

# תזמון פונקציות

setTimeout

setInterval

# setTimeout

---

כאשר רוצים להריץ את קוד ה javascript אחרי זמן מסויים (milliseconds)

```
setTimeout("console.log('Hello World!')", 2000);
```

הרישום ללוג יתבצע לאחר שתי שניות (seconds 2 = milliseconds 2000)

# setTimeout

---

```
<body>
  <h1 id="counter">0</h1>
  <script>
    function increaseCountAfter5Sec() {
      let s = document.querySelector("#counter");
      s.innerHTML = parseInt(s.innerHTML) + 1;
    }

    setTimeout(increaseCountAfter5Sec, 1000)
  </script>
</body>
```

## הפונקציה מקבלת :

פונקציה לביצוע

פרמטר נוסף שמציין עוד כמה זמן  
יש להפעיל את הפונקציה: 1000 = 1 sec

# תזמון פונקציות

---

בדומה לאירוע דינאמי, ניתן לרשום את הפונקציה לביצוע כך:

```
setTimeout(function () {  
    let s = document.querySelector("#counter");  
  
    s.innerHTML = parseInt(s.innerHTML) + 5;  
  
}, 1000)
```

# clearTimeout

פונקציית setTimeout מחזירה מספר ייחודי, באמצעות מספר זה ניתן למנוע את ביצוע הפונקציה ע"י פונקציית clearTimeout.

```
<body>
  <span id="counter">0</span>
  <script>
    let increaseCount = setTimeout(
      function() {
        console.log("start");
        let s = document.querySelector("#counter");

        console.log(parseInt(s.innerHTML));

        s.innerHTML = parseInt(s.innerHTML) + 1;
        console.log(parseInt(s.innerHTML));

      }, 1000);

    clearTimeout(increaseCount);
  </script>
</body>
```

הפונקציה מקבלת את המספר הייחודי  
שיצרה setTimeout

```
//Fruit Arrays 1 & 2 declaration & assignment
const fruitArr1 = ['apple', 'mango'];
const fruitArr2 = ['banana', 'papaya'];

//If the fruit name is mango, the clearTimeout method will clear the timeout value and mango fruit will
not be displayed

function checkFruit(fruitArr) {
    for (var i in fruitArr) {
        const timeout = setTimeout(displayFruit, 2000, fruitArr[i]);
        if (fruitArr[i] === 'mango') {
            clearTimeout(timeout);
        }
    }
}

//Function to display the selected fruit name
function displayFruit(fruit) {
    console.log(`The fruit selected is ${fruit}`);
}

//Invoke checkFruit method
checkFruit(fruitArr1);
checkFruit(fruitArr2);
```

# modal

---

כאשר רוצים להציג הודעה ניתן להשתמש ב modal למשל <dialog>

```
<dialog id="myDialog">This is a modern alert!</dialog>
```

```
<script>
```

```
  const dialog = document.getElementById('myDialog');
```

```
  dialog.showModal();
```

```
  setTimeout(() => dialog.close(), 2000); // Auto-close after 2 seconds
```

```
</script>
```

# setInterval

---

```
<body>
  <span id="counter">0</span>
  <script>
    function increaseCountAfter1Sec() {

      let s = document.querySelector("#counter");
      s.innerHTML = parseInt(s.innerHTML) + 1;

    }

    setInterval(increaseCountAfter1Sec, 1000)
  </script>
</body>
```

הפונקציה תבצע את הקוד שוב ושוב לפי  
הזמן שנקבע:

יצרנו מונה שיגדיל בכל שנייה את הערך ב-  
span

\*הפסקת הפונקציה ע"י פונקציית  
.clearInterval



# חיקוי setIntvterval ע"י setTimeout

באמצעות רקורסיה ניתן לרשום קוד שחוזר על עצמו עם פונקציית setTimeout:

```
<body>
  <span id="counter">0</span>
  <script>
    function increaseCountAfter1Sec() {
      let s = document.querySelector("#counter");

      s.innerHTML = parseInt(s.innerHTML) + 1;
      setTimeout(increaseCountAfter1Sec, 1000)

    }

    setTimeout(increaseCountAfter1Sec, 1000)
  </script>
</body>
```

# תרגיל פותרים יחד

---

- כתוב count down timer הסופר את הזמן מעכשיו עד ההפסקה

0d 0h 29m 14s

- כאשר מגיעה ההפסקה יופיע

**TIME UP!!**

---

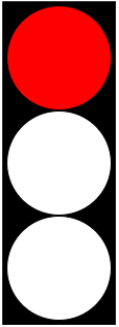
CountDount.html •

# clearInterval

---

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8" />
  <title></title>
</head>
<body>
  <progress value="0" max="10" id="progressBar"></progress>

  <script>
    let timeleft = 10;
    let downloadTimer = setInterval(function () {
      if (timeleft <= 0) {
        clearInterval(downloadTimer);
      }
      document.getElementById("progressBar").value = 10 - timeleft;
      timeleft -= 1;
    }, 1000);
  </script>
</body>
</html>
```



# תרגיל מסכם

צרו רמזור שעובד לפי החוקים הבאים:

- אור אדום – 5 שניות
- אור צהוב – 1 שנייה
- אור ירוק 5 שניות ולאחר מכן יחזור לצהוב וכן הלאה.

```
table { background-color: black; }  
td {  
    width: 30px;  
    height: 30px;  
    border-radius: 50%;  
    background-color: white;  
}
```