

תרגיל 3 מכונות פלונים

המטרה: לנפח בלון עד ... שהבלון יתפוצץ.

הביצוע: נציג תמונה של בלון, נשנה את גודל התמונה עד שהתמונה תגיע לגודל מסוים ואז נחליף את תמונת הבלון בתמונת בלון מתפוצץ.

א. תמונות:

1. חפש באינטרנט תמונה של בלון ותמונה המתארת בלון מתפוצץ ושמור אותם בתיקיית התמונות באתר.
תן לתמונות שם משמעותי.



baloon.png



baloonPop.png

למשל:

ב. הצגת התמונה בדף:

2. צור דף חדש, והצג במרכזו את תמונת הבלון.
בתגית img תן לתמונה שם: name = "ball"

```
<body>
  <br /><br /><br />
  <center>
    <img src = "pics/baloon.png" name = "ball" alt = "ballon" />
  </center>
</body>
```

ג. כתיבת ב-script:

```
<head>
<script language="javascript" >
```

3. נגדיר שני משתנים ballA ו-ballB שיכילו את התמונות המשתתפות – תמונת הבלון ותמונת הפיצוץ.
נגדי משתנה picSrc שיקבל את תמונת הבלון, כלומר את התמונה ballA:

```
var ballA = "pics/baloon.png";
var ballB = "pics/ballonPop.png";
var picSrc = ballA;
```

התכונה width קובעת את רוחב התמונה. כזכור, אם לא ציינו במקביל את התכונה height, יתאים עצמו גובה התמונה לרוחב. מכאן ששינוי תכונה ה-width משנה את גודל התמונה כולה.

מכיוון שגודל התמונה ישתנה בכל פעם, נשמור במשנה size את גודל התמונה ההתחלתי, ובמשתנה newSize את הגודל הנוכחי:

```
var size = 200;           הגדרת משתנה בשם size שערכו ההתחלתי הוא 200 :
var newSize = size;     הגדרת משתנה בשם newSize שערכו ההתחלתי כערכו של size :
                          ניתן להגדיר יותר ממשתנה אחר בשורת ההגדרה, במקרה זה יש להפריד בין ההגדרות בפסיק :
```

```
var size = 200, newSize = size;
```

נוסיף את הפונקציה show שתקבל את מקור התמונה ואת גודלה הנוכחי, ותקבע את תכונות התמונה ball שנקבעה בתגית img ב-body :

```
function show(picSrc, newSize)
{
    ball.src = picSrc;           // קביעת מקור התמונה
    ball.width = newSize;       // קביעת רוחב התמונה
}
```

נסגור את התגית `</script>` ונריץ את הדף.

שים ♥ : התמונה המוצגת הינה תוצאה של התגית img בלבד. הסקריפט לא השפיע עדיין על אופן הצגת התמונה.

4. נוסיף לסקריפט את הפונקציה `changImg()` :

- הפונקציה תופעל באופן מחזורי, בכל 2 מאיות השנייה (200 אלפיות השניה).
- לאחר כל 10 מחזורים שבהם יגדל הבלון ב- 10 נקודות, "יתפוצץ" הבלון, גודלו יחזור לגודל ההתחלתי ויתחיל תהליך הניפוח מחדש.
- הפונקציה פועלת על המשתנה i שערכו ההתחלתי הוא 1.
- בכל כניסה לפעולה (בעקבות מחזור חדש שנקבע ב- `setInterval`), תחושב שארית החלוקה של i ב- 10.
 - עבור ערכי i : $\Rightarrow 1, 2, 3, \dots, 8, 9, 0, 11, 12, \dots, 18, 19, 20, 21, 22, \dots$
 - j יקבל את הערכים הבאים : $\Rightarrow 1, 2, 3, \dots, 8, 9, 0, 1, 2, \dots, 8, 9, 0, 1, 2, \dots$
- בכל פעם ש- j מתאפס (היו 10 סיבובים) תוחלף תמונת הבלון בתמונת הפיצוץ וגודל התמונה יחזור לגודל המקורי.
- בסיום תופעל הפונקציה show שתקבל כפרמטר את המתמונה והגודל שנקבעו בסיבוב הנוכחי ותציג את התמונה בהתאם.
- ערכו של i יגדל ב-1 עבור הסיבוב הבא.

```

var i = 1;
setInterval("changImg()", 200);
function changImg ()
{
    var j = i % 10;
    if (j == 0) {
        picSrc = ballB;
        newSize = size;
    }
    else {
        picSrc = ballA;
        newSize += 10; // הוסף 10 ל- newSize
    }
    show(picSrc, newSize); // הגדל את ערכו של המשתנה ב- 1
    i++;
}

```

קיצורים בשפה:

הגדל את ערכו של המשתנה y ב-1 : $y = y + 1;$
 $y ++;$
הקטן את ערכו של המשתנה k ב-1 : $k = k - 1;$
 $k --;$
הוסף למשתנה a 10 : $a = a + 10;$
 $a += 10;$

5. אם נרצה לשנות את גודל מחזור הסיבובים, נשמור את מספר הסיבובים במשתנה מיוחד n . שינוי ערכו של n ישנה את מספר הפעמים שהבלון יגדל עד שיתפוצץ. נוכל לקלוט את ערך n בתיבת קלט `prompt`. (חשוב! היכן נשתמש במשתנה n ?) הקלט יתבצע פעם אחת בלבד, בהפעלה של הדף.

ד. לולאת השהייה:

נרצה שאחרי כל פיצוץ של בלון תהיה השהייה של 2 שניות.

6. פונקציית השהייה:

- הפונקציה (`delay`) מקבלת כפרמטר את מספר השניות (`sleep`) שיש להשהות את הביצוע.
 - בכניסה לפונקציה נשמרת השעה במשתנה `date` (שורה 2).
 - נוצר המשתנה `curDate` (הזמן הנוכחי) שבשלב זה ערכו ריק - `null`. (שורה 3).
 - מופעלת לולאה מסוג: **בצע ... כל עוד מתקיים ...** (שורות 4 - 6).
- הלולאה שמה ב- `curDate` את הזמן הנוכחי שוב ושוב (שורה 5) כל עוד מתקיים שהפרש הזמנים בין הכניסה ללולאה והזמן הנוכחי קטן ממספר השניות שב- `sleep` (שורה 6).

```

1. function delay(sleep) {
2.     var date = new Date();
3.     var curDate = null;
4.     do {
5.         curDate = new Date();
6.     } while (curDate - date < sleep);
7. }

```

הזמן נמדד באלפיות שנייה.

חיסור התאריך שב- `date` מהתאריך הנוכחי, ייתן את מספר אלפיות השנייה שחלפו בין התאריכים. ככל שמתרחשים יותר "סיבובי לולאה" עוברות יותר אלפיות השנייה עד אשר חולפות 2000 אלפיות השנייה, כלומר ל- 2 שניות.

7. נוסף לפונקציה start קטע קוד המפעיל את פונקציית ההשהייה.

הבלון מתנפח עבור כל ערך של i שאינו מתחלק במספר הסיבובים, ובכל פעם שהשארית מתאפסת, מתפוצץ. לכן נוסף בדיקה:

אם השארית שווה ל-0,

הפעל את לולאת ההשהייה עם הערך 2000

קבע את גודל הבלון הנוכחי newSize להיות הגודל ההתחלתי size

```
if (j == 0)
{
    delay(2000);
    newSize = size;
}
```

הפונקציה אחרי התוספת:

```
function start()
```

```
{
    j = i % n;
    if (j == 0)
    {
        delay(1000);
        newSize = size;
    }
    if (j == (n - 1))
    {
        newSize += 20;
        picSrc = ball1;
    }
    else
    {
        newSize += 10;
        picSrc = ball0;
    }
    show(picSrc, newSize);
    i++;
}
```

הקוד שהתווסף

אם זהו הניפוח האחרון בסיבוב זה הגדל את התמונה ב-20 נקודות החלף את התמונה לתמונת הפיצוץ

אחרת הגדל את התמונה ב-10 נקודות החלף את התמונה לתמונת ckuin

הפעלת הפונקציה דימי המקבלת את התמונה והגודל

תרגיל:

הוסף לדף עוד בלון בצבע אחר. כל בלון ינופח ויתפוצץ אחרי מספר אינטרוואלים (מחזורים) שונה. למשל, האחד אחרי n אינטרוואלים, והשני אחרי $n-3$ אינטרוואלים.

