

App.js and App.css for Drum Machine

App.js

```
import React from 'react';
import './App.css';

/*audioSourceArr: [["Heater-1.mp3", "Heater 1", "Q"], ["Heater-2.mp3", "Heater 2", "W"], ["Heater-3.mp3", "Heater 3", "E"], ["Heater-4_1.mp3", "Heater 4", "A"], ["Heater-6.mp3", "Heater 6", "S"], ["Dsc_Oh.mp3", "Clap", "D"], ["Kick_n_Hat.mp3", "Kick n' Hat", "Z"], ["RP4_KICK_1.mp3", "Kick", "X"], ["Cev_H2.mp3", "Closed HH", "C"]]*//

class Controller extends React.Component {
  constructor (props) {
    super (props);
    this.state = {
      audioSourceArr: [["Heater-1.mp3", "Heater 1", "Q"], ["Heater-2.mp3", "Heater 2", "W"], ["Heater-3.mp3", "Heater 3", "E"], ["Heater-4_1.mp3", "Heater 4", "A"], ["Heater-6.mp3", "Heater 6", "S"], ["Dsc_Oh.mp3", "Clap", "D"], ["Kick_n_Hat.mp3", "Kick n' Hat", "Z"], ["RP4_KICK_1.mp3", "Kick", "X"], ["Cev_H2.mp3", "Closed HH", "C"]],
      play: "",
      buttonBack: "#808080",
      volume: 10,
      statusBoxText: ""
    }
    this.soundPlay = this.soundPlay.bind(this);
    this.volumeChange = this.volumeChange.bind(this);
  }

  soundPlay (id, status) {
    let mySound = new Audio ('https://cdn.freecodecamp.org/testable-projects-fcc/audio/' + id)
    mySound.volume = (this.state.volume / 100);

    //console.log("soundPlay runs")
    //console.log(id)
    //console.log("https://cdn.freecodecamp.org/testable-projects-fcc/audio/" + id)
    mySound.play();

    const statusOut = () => {
      this.setState ({
        statusBoxText: status
      })
      setTimeout( () => {
        this.setState({
          statusBoxText: ""
        })
      }, 700)};

    if (this.state.statusBoxText === "") {
      statusOut();
    }
  }
}
```

```
};
```

```
}
```

```
volumeChange (e) {  
  this.setState ({  
    volume: e.target.value  
  })  
  console.log(this.state.volume);  
  console.log("the above is this.state.volume");  
  const statusOut = () => {  
    this.setState ({  
      statusBoxText: "Volume: " + this.state.volume  
    })  
  }  
}
```

```
setTimeout( () => {  
  this.setState({  
    statusBoxText: ""  
  });  
}, 700));  
if (this.state.statusBoxText === "") {  
  statusOut();  
};
```

```
}
```

```
render () {  
  const controllerArr = (this.state.audioSourceArr.map ( index => <button onClick = { () =>  
    this.soundPlay(index[0], index[1])} className = "button">{index[2]}</button>))
```

```
  return (  
    <div >  
      <div id="background-container" style={{background: "#808080", width: "100%", height: "100vh",  
        display: "flex", justifyContent: "center", alignItems: "center"}}>  
        <div id="player-container" style={{background: "#B3B3B3", border: "6px solid #FFA500", width:  
          "380px", height: "550px", display: "flex", flexDirection: "column"}}>  
          <div style = {{background: "#B3B3B3", width: "380px", height: "300px", display: "grid",  
            gridTemplateColumns: "1fr 1fr 1fr"}}>  
            {controllerArr}  
          </div>  
          <div id="status-container" style={{display: "flex", flexDirection: "column", margin: "auto"}}>  
  
            <div id="statusBox" style = {{background: "#808080", height: "100px", width: "200px", textAlign:  
              "center", border: "3px solid #000", borderRadius:  
                "5px"}}><h1>{this.state.statusBoxText}</h1></div>
```

```

<div class="slidecontainer" style={{display: "flex", flexDirection: "row", justifyContent: "center" }}>
  <input type="range" min="1" max="100" value={this.state.volume} class="slider" id="myRange"
    onChange = {this.volumeChange} style = {{accentColor: "#FFA500", marginTop: "30px"}}></input>
</div>
</div>
</div>
</div>
);
}
}
}

```

```

export default Controller;
//ReactDOM.render(<App />, document.getElementById("root"));
//ReactDOM.render(<TypesOfVehicles />, document.getElementById("node-id"));

```

App.css

```

.App {
  text-align: center;
}

.App-logo {
  height: 40vmin;
  pointer-events: none;
}

@media (prefers-reduced-motion: no-preference) {
  .App-logo {
    animation: App-logo-spin infinite 20s linear;
  }
}

.App-header {
  background-color: #282c34;
  min-height: 100vh;
  display: flex;
  flex-direction: column;
  align-items: center;
  justify-content: center;
  font-size: calc(10px + 2vmin);
  color: white;
}

```

```
}
```

```
.App-link {  
  color: #61dafb;  
}
```

```
.button {  
  width: 100px;  
  height: 80px;  
  background: #808080;  
  color: #000;  
  margin: 10px;  
  border: 2px solid #000000;  
  border-radius: 4px;  
  box-shadow: 10px 10px 5px 0px rgba(0, 0, 0, 0.5);  
  font-size: 20px;  
  font-weight: bold;  
}
```

```
.button:active {  
  background-color: #FFA500;  
}
```

```
#button-container {  
  display: grid;  
  grid-template-columns: 1fr 1fr 1fr;  
}
```