

לולאות 1

מה זה:

לולאות הינם כלי לריצה בחזרתיות על בלוק פקודות מספר פעמים שהוגדרה מראש או לא ידועה מראש. כלומר, ישנם לולאות שאנחנו נגדיר כמה פעמים נרצה שיחזרו על אותו בלוק ויש לולאות שיחזרו על אותו בלוק כל זמן שעומדים בתנאי ריצה שלהם.

למה זה משמש:

לולאות משמשות אותנו במקרים שנרצה לחזור על אותם שורת קוד מספר פעמים (שיכול להשתנות מפעם לפעם) בלי לשכפל את אותם שורות קוד ו"לזבל" את הקוד שלנו. **לדוגמה:**
אני רוצה לקבל מספר מהמשתמש ולהכפיל אותו 5 פעמים ב2 ולהדפיס את התוצאה של כל ריצה, יותר קל אם נעשה זאת בעזרת 2 שורות קוד שיחזרו על עצמם כך:

```
int x = int.Parse(Console.ReadLine());  
x *= 2;  
Console.WriteLine(x);
```

בלי לולאה היינו צריכים לכתוב את השתי שורות קוד האחרונות 5 פעמים ו"לזבל" לעצמנו את הקוד(ומה יקרה אם נצטרך לחזור על הפעולה 40 פעמים? ...)

לולאות 2

סוגי לולאות:

קיימות שני סוגי לולאות עיקריים ומכל סוג נכיר כמה מהם בהמשך הדרך:

1. לולאה עם מספר ריצות לא מוגדר.

2. לולאה עם מספר ריצות מוגדר.

מספר חזרות לא מוגדר:

1. לולאת While (כל זמן): בלולאה זו מגדירים בתחילתה תנאי עצירה שכל זמן שהלולאה עומדת בתנאי, הלולאה

תמשיך לרוץ. **לדוגמה:**

```
int x = 5;  
while(x < 10) // בדיקה האם להיכנס לבלוק הפקודות  
{ תחילת בלוק הפקודות שיקרו בכל איטרציה //  
    Console.WriteLine(x * 3);  
    x++;  
} // סיום בלוק הפקודות של כל איטרציה
```

לולאות 3

2. לולאת do while (כל זמן, עם ביצוע אחד לפחות):

לולאה זו ממש כמו לולאת while שראינו הרגע, רק שאיתה אנחנו יודעים שהמערכת תרוץ לפחות פעם אחת על הבלוק פקודות של הלולאה.

כלומר do: תבצע פעולה/פעולות ואז תרוץ עליהן while כל זמן ש...

לדוגמה:

```
int x;                                // הצבת משתנה ללא ערך כלשהו
do
{
    // תחילת בלוק הפקודות שיתבצע בפעם הראשונה ובכל איטרציה
    x = int.Parse(Console.ReadLine());
    Console.WriteLine(x);
} while(x != 0);                      // סיום בלוק הפקודות ובדיקה האם לחזור על הבלוק
Console.WriteLine("my x = {0}",x);    // הדפסה של הערך הסופי של המשתנה שלנו **
```

** עם לולאת while רגילה התכנית לא הייתה רצה כי היא לא יכולה להדפיס משתנה שלא אותחל (x)

לולאות 4

מספר חזרות מוגדר:

לולאת for (למשך):

ללולאה זו יש שיטת הרכבה סטנדרטית: הצבה <= בדיקה <= השמה.

1. **הצבה:** מציבים משתנה int בשם i (קיצור של אינדקס) על ערך 0

2. **בדיקה:** בודקים שהמשתנה שיצרנו (i) לא הגיע למספר ריצות שהגדרנו לו לרוץ מראש.////

3. **השמה:** מגדילים את משתנה i ב-1.

אחרי שהגדרנו את הלולאה שלנו המערכת תרוץ בסדר הזה: ביצוע פעולות <= השמה <= בדיקה <=< יציאה

דוגמה:

```
int x = 0, y = 0;
for(int i = 0; i < 5; i++)
{
    x += 3;
    Console.WriteLine(x);
    y = x;
}
Console.WriteLine(y);
```

```
// x ו y הצבת משתנים
// שלושת השלבים להגדרת הלולאה עם 'i' בין שלב לשלב
// תחילת בלוק הפקודות שיבוצעו בכל איטרציה

// סוף בלוק הפקודות לכל איטרציה
// פעולה שתבצע רק אחרי סיום כל הלולאה.
```

לולאות 5

שימוש בbreak וcontinue:

break: "שבור" את הלולאה. כלומר, אם באמצע ריצה בתוך איזושהי לולאה השגנו את מה שאנחנו צריכים אנחנו נשתמש בbreak כדי שהמערכת לא תמשיך לרוץ על שורות מיותרות.

continue: תמשיך לאיטרציה הבאה בלי לסיים את האיטרציה הנוכחית

לדוגמה:

אם נרצה לבנות קוד שבהקשת 'Q' נצא מהלולאה. וב'C' נמשיך לאיטרציה הבאה. אחרת, נדפיס את מה שהוקש:

```
while(true)                                // לולאה אין סופית
{
    string s = Console.ReadLine();          // קבלת קלט מהמשתמש
    if(s == "Q")                             // צא מהלולאה
        break;
    else if(s == "C")                         // המשך לאיטרציה הבאה
        continue;
    else
        Console.WriteLine(s);               // אחרת תדפיס את הערך
}
Console.WriteLine("End Program");           // יקרה רק אחרי שהלולאה תסתיים
```

לולאות 6

תרגולים while:

1. בקש מספר מהמשתמש ושמור אותו במשתנה x ולאחמ"כ צור לולאת while שתרוץ כל זמן שא קטן מ20. שמור במשתנה נוסף את מספר הריצות שעשינו (שיתעדכן בכל איטרציה) והדפס אותו רק לאחר סיום הלולאה.
2. *אתגר (לא חובה) : צור לולאה בתוך הלולאה הראשית שתרוץ חצי ממה שא שווה (אם x שווה 16 אז תרוץ 8 פעמים וכדו') בתנאי שא גדול מ14 וגם שיהיה מספר זוגי
a. הלולאה הפנימית תרוץ הפוך (תקבל את החילוק של הא ובכל איטרציה תרד מספר אחד) ובכל ריצה תדפיס :
"Inner Loop : {*הערך הנוכחי*}"

קישורי עזר:

- [tutorialsteacher לולאת while:](#)
- [w3schools לולאת while ו do while](#)
- [tutorialsteacher לולאת do while:](#)

לולאות 7

תרגולים for:

1. קבל מהמשתמש מספר (int) גדול מ 3 ורוץ בלולאת for פי שתיים פעמים מהמספר שקיבלת (אם קיבלנו 4 נרוץ 8 פעמים וכדו') ובכל איטרציה הדפס את האינדקס
2. *אתגר* (לא חובה): צור לולאת for עם שתי הצבות של int הראשון = 0 והשני = 10 כלומר:

(int i = 0, int j = 10;...)

רוץ על הלולאה כל זמן ש-i קטן או שווה ל 10

ובכל איטרציה בשלב ה"השמה" הגדל את i ב 1 והקטן את j ב 1 ובבלוק פקודות הדפס את השווי של $j + i$. כך:
"i + j = {תוצאה}"

קישורי עזר:

- [W3Schools מעבר על for](#) (מי שרוצה יש שם עוד סוג לולאה שנכיר בהמשך)
- [torialsteacher הסבר על for](#)

לולאות 8

תרגולים Continue Break:

1. הרחב את תרגיל 1 של לולאת for:

א. אם זו איטרציה ראשונה דלג על ההדפסה.

ב. אם האינדקס גדול מ10 תצא מהלולאה

ג. אם זו איטרציה אחרונה הדפס גם את המילה "END" בתוך הלולאה(אחרי ההדפסה של האינדקס).

2. צור לולאת while איו סופית ובתוכה בקש מהמשתמש מספר. רק אם המשתמש הקיש את הספרות 23 הלולאה תיעצר. אחרת, המשתמש יתבקש שוב להזין מספר חדש. רק אחרי שהמשתמש הקיש 23 והלולאה "נשברה" הדפס למסך כמה פעמים הלולאה רצה.

קישורי עזר:

- [W3Schools מעבר continue break](#)
- [WebMaster הסבר בעברית על continue break:](#)

שיעורי בית לולאות