



WEEK-TWO

JSX, JS Expressions in JSX & ES6 Template Literals,
Styling React Element



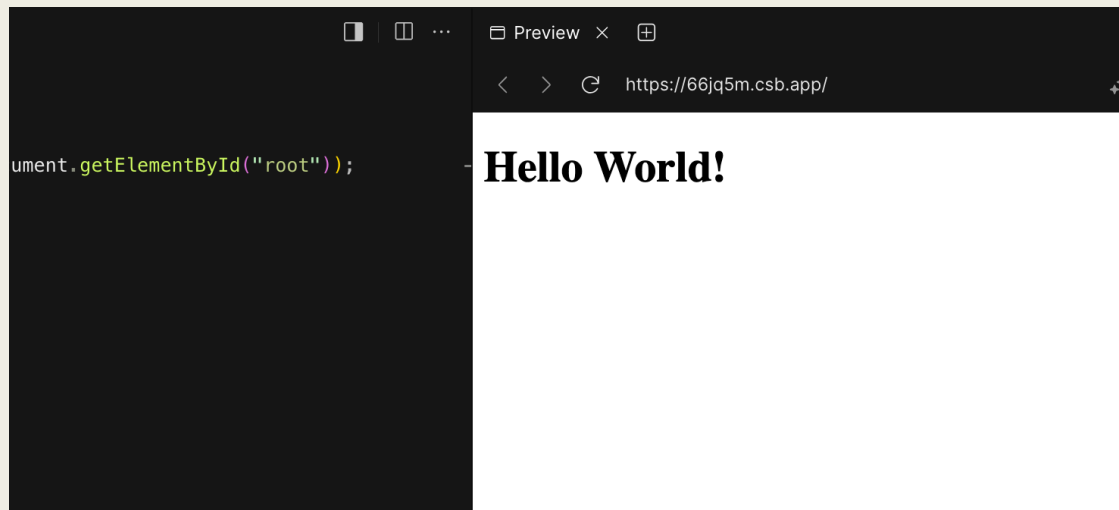
Codesandbox

1. Find boiler plate code-sandbox from - <https://codesandbox.io/s/introduction-to-jsx-h8ub1?fontsize=14&module=/public/index.html+babeljs.io&file=/public/index.html>
2. In order to save your work, you need to fork and login with github.
3. Change the title of the app to JSX and view that in the browser.

The screenshot shows a code editor with a dark theme. On the left, a file explorer shows a project structure with a 'public' folder containing 'index.html' and 'styles.css', and a 'src' folder containing 'index.js' and 'package.json'. The 'index.js' file is open in the editor, showing the following code:

```
1 let React = require("react");
2 let ReactDOM = require("react-dom");
3
4 ReactDOM.render(<h1>Hello World!</h1>, document.getElementById("root"));
5
```

Below the file explorer, a 'Dependencies' section shows the installed packages: 'react' at version 18.2.0 and 'react-dom' at version 16.11.0.



Steps: Hello world! In react

4. Add react and react dom dependencies.

5. Require react and react dom dependencies.

6. Use render method to write HTML

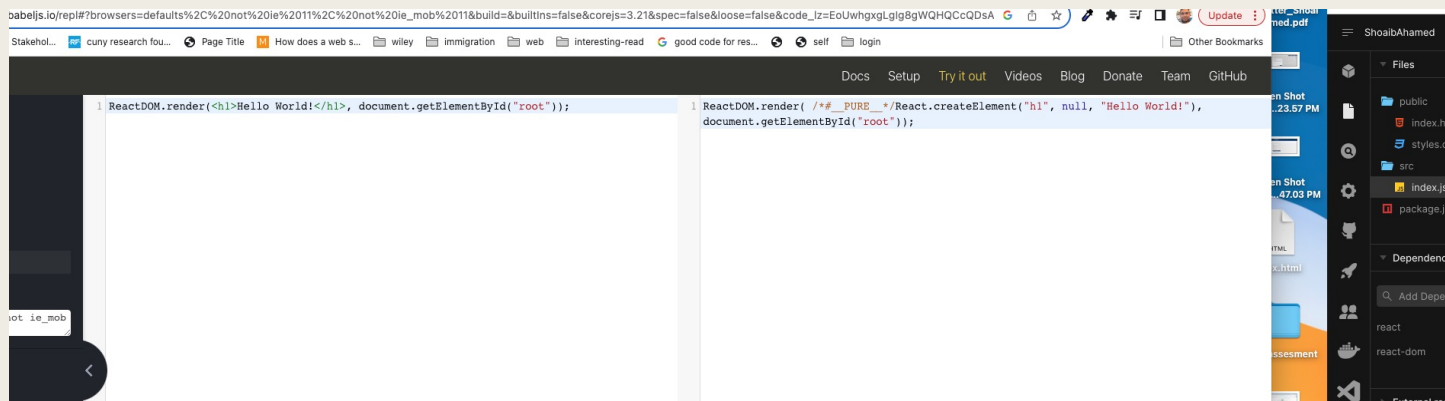
7. View in the browser.

8. Files that have HTML right in the body of JS file is called JSX.

9. The compiler that picks up HTML and compiles to JS is called Babel.

Babel

- [Babel](#)



Challenge : Add a `<p>` element.

Solution :

```
<> index.html  JS index.js  x
1  let React = require("react");
2  let ReactDOM = require("react-dom");
3
4  ReactDOM.render(
5    <div>
6      <h1>Hello World!</h1>
7      <p>Hello Universe</p>
8    </div>,
9    document.getElementById("root")
10 );
11 |
```

Challenge :

- Go to this link - <https://codesandbox.io/s/introduction-to-jsx-zp681?fontsize=14&file=/src/index.js:0-160>

Then *1.Create a react app from scratch.*

2.It should display a h1 heading.

3.It should display an unordered list (bullet points).

4.It should contain 3 list elements.

Solution

The screenshot shows a code editor with a dark theme. The top bar indicates the user is 'ShoibAhamed' and the project is 'Drafts / JSX-Code-Challenge (forked)'. The left sidebar shows the file structure with 'public', 'src', 'index.js', and 'package.json'. The 'Dependencies' section lists 'react' and 'react-dom' at version 16.11.0. The main editor displays the code for 'index.js' and 'index.html'. The code in 'index.js' uses 'require' to load 'react' and 'react-dom', then uses 'ReactDOM.render' to render a list of grocery items into a root element. The code in 'index.html' is a simple HTML document with a root element. The right sidebar shows a browser window displaying the rendered application, which is a 'Grocery List' with three items: Rice, Egg, and Potato. The bottom status bar shows 'Console', 'Problems', and 'React DevTools'.

```
1 const React = require("react");
2 const ReactDOM = require("react-dom");
3
4 ReactDOM.render(
5   <div>
6     <h1>Grocery List</h1>
7     <ul>
8       <li>Rice</li>
9       <li>Egg</li>
10      <li>Potato</li>
11    </ul>
12  </div>,
13  document.getElementById("root")
14 );
15
16 //Create a react app from scratch.
17 //It should display a h1 heading.
18 //It should display an unordered list (bulle
19 //It should contain 3 list elements.
```

Grocery List

- Rice
- Egg
- Potato

JS Expressions in JSX & ES6 Template Literals

https://developer.mozilla.org/enUS/docs/Web/JavaScript/Reference/Template_literals

■ `const number = 7;`

```
import React from "react";
import ReactDOM from "react-dom";

const number = 7;

ReactDOM.render(
  <div>
    <h1>Hello World!</h1>
    <p> My lucky number is {number}</p>
  </div>,
  document.getElementById("root")
);
```

Hello World!

My lucky number is 7.

Challenge

- Declare two variables for your first and last name
- Print them in the <div>.

Solution

```
index.js x
1 import React from "react";
2 import ReactDOM from "react-dom";
3
4 const number = 7;
5 const fName = "Shoaib";
6 const lName = "Ahamed";
7
8 ReactDOM.render(
9   <div>
10     <h1>Hello World!</h1>
11     <p> My lucky number is {fName + " " + lName}</p>
12   </div>,
13
14   document.getElementById("root")
15 );
16
```

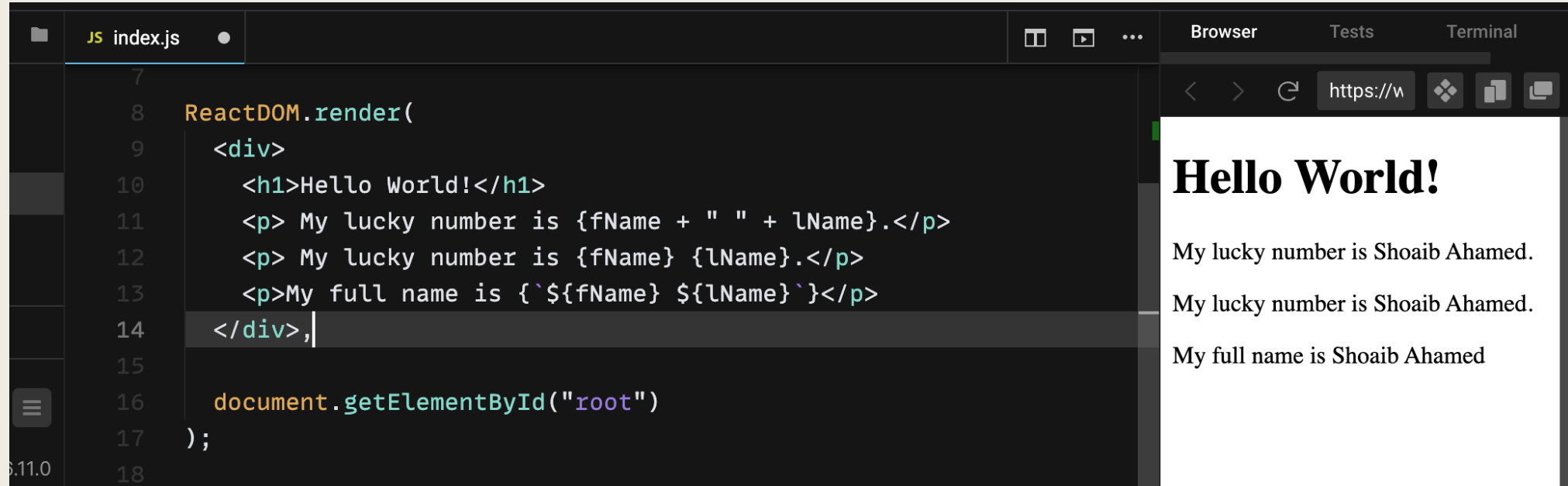
Browser Tests Terminal

https://w

Hello World!

My lucky number is Shoaib Ahamed.

Solution : Another way of doing it



```
7
8 ReactDOM.render(
9   <div>
10     <h1>Hello World!</h1>
11     <p> My lucky number is {fName + " " + lName}</p>
12     <p> My lucky number is {fName} {lName}</p>
13     <p>My full name is `${fName} ${lName}`</p>
14   </div>,
15
16   document.getElementById("root")
17 );
18
```

Browser Tests Terminal

https://w

Hello World!

My lucky number is Shoaib Ahamed.

My lucky number is Shoaib Ahamed.

My full name is Shoaib Ahamed

Assignment : Go [here](https://codesandbox.io/p/sandbox/javascript-expressions-in-jsx-practice-1fde0)

<https://codesandbox.io/p/sandbox/javascript-expressions-in-jsx-practice-1fde0>

- *//Create a react app from scratch.*
- *//It should display 2 paragraph HTML elements.*
- *//The paragraphs should say:*
- *//Created by YOURNAME.*
- *//Copyright CURRENTYEAR.*
- *//E.g.*
- *//Created by Shoaib Ahamed .*
- *//Copyright 2019.*

Styling React Element

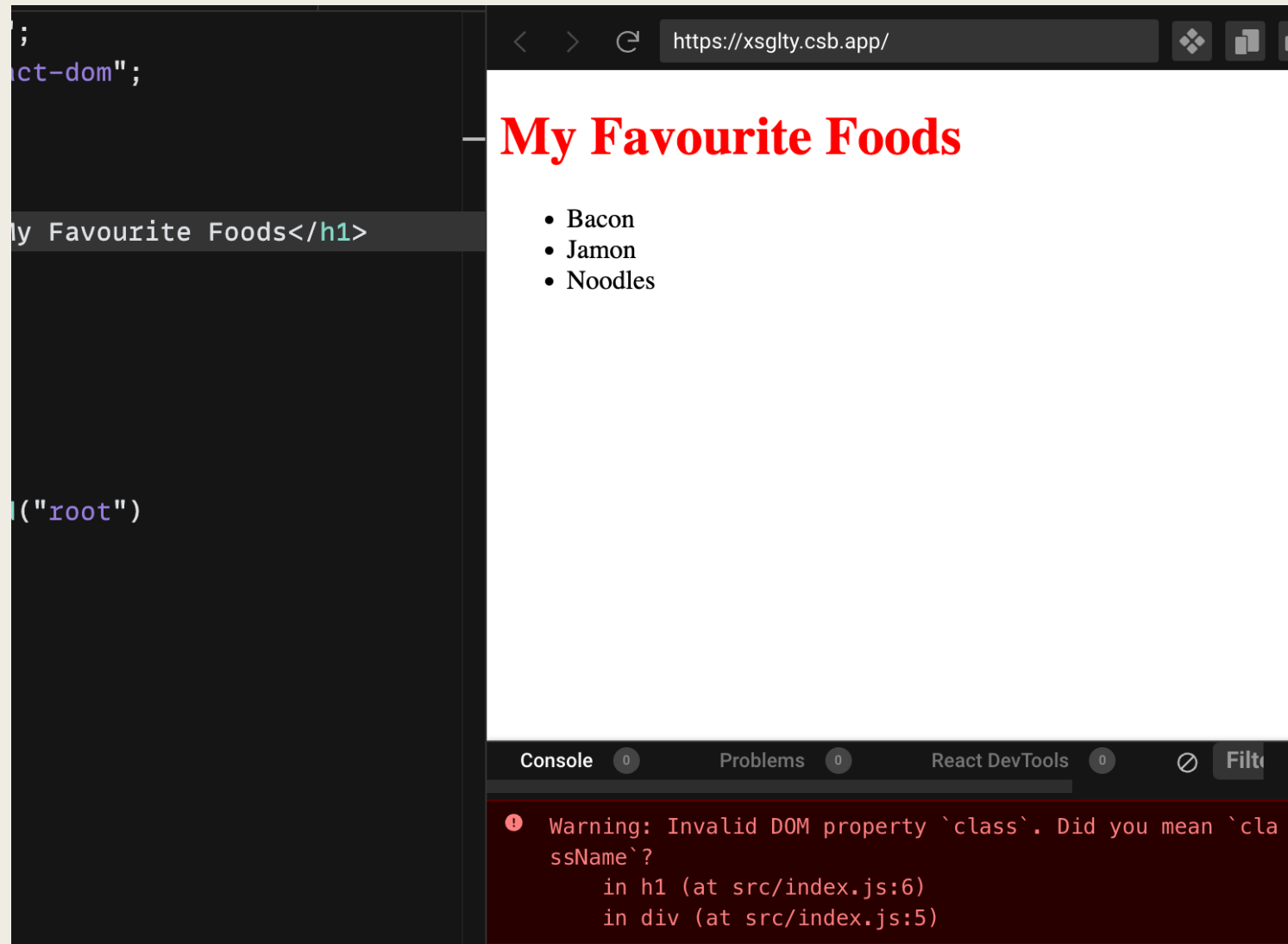
■ We will discuss –

1. How to style your react element
2. How to add attributes.

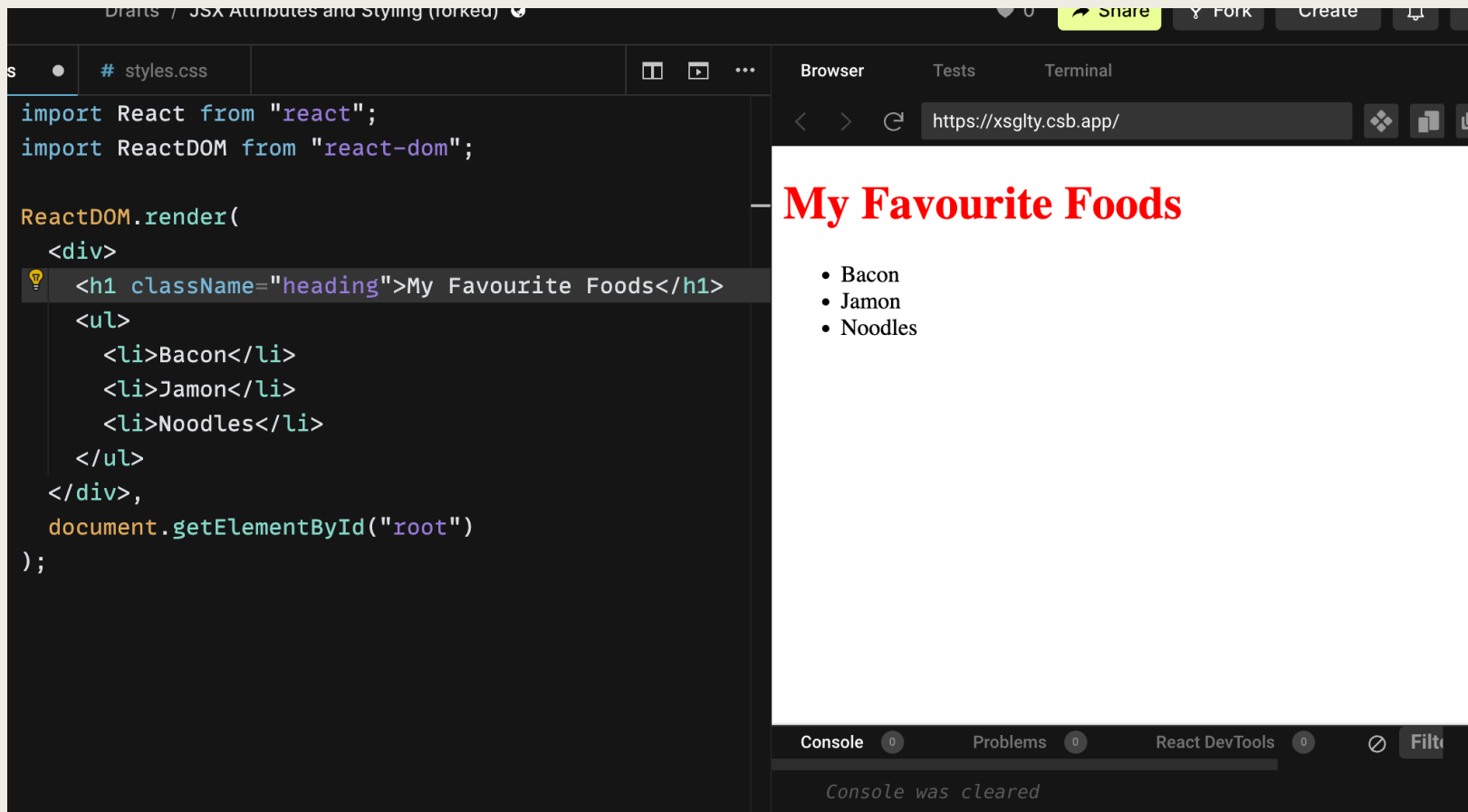
Go to this sandbox - <https://codesandbox.io/s/javascript-expressions-in-jsx-practice-completed-ekgvl?fontsize=14>

Then fork it to make your own copy.

- In order to change the color of `<h1>` element, you can add a class and target the class in `styles.css` file and give it a color.



- If we change class to className, the error will go away. Attribute name in react needs to be camelCase.



The screenshot shows a code editor with a dark theme. The left pane displays the following code:

```
import React from "react";
import ReactDOM from "react-dom";

ReactDOM.render(
  <div>
    <h1 className="heading">My Favourite Foods</h1>
    <ul>
      <li>Bacon</li>
      <li>Jamon</li>
      <li>Noodles</li>
    </ul>
  </div>,
  document.getElementById("root")
);
```

The right pane shows a browser preview at <https://xsglty.csb.app/>. The rendered output is:

My Favourite Foods

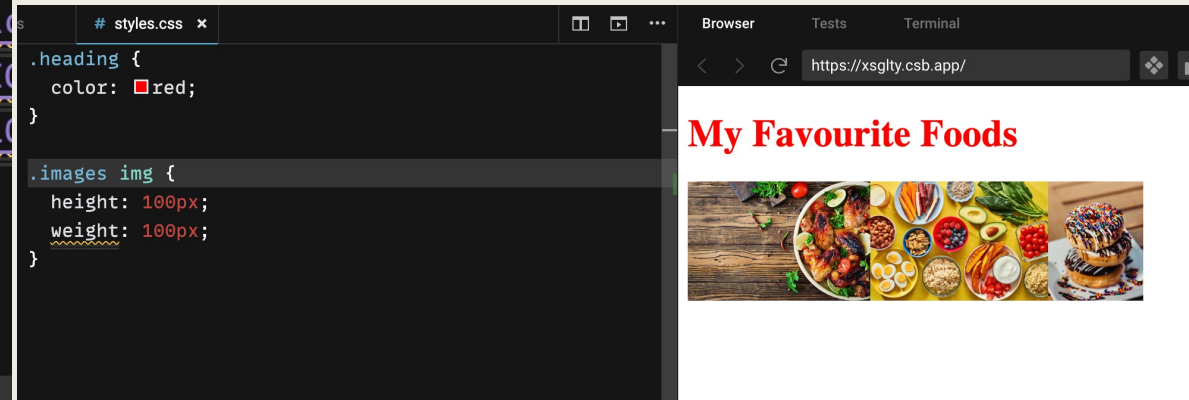
- Bacon
- Jamon
- Noodles

The bottom of the editor shows a console with the message "Console was cleared".

Class challenge:

- Replace the UI element with div element and have three image elements inside of that div element.
- Style the image sizes 100px by 100px.

```
ReactDOM.render(
  <div>
    <h1 className="heading">My Favourite Foods</h1>
    <div className="images">
      
      
      
    </div>
  </div>,
  document.getElementById("root")
);
```



SOLUTION:

Learning

- All the html attributes work in JSX. [List of HTML elements](#)
- The multiword html attributes has to be camel case in order to use in JSX.
- Html tag becomes or in react.
- We can also put an image as a value of js variable and use it like this –
- `const img = "https://img.freepik.com/free-photo/fresh-pasta-with-hearty-bolognese-parmesan-cheese-generated-by-ai_188544-9469.jpg"`

``

Inline styling in JSX

- Go to this link and fork this - <https://codesandbox.io/s/jsx-attributes-and-styling-completed-i1u11?fontsize=14>
- For inline css in react, create JS object with CSS properties like this

```
import ReactDOM from "react-dom";
```

```
const customStyle = {  
  color: "red",  
  fontSize: "20px",  
  border: "2px solid black"
```

```
};|
```

```
ReactDOM.render(  
  <div style={customStyle}>
```

Then apply the style object in inline css

```
import React from "react";
import ReactDOM from "react-dom";

const customStyle = {
  color: "red",
  fontSize: "20px",
  border: "2px solid black"
};

ReactDOM.render(
  <h1 style={customStyle}>Hello World!</h1>,
  document.getElementById("root")
);
```



- Have a look in all CSS property list - <https://www.w3schools.com/cssref/index.php>
- Play around with the properties and try use them in JSX.
- Remember the property names needs to be camelCase e.g font-size will be fontSize.

Assignment :

- Go here - <https://codesandbox.io/s/inline-styling-in-jsx-completed-zpc72?fontsize=14&file=/src/index.js:0-367>
- *//Create a React app from scratch.*
- *//Show a single h1 that says "Good morning" if between midnight and 12PM.*
- *//or "Good Afternoon" if between 12PM and 6PM.*
- *//or "Good evening" if between 6PM and midnight.*
- *//Apply the "heading" style in the styles.css*
- *//Dynamically change the color of the h1 using inline css styles.*
- *//Morning = red, Afternoon = green, Night = blue.*

