React and not on CSS properties.

```
container: {
  width: '95%',
  margin: '20px 20px',
  padding: '20px',
},
poster: {
  maxWidth: '95%',
  objectFit: 'contain',
},
wrapper: {
  marginTop: '20px',
},
headline: {
  fontSize: '1.2em',
```

```
  },
  showingTimesWrapper: {
    display: 'flex',
    flexWrap: 'wrap',
  },
  title: {
    background: 'black',
    color: 'white',
    fontWeight: 'bold',
    padding: '20px',
    margin: '10px',
  },
```

PickSeats.js

9. You've already got a couple of styles in PickSeats in order to format the seat image. Add these as additional styles:

```
header: {
  width: "95vw",
  margin: "10px auto",
  padding: "10px",
},
tablesSection: {
  paddingBottom: '40px',
  display: 'grid',
  gridTemplateColumns: '1fr 1fr 1fr 1fr 1fr',
},
wrapper: {
  margin: "20px",
},
tableWrapper: {
  display: 'flex',
  justifyContent: "center",
  alignItems: "center",
},
tableItself: {
  height: "40px",
  backgroundColor: "blue",
  borderRadius: "20px",
  color: "white",
  fontSize: "20px",
  display: 'flex',
  justifyContent: "center",
  alignItems: "center",
},
seatsWrapper: {
  display: "flex",
  justifyContent: "center",
  alignItems: "center",
```

```
},
```

Bonus!

10. If you have extra time in this and future labs, feel free to add styles to make your site look as good as you can.