

Name: Subodh Singh

Project: Alarm clock

Note: The code is written with Blue pen
& the comments with black pen

1) fetching all the html elements.

1) const hour = document.querySelector('hour');

2) const minute //

3) const amPm =

etc //

3) const ringtone = new Audio('file/path.
extension');

2) making the current time 00:00:00 am/pm
to the current time on the system.

Get using date = new Date(); and changing
the time time interval of 1sec.

SetInterval(C) =>

let date = new Date();

let h = date.getHour(); } → Methods of

let m = date.getMinute(); } Date class.

let s = date.getSeconds(); } → methods of

let ampm = 'AM'; } → methods of

If (h > 12) { }

h = h - 12; }

ampm = 'PM'

Suppose the
time is 14:00

$$\therefore h = 14 - 12 \\ h = 2$$

~~h = h == 0 ? h = 12 : h~~ → If the value is 0 then set it to 12

Background

~~h = h == 0 ? h = 12~~

$h = h < 10 ? "0" + h : h;$

$m = m < 10 ? "0" + m : m;$

$s = s < 10 ? "0" + s : s;$

current Time • Tex + Content = $"\$ (h) : \$ (m) : \$ (s)"$ \$ am pm

// time will be update according to the system

blocking when to play the alarm but before adding when to play we need to understand the localStorage class.

// local Storage : the local storage allows us to save key / value pairs in the browser (w3schools). It stores data with no expiration. Even if you refresh the page or close the browser tab, the data is deleted when the browser window is closed. It is just like an object with key values and pair but the only difference is it stores in the browser. we are going to localStorage.setItem(key, value) and get the value of a particular key we will use localStorage.getItem(key).

If we set this local storage afterwards let before lets get them.

```
If (localStorage.getItem("alarmtime") ==  
'3:30pm' || localStorage.getItem(  
("wanttoplay") == "yes")) {  
    playAlarm();  
}
```

```
3,  
}, 1000);
```

3) const setAlarm (for Setting Alarm)

const setAlarm = () => {

// Set alarm to the (All the localstorage Set
// whg)

localStorage.setItem ('isAlarmSet', 'true');

localStorage.setItem ('alarmTime', 'time');

localStorage.setItem ('contentClass', 'content
flex disable')

Content.className = localStorage.getItem
('contentClass');

// Disabling time Selection

// Setting button text to clear the Alar-
m.

localStorage.setItem ('btnText', 'Clear Alarm')

SetAlarmBtn.textContent = localStorage.
getItem ('btnText');



// Setting Alarm Time indicator.

alarmTimeIndicator.textContent = "Alarm
Time set to: " + localStorage.getItem
("alarmTime");

alarmTimeIndicator.className = "xx";

// Set userExited to avoid dom exception.

localStorage.setItem("userExited", "xxx");

// getting Alarm time from user.

let time = \$("hour.value"), \$("minute.value")
; \$("ampm.value");

If (time.includes("Hours")) || time.includes
("Minute") || time.includes("AM/PM")
alert ("Please enter a valid time");
return;

}

// clearing alarm.

if (localStorage.getItem('isAlarmSet') == 'true') {

localStorage.setItem('alarmTime', '00:00:00 AM');
ringtone.pause();

// handle selection of time

localStorage.setItem('contentClass', 'contentFlex');

Content.className = localStorage.getItem('ContentClass');

localStorage.setItem('btnText', 'Set Alarm')

Set.Alarm.textContent = localStorage.getItem('btnText');

resumeBtn.hidden = true.

alarmTimeIndicator.textContent = "Alarm Time is Set to :";

alarmTimeIndicator.className = "d-no";

localStorage.setItem("WantToPlay", "no");

return localStorage.setItem("isAlarmSet", false);

|| What was happening alone here is the example.

① Inside the function, the code checks if the alarm is currently set by examining a value stored in the web browser.

If (localStorage.getItem("isAlarmSet") == "true")

localStorage.getItem('isAlarmSet') ==
(true) retrieves the value
associated with the key 'isAlarmSet'
from the browser local storage.

If the value is true it means
the alarm is set.

2. If the alarm is set the following
action will be performed.

→ Resetting the time to 00:00:00AM

localStorage.setItem('AlarmTime', '00:00:
00')

→ Pause ringtone

ringtone.pause();

→ Making the selection of a new
alarm time by updating a CSS class

localStorage.setItem('contentClass', 'contentflex');

Class

content. Class Name = localStorage.get('Content Class');

→ Changing the text button

localStorage.setItem('btnText', 'Set Alarm');
SetAlarmBtn.textContent = localStorage.getItem('btnText');

→ Hiding alarm button

Resumbtn.isHidden = true

→ Resetting alarm time indicator

alarmTimeIndicator.textContent = 'Alarm
Time is set to:

alarmTimeIndicator.className = 'd-none';

→ Setting the value ('WantToPlay',
'no');
Finally changing the value in local
storage.

localStorage.setItem('WantToPlay',
'false');

function playSound() {

if ((localStorage.getItem('userExited') ==
'xxx') || (localStorage.getItem('wantToPlay') == 'yes')) {
ringtone.play();
eugene.loop();
}

```
const hideWelcomeScreen = () => {
    welcomeBackScreen.className = "d-none";
}

welcome
```

```
alarmTimeIndicator.textContent = "Alarm Time  
Set to: " + localStorage.getItem("alarm  
Time");
```

```
localStorage.setItem("userExited", "xxx");
```

```
localStorage.setItem("wantToPlay", "ye  
s");
```

3

// All the Import function are done
now.

It's easy to understand but
Inside the if statement that checks the
conclusion using logical OR operator
The first condition checks whether
the value associated with key 'userExist'
ed in the web local language is equal
to XXX. However there is mistake in this condition
and in second if the value pair for
'WantToPlay' is equal to 'Yes'
Play the Sound.

In Summary the 'playSound' function
play sound presumably if either
of this following statement is true
and to play the sound in loop
if the function is called

// Setting up hours

```
for (i = 12; i > 0, i--) {
```

 o = i < 10 ? "0" + i : ij

```
    let option = '' + ij + ''
```

 hour.firstElementChild.insertAdjacentHTML('

 after', option);

3

// Same logic goes for minute and
am_pm.

```
if (localStorage.getItem('userExited')) {
```

```
    localStorage.setItem('userExited', 'false');
```

3 else {

```
    if (localStorage.getItem('userExited') ==
```

```
        ('true') && localStorage.getItem('isAlarmed') ==
```

```
        ('true')) {
```

Welcome Back Screen - classmate

= WelcomeBack (or 3)

3

11 ~~Off~~ Local Storage . get Item ('unauthenticated')
this line checks whether there is an
in the local storage object ~~object~~ the
user existed . Inside the block the
Value is false

— 16 —