בס"ד

Full stack client

HTML DOM + Canvas

DOM HTML

- Document Object Model:
- רכיבי HTML מתפקדים כרכיבים.
- JavaScript יכול לשנות רכיבי HTML.
- כאשר אתה רוצה לגשת לרכיב HTML עם JavaScript, יש למצוא תחילה את האלמנט לפי מזהה.

```
<p id="demo"></p>

<script>
document.getElementById("demo").innerHTML = "Hello World";
console.log(document.getElementById("demo"));

</script>
```

אלמנט HTML canvas

- Canvas הוא רשת דו-ממדית.
- בפינה השמאלית העליונה של canvas יש את הקואורדינטות (0,0).
- ב- HTML, אלמנט <canvas> נראה כך:

```
<canvas width = "400" height = "400" style = "border: solid;" ></canvas>
```

- לרכיב <canvas> עצמו אין יכולות ציור. יש להשתמש ב-JavaScript כדי לצייר עליו גרפיקה כלשהי.

הוספת אלמנטים ל-canvas באמצעות HTML DOM (Document Object Model)

- כאשר דף אינטרנט נטען, הדפדפן יוצר מודל של האובייקטים בדף.
- השיטה ("getElementById("element ID – מחזיר את ה-reference של אלמנט ה-<canvas>.

```
var canvas = document.getElementById("myCanvas")
```

- השיטה ("getContext("2d מחזירה את ה-context אובייקט הכולל פעולות לציור.

```
var ctx = canvas.getContext("2d");
```

- קבלנו ממשק, שהוא חלק מ-.
- באמצעות Canvas API, ניתן ליצור אנימציה, גרפיקה של משחק, הדמיית נתונים, מניפולציה של תמונות ועיבוד וידאו בזמן אמת.

- ctx.fillStyle – הגדר צבע מילוי
- ctx.fillRect(X התחלה , Y התחלה , רוחב , גובה) – צייר מלבן

```
<canvas id="myCanvas" width="200" height="200" style="border: 1px solid;" ></canvas>

<script>
	var canvas = document.getElementById("myCanvas");
	var ctx = canvas.getContext("2d");
	ctx.fillRect(50,50,10,20);
</script>
```

שימוש ב-document.onkeydown event:
תזוזת אלמנט לפי מקשי חיצים

- JavaScript יכול להגיב לאירועי לחיצה על מקלדת
- דוגמה 1:

```
<p>keycode:</p>
<p id="keycode_value">--</p>

<script>
//-----------------------------------------------
   function keyDownEventHandler()
   {
      document.getElementById("keycode_value").innerHTML = event.keyCode;
      document.getElementById("y_value").innerHTML = y;
   }
//-----------------------------------------------

   document.onkeydown = keyDownEventHandler;
</script>
```

- דוגמה 2:

```
<canvas id="myCanvas" width="200" height="200" style="border:1px solid;"></canvas>
<br>
<p>X:</p>
<p id="x_value">--</p>
<p>Y:</p>
<p id="y_value">--</p>

<script>
//-----------------------------------------------------
   function keyDownEventHandler() {

      //clean
      ctx.fillStyle = 'white';
      ctx.fillRect(0, 0, 400, 300);

      if (event.keyCode == 40)
         y += 20;
      if (event.keyCode == 38)
         y -= 20;
      if (event.keyCode == 37)
         x -= 20;
      if (event.keyCode == 39)
         x += 20;

      document.getElementById("x_value").innerHTML = x;
      document.getElementById("y_value").innerHTML = y;

      //draw rectangular
      ctx.fillStyle = "black";
      ctx.fillRect(x, y, 20, 20);
   }
//-----------------------------------------------------

   canvas = document.getElementById('myCanvas');
   ctx = canvas.getContext('2d');

   var x = 40;
   var y = 40;

   document.onkeydown = keyDownEventHandler;

</script>
```

שימוש ב-setInterval event: תזוזת אלמנט לפי טיימר

- הפעולה setInterval() קוראת לפונקציה במרווחים שצוינו (במילישניות).
- דוגמה 1:

```
<p id="demo"></p>

<script>
    //---------------------------
    function print_hello() {
        element.innerHTML += "Hello"
    }
    //---------------------------

    var element = document.getElementById("demo");
    setInterval(print_hello, 1000);

</script>
```

- דוגמה 2:

```
<canvas id="myCanvas" width="400" height="300" style="border:1px solid;"></canvas>
<br>
<p>X:</p>
<p id="x_value">--</p>
<p>Y:</p>
<p id="y_value">--</p>

<script>
//-----------------------------------------------------
   function TimeEventHandler() {

      //clean
      ctx.fillStyle = 'white';
      ctx.fillRect(0, 0, 400, 300);

      x += 20;

      document.getElementById("x_value").innerHTML = x;
      document.getElementById("y_value").innerHTML = y;

      //draw rectangular
      ctx.fillStyle = "black";
      ctx.fillRect(x, y, 20, 20);

   }
//-----------------------------------------------------

   canvas = document.getElementById('myCanvas');
   ctx = canvas.getContext('2d');

   var x = 100;
   var y = 100;

   ctx.fillStyle = "black";
   ctx.fillRect(x, y, 20, 20);

   setInterval(TimeEventHandler, 300);

</script>
```