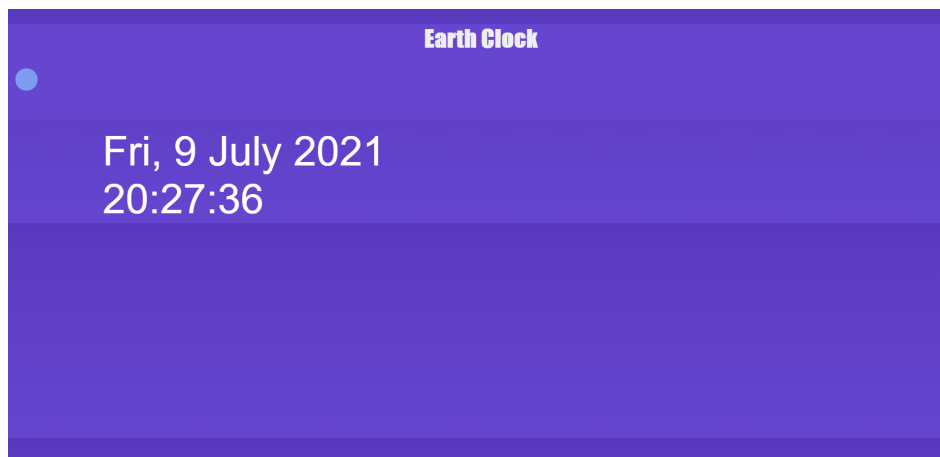
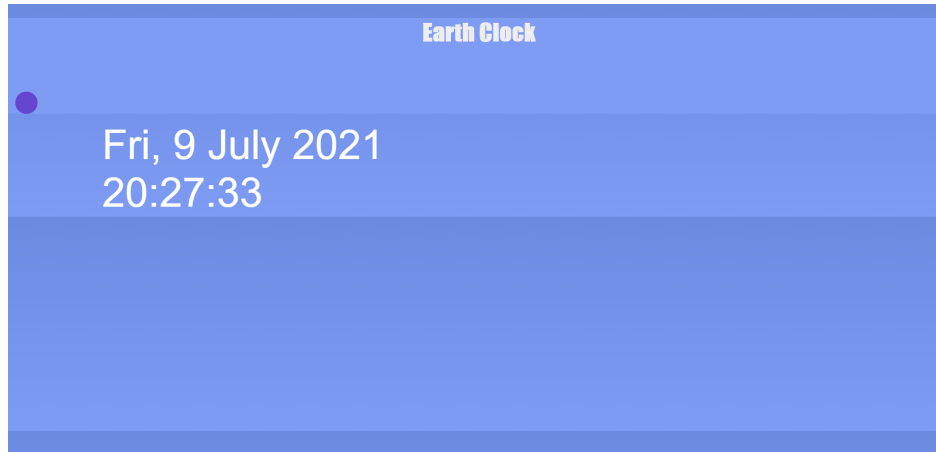


Challenge 2 Description

For this challenge, I learned how to create let functions and how to utilize these functions to create a working clock.

The challenge is to create an animated clock with a theme change. Here is how the website looks like with Day mode and Night mode. The button is located on top of the clock itself (the colored circles).



For the theme change, I had to link the original javascript to 2 other CSS data. Both CSS data use gradient functions to create the gradient effect on the background.

```
1 body {  
2   background-image: linear-gradient(#608AE1, #7f9cf5);  
3   color: #000000;  
4 }  
5  
6 header {  
7   background-color: #7f9cf5;  
8   color: #eee;  
9 }  
10
```

```
1 body {  
2   background-image: linear-gradient(#593ABF, #6646CF);  
3   color: #3c366b;  
4 }  
5  
6 header {  
7   background-color: #6646CF;  
8   color: #eee;  
9 }  
10  
11
```

```

38 let switches = document.getElementsByClassName('switch');
39
40 let style = localStorage.getItem('style');
41
42 if (style == null) {
43   setTheme('day');
44 } else {
45   setTheme(style);
46 }
47
48 for (let i of switches) {
49   i.addEventListener('click', function () {
50     let theme = this.dataset.theme;
51     setTheme(theme);
52   });
53 }
54
55 function setTheme(theme) {
56   if (theme == 'day') {
57     document.getElementById('switcher-id').href = 'day.css';
58   } else if (theme == 'night') {
59     document.getElementById('switcher-id').href = 'night.css';
60   }
61   localStorage.setItem('style', theme);
62 }

```

This is the rest of the javascript. I used an if function to activate the switch between day mode and night mode. I also wrote if (style == null) { setTheme('day') } to make sure that the website shows the day mode first when the page loads in.

Here is a breakdown on how I made the animated clock

```

1
2 function currentTime() {
3   let date= new Date();
4   let hour= date.getHours();
5   let minute= date.getMinutes();
6   let second= date.getSeconds();
7   hour = updateTime(hour);
8   minute = updateTime(minute);
9   second = updateTime(second);

```

First, in javascript, I made a new function that has let functions inside. With updateTime, it will update the time according to the device's clock.

```

10 let months = ['January', 'February', 'March', 'April', 'May', 'June', 'July', 'August', 'September', 'October', 'November', 'December'];
11 let days = ['Sun', 'Mon', 'Tue', 'Wed', 'Thu', 'Fri', 'Sat'];
12 let currentDate= date.getDate();
13 let currentDay= days[date.getDay()];
14 let currentMonth= months[date.getMonth()];
15 let currentYear= date.getFullYear();
16 let theDate = currentDay+" "+currentDate+" "+currentMonth+" "+currentYear;
17 document.getElementById("date").innerHTML = theDate;
18
19 document.getElementById ("clock").innerText=hour+" "+minute+" "+second;
20 let time= setTimeout(function(){
21   currentTime()
22 }, 1000
23 );
24 }
25

```

For the clock to show the date and month, I made let months filled with the name of the months and days. To make it show up, I use document.getElementById("date").innerHTML.

For the clock, the function is set to 1000 milliseconds (1 second).

```
26 // Adds 0 to single digit
27 function updateTime(c) {
28
29     if(c < 10) {
30         return "0" + c;
31     }
32     else{
33         return c;
34     }
35 }
36
37 currentTime();
```

Line 36, Column 1

I realized when I executed the previous code, the clock shows single-digit hours with one number (for example: 7:00:00 instead of 07:00:00). To solve this, I use the if statement for updateTime. If the time is less than 10, it will add 0 in front of the number.