

## איך כותבים את זה - פסקל / Java

### טיפוסי משתנים:

Java	פסקל - Pascal	
<b>int</b> (בתחום ±2,147,483,648)	<b>integer</b> (בתחום ±32,767)	מספרי שלם
<b>long</b>	<b>longint</b> (בתחום ±2,147,483,648)	מספרי שלם ארוך
<b>double</b>	<b>real</b>	מספר ממשי
<b>char</b>	<b>char</b>	תו
String	<b>string</b>	מחרוזת

<b>static final</b> שם-הקבוע = טיפוס-נתונים <b>static final int</b> N = 10; <b>static final double</b> X = 2.34;	ערך = שם-הקבוע <b>const</b> ; <b>const</b> N = 10; X = 2.34;	<b>קבועים:</b> דוגמא:
--	---	--------------------------

<b>int</b> a, b; <b>double</b> x, y;	<b>var</b> a, b : <b>integer</b> ; x, y : <b>real</b> ; בפסקל אין אתחול בהגדרה	<b>הגדרת משתנים:</b>
---	---	----------------------

### מבנה התכנית:

Java	פסקל - Pascal
<b>public class</b> שם-המחלקה { <b>public static void</b> main(String[] args){ הגדרת המשתנים הוראה; הוראה; } }	שם-התכנית <b>program</b> ; <b>const</b> רשימת קבועים <b>var</b> הגדרת המשתנים <b>begin</b> הוראה; הוראה; <b>end.</b>

שם : ♥ מסתיים בנקודה

### תיעוד והערות:

// הערת שורה	אין. השתמש ב- { }	הערת שורה
/* הערה */	{ הערה } (* הערה *)	הערת קטע

- בפסקל אין חשיבות לגודל אות. ב-java יש חשיבות לגודל אות (case sensitive)
- משתנה מסוג **longint** בפסקל שקול למשתנה מסוג **int** ב-Java

קלט/פלט:

Java	Pascal - פסקל	
System.out.print ("מחרוזת"); System.out.println ("מחרוזת");	write ('מחרוזת'); writeln ('מחרוזת');	פלט מחרוזת
System.out.print (שם-משתנה); System.out.print (שם-משתנה + " " + שם-משתנה); System.out.println (שם-משתנה); System.out.println (שם-משתנה + " " + שם-משתנה);	write (שם-משתנה); write (שם-משתנה, ' ', שם-משתנה); writeln (שם-משתנה); writeln (שם-משתנה, ' ', שם-משתנה);	הדפסת תוכן משתנה
System.out.print (שם-משתנה + " מחרוזת"); System.out.print (שם-משתנה + " מחרוזת"); System.out.println (שם-משתנה + " מחרוזת"); System.out.println (שם-משתנה + " מחרוזת");	write (שם-משתנה, ' מחרוזת'); write (' מחרוזת ', שם-משתנה, ' מחרוזת'); writeln (שם-משתנה, ' מחרוזת'); writeln (' מחרוזת ', שם-משתנה, ' מחרוזת');	הדפסת משתנים ומחרוזות
	<b>var</b> : הגדרת משתנים:  a : integer; x : real; tav : char; str : string;	
<b>int</b> a = input.nextInt();	הוראות קלט בגוף התכנית : read (שם-משתנה, שם-משתנה); read (a, x, tav, str);	קלט מספר שלם
<b>double</b> x = input.nextDouble();	readln (שם-משתנה, שם-משתנה);	קלט מספר ממשי
<b>char</b> tav = input.next().charAt(0);	readln (a, x, tav);	קלט מספר תו
String str = input.next (); // קלט עד תו הרווח הראשון; String str = input.nextLine (); // קלט שורה	readln (str);	קלט מחרוזת

<ul style="list-style-type: none"> <li>בכל הוראת קלט ניתן לקלוט בדיוק משתנה אחד.</li> <li>לפני כל הוראת קלט יש להציג בקשה :</li> </ul> <pre>System.out.print (" → הקש מספר שלם"); <b>int</b> a = input.nextInt();</pre>	<ul style="list-style-type: none"> <li>ניתן לקלוט יותר ממשתנה אחד בהוראת הקלט.</li> <li>ניתן לקלוט למשתנים מסוגים שונים באותה הוראה הקלט.</li> <li>לפני כל הוראת read/readln יש לרשום הוראת write/writeln שתתאר את הקלט.</li> </ul> <pre>write (' → הקש מספר שלם'); read (a);</pre>	דגשים
---	---	-------

השמה:

a = 5;	a := 5;	
int a = 0;	לא קיים (כי המשתנים מוגדרים מחוץ לתכנית ב-var). כדי לאתחל משתנה נציב לתוכו את הערך הרצוי: a := 0;	אתחול בהגדרה

פעולות חישוביות:

Java	פסקל - Pascal	
int a, b, c; double x, y, z;	var a, b, c : integer; x, y, z : real;	
c = a + b;	c := a + b;	חיבור
c = a - b;	c := a - b;	חסור
c = a * b;	c := a * b;	כפל
שלם / שלם ← שלם ממשי / ממשי ← ממשי שלם / ממשי ← ממשי שלם / שלם (double) ← ממשי	שלם / שלם ← ממשי ממשי / ממשי ← ממשי שלם / ממשי ← ממשי ממשי / שלם ← ממשי	חילוק:
z = x / y; z = (double)a / b;	z := x / y; z := a / b;	חילוק בממשיים <ul style="list-style-type: none"> <li>המרה מפורשת ב-java</li> <li>המרה מרומזת בפסקל</li> </ul>
c = a / b;	c := a div b;	חילוק בשלמים (מנה)
c = a % b;	c := a mod b;	שארית
ניתן להמיר משלם לממשי ולהיפך בהמרה מפורשת (casting).	פעולת החילוק / מבצעת המרה אוטומטית לממשיים.	דגשים: <ul style="list-style-type: none"> <li>המרה</li> </ul>
	האופרטור / מחזיר תמיד ערך ממשי. div ו-mod אינם מוגדרים לממשיים. המשמעות – לא ניתן לבצע div ו-mod על מספרים ממשיים	<ul style="list-style-type: none"> <li>חילוק</li> </ul>

קיצורים:

a ++ ;	inc (a); { a := a + 1; }	הגדלה עצמית
a -- ;	dec (a); { a := a - 1; }	הקטנה עצמית

הוראות השפה:

Java		פסקל - Pascal		
שונה !=	שווה ==	שונה <>	שווה =	סוגי יחס
גדול או שווה >=	גדול מ- >	גדול או שווה >=	גדול מ- >	
קטן או שווה >=	קטן מ- <	קטן או שווה >=	קטן מ- <	

	בלוק הוראות:
<pre>{     בלוק הוראות }</pre>	<pre>begin     בלוק הוראות end;</pre> <p>אין לשים נקודה-פסיק ; אחרי המילה begin</p> <p>שים ♥ : מסתיים בנקודה-פסיק ;</p>

<p><b>if</b> (פסוק-לוגי) ביצוע</p> <pre>if (a &gt; 5)     b = b + 1;</pre>	<p><b>if</b> פסוק-לוגי <b>then</b> ביצוע</p> <pre>if a &gt; 5 then     b := b + 1;</pre>	<p><b>אם ...</b> (הוראה יחידה)</p>
<p><b>if</b> (פסוק-לוגי)</p> <pre>{     הוראה;     הוראה; }</pre> <p><b>if</b> (a == 3)</p> <pre>{     c = b * 2;     System.out.println (c + " : התוצאה"); }</pre>	<p><b>if</b> פסוק-לוגי <b>then begin</b></p> <pre>הוראה; הוראה; end;</pre> <p><b>if</b> a = 3 <b>then begin</b></p> <pre>c := b * 2; writeln (c , ' : התוצאה'); end;</pre>	<p><b>אם ...</b> (בלוק הוראות)</p>
<p><b>if</b> (פסוק-לוגי) ביצוע 1</p> <p><b>else</b> ביצוע 2</p> <p><b>if</b> (a != 5)</p> <pre>b = b + 1; else     b = b - 1;</pre>	<p><b>if</b> פסוק-לוגי <b>then</b> ביצוע 1</p> <p><b>else</b> ביצוע 2</p> <p><b>if</b> a &lt;&gt; 5 <b>then</b></p> <pre>b := b + 1; else     b := b - 1;</pre> <p>שים ♥ : אין לשים ; לפני else</p>	<p><b>אם ... אחרת ...</b> (הוראה יחידה)</p>

<pre> <b>if</b> (פסוק-לוגי) {     1הוראה;     2הוראה; } <b>else</b> {     3הוראה;     4הוראה; }  <b>if</b> (a &gt; 3) {     System.out.print (" → הקש מספר");     b = input.nextInt(); } <b>else</b> {     b = a;     c = c + 1; }                 </pre>	<pre> <b>if</b> פסוק-לוגי <b>then</b> <b>begin</b>     1הוראה;     2הוראה; <b>end</b> <b>else</b> <b>begin</b>     3הוראה;     4הוראה; <b>end</b>;  <b>if</b> a &gt; 3 <b>then</b> <b>begin</b>     write (' → הקש מספר');     read (b); <b>end</b> <b>else</b> <b>begin</b>     b := a;     c := c + 1 <b>end</b>;                 </pre>	<p><b>אם... אחרת...</b> (בלוק הוראות)</p> <p>שים ♥ : אין לשים ; לפני else</p>
<pre> <b>switch</b> (משתנה) {     <b>case</b> 1-ערך : ביצוע;                     <b>break</b>;     <b>case</b> 2-ערך : ביצוע;                     <b>break</b>;     <b>case</b> 3-ערך : הוראה; // ביצוע 3                     הוראה;                     <b>break</b>;     <b>case</b> 4-ערך : ביצוע;                     <b>break</b>;     <b>default</b> : ביצוע;                     <b>break</b>; }                 </pre> <p>רק ערכים בדידים</p>	<pre> <b>case</b> משתנה <b>of</b>     1-ערך : ביצוע;     2-ערך : ביצוע;     3-ערך : <b>begin</b>                 הוראה; { ביצוע 3 }                 הוראה;             <b>end</b>;     4-ערך : ביצוע;     <b>else</b> 5-ביצוע; <b>end</b>;      5..8 : תחום ערכים - מערך .. עד ערך :     1,5,7,9,12..16 : אחד מקבוצת נערכים :                 </pre>	<p><b>ברירת החלטה</b></p> <ul style="list-style-type: none"> <li>המשתנה הנבדק הוא מטיפוס סדור : מספר שלם או תו</li> <li>בפסקל : מתבצעת השורה המתאימה בלבד.</li> <li>ב-java : מבצע break עד ה- break</li> </ul>

**קשרים לוגיים**

&&	and	וגם
	or	או
!	not	לא
<pre> <b>if</b> ( a &gt; 3 &amp;&amp; (b == 5    c != 0) ) ...                 </pre>	<p>חובה לעטוף כל תנאי לוגי פשוט בסוגריים :  <b>if</b> (a &gt; 3) <b>and</b> ((b = 5) <b>or</b> (c &lt;&gt; 0)) <b>then</b> ...</p>	

## לולאות:

<p><b>for</b> (קידום-אינדקס; ערך-סיום <math>\leq</math> אינדקס; ערך-תחילי = אינדקס)</p> <pre>{     הוראה;     הוראה; }</pre> <p><b>for</b> (i = 1 ; i <math>\leq</math> 5 ; i++)</p> <pre>{     System.out.print(" מספר → ");     a = input.nextInt ();     sum = sum + a; }</pre>	<p><b>do</b> ערך-סיום <b>to</b> ערך-תחילי := אינדקס</p> <pre><b>begin</b>     הוראה;     הוראה; <b>end;</b></pre> <p><b>for</b> I := 1 <b>to</b> 5 <b>do</b></p> <pre><b>begin</b>     write (" מספר → ");     read (a);     sum := sum + a; <b>end;</b></pre>	<p><b>לולאת for</b> (לולאה עולה)</p>
<p><b>for</b> (הקטנת-אינדקס; ערך-סיום <math>&gt;</math> אינדקס; ערך-תחילי = אינדקס)</p> <pre>{     הוראה;     הוראה; }</pre> <p>sum = 0;</p> <p><b>for</b> (i = 5 ; i <math>&gt;=</math> 1 ; i --)</p> <pre>{     System.out.print (" מספר → ");     a = input.nextInt ();     sum = sum + a; }</pre> <p>System.out.println (sum + " : סכום המספרים");</p>	<p><b>do</b> ערך-סיום <b>downto</b> ערך-תחילי := אינדקס</p> <pre><b>begin</b>     הוראה;     הוראה; <b>end;</b></pre> <p>sum := 0;</p> <p><b>for</b> i := 5 <b>downto</b> 1 <b>do</b></p> <pre><b>begin</b>     write (" מספר → ");     read (a);     sum := sum + a; <b>end;</b></pre> <p>writeln (sum, ' : סכום המספרים');</p>	<p><b>לולאת for</b> (לולאה יורדת)</p>
<p><b>while</b> (ביטוי-לוגי)</p> <pre>{     הוראה;     הוראה; }</pre> <p>sum = 0;</p> <p>System.out.print (" מספר ראשון → ");</p> <p>a = input.nextInt ();</p> <p><b>while</b> (a <math>\neq</math> 999)</p> <pre>{     sum = sum + a;     System.out.print (" עוד מספר → ");     a = input.nextInt (); }</pre> <p>System.out.println (sum + " : סכום המספרים");</p>	<p><b>do</b> ביטוי-לוגי <b>while</b></p> <pre><b>begin</b>     הוראה;     הוראה; <b>end;</b></pre> <p>sum := 0;</p> <p>write (" מספר ראשון → ");</p> <p>read (a);</p> <p><b>while</b> a <math>\neq</math> 999 <b>do</b></p> <pre><b>begin</b>     sum := sum + a;     write (" עוד מספר → ");     read (a); <b>end;</b></pre> <p>writeln (sum, ' : סכום המספרים');</p>	<p><b>לולאת while</b></p>

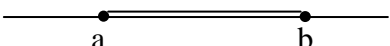
פונקציות מתמטיות:

Java			פסקל - Pascal			
	מתקבל	מוחזר		מתקבל	מוחזר	טיפוס הערך
Math.abs(x)	שלם ממשי	שלם ממשי	abs (x)	שלם ממשי	שלם ממשי	ערך מוחלט $ x $
Math.sqrt (x)	שלם ממשי	ממשי	sqrt (x)	שלם ממשי	ממשי	שורש ריבועי $\sqrt{x}$
Math.pow (x, 2)	שלם ממשי	ממשי	sqr (x)	שלם ממשי	שלם ממשי	חזקת 2 $x^2$
Math.pow (x, y)	שלם ממשי	ממשי	לא קיים (חישוב בלולאה)			חזקה $x^y$
(int) Math.round (x)	ממשי	שלם ארוך	round (x)	ממשי	שלם	עיגול לשלם הקרוב ( $4 \leftarrow 3.67$ )
(int) x	ממשי	שלם	trunc (x)	ממשי	שלם	החלק השלם ( $3 \leftarrow 3.67$ )

דגשים : במשפט הזימון לפונקציה חובה לציין מה לעשות עם הערך המוחזר :

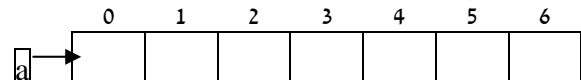
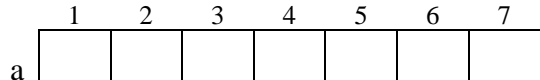
- השמה :  $x \leftarrow \text{functionName}$  (פרמטרים)
- שילוב בתנאי לוגי :  $\text{if (functionName (פרמטרים) > 25) ...}$
- הדפסת הערך המוחזר :  $\text{functionName}$  (פרמטרים) את כפלט את
- פרמטר לפונקציה אחרת :  $y \leftarrow \text{functionName1 (functionName2(פרמטרים) , ערך)}$

מספר אקראי

<b>import java.util.Random;</b>		לפני התכנית :
	יצירת עצם : randomize;	בתוך גוף בתכנית :
Random rnd = <b>new</b> Random();		
x = rnd.nextInt (n);	x := random (n);	מספר שלם בתחום 0 עד n (לא כולל) $0 \leq x < n$
x = rnd.nextInt (n) + 1;	x := random (n) + 1;	מספר שלם בתחום 1 עד n (כולל) $1 \leq x \leq n$
n = b - a + 1; x = rnd.nextInt (n) + a;	n := b - a + 1; x := random (n) + a;	מספר שלם בתחום a עד b (כולל) 

Java	פסקל - Pascal	פונקציות ופרוצדורות
<pre>static void procName (פרמטרים) { } </pre>	<pre>procedure procName (פרמטרים); var משתנים פנימיים ; begin : end; </pre>	פרוצדורה
<pre>static void aaa (int a, int b, double x) {     int i;     : } </pre>	<pre>procedure aaa (a, b: integer; x : real); var i : integer begin : end; </pre>	
<pre>static funcName(פרמטרים) טיפוס- מוחזר { : return ערך להחזרה; } </pre>	<pre>function funcName (פרמטרים): טיפוס-מוחזר; var משתנים פנימיים; begin : funcName := ערך להחזרה ; end; </pre>	פונקציה
<pre>static double sum (int a, int b, double x) {     double total ;     total = a + b + x;     return total; } </pre>	<pre>function sum (a, b : integer; x : real) : real; var total : real; begin     total := a + b + x;     sum := total; end; </pre>	

העברת פרמטרים בפסקל: לפי ערך או לפי כתובת (עם var).  
העברת פרמטרים בסיסיים (int, double, char, boolean) ב-Java: רק לפי ערך

<pre>arr = new [10] טיפוס-נתונים [] טיפוס-נתונים</pre>	<pre>var arr : array [1..10] of טיפוס-נתונים;</pre>	מערך חד-ממדי
<pre>int [] a = new int [7]; double [] x = new double [N];</pre>	<pre>var a : array [1..7] of integer; x : array [1..N] of real;</pre>	הגדרה
<pre>x[2] = 2 * a[0];</pre>	<pre>x [3] := 2 * a[1];</pre>	פנייה לתא
		

גודל המערך בפסקל: ערך תחילי עד ערך סיום (יכול להיות כל ערך שלם ובלבד שערך תחילי יהיה קטן מערך סיום).  
מספר התא הראשון במערך ב-Java יהיה תמיד 0.

<pre>for (int i = 0 ; i &lt; a.length ; i ++)</pre> <p>גודל המערך הוא a.length שם-המערך  מספר התא האחרון במערך הוא length - 1</p>	<pre>for i := 1 to 7 do</pre> <p>a[i] := i;</p>	סריקת מערך
---	---	------------

טיפוס-נתונים <code>arr = new [][]</code> טיפוס-נתונים <code>[5][7]</code> ;	טיפוס-נתונים <code>var mat : array [1..5,1..7] of</code> ;	מערך דו-ממדי (מטריצה)
<code>int [][] mat = new int [6][7] ;</code> <code>double [][] x = new double [N][M];</code>	<code>var mat : array [1..6,1..7] of integer;</code> <code>x : array [1..N,1..M] of real;</code>	
סכום התא ה"ראשון" והתא ה"אחרון" במערך: <code>mat[3][4] = mat [0][0] + mat [5][6];</code>	סכום התא ה"ראשון" והתא ה"אחרון" במערך: <code>mat[3,4] := mat[1,1] + mat[6,7] ;</code>	פניה לתא
<code>mat.length</code> ⇒ מספר השורות במטריצה <code>mat[i].length (1)</code> ⇒ מספר העמודות בשורה i <code>for (int i = 0 ; i &lt; mat.length ; i++)</code> <code>for (int j = 0 ; j &lt; mat[i].length ; j++)</code> <code>mat [i,j] = ....</code>	N = מספר השורות M = מספר העמודות <code>for i := 1 to N do</code> <code>for j := 1 to M do</code> <code>mat [i,j] := ...</code>	סריקת המערך

String str ; שים ♥ : String באות גדולה	<code>var str1 : string ;</code> <code>str2 : string [20];</code>	מחרוזת באורך 256 תווים מחרוזת באורך 20 תווים	מחרוזות
<code>str = "bla-bla-bla" ;</code> <code>str = new String ("bla-bla-bla") ;</code>	<code>str1 := 'bla-bla-bla' ;</code>		
מחרוזת ב-Java היא אובייקט			
<code>String str1 = input.next();</code> קלט מילה בודדת <code>String str2 = input.nextLine();</code> קלט משפט	<code>readln (str);</code>		קלט
<code>num = str.length();</code>	<code>num := length (str);</code>		אורך
<code>str = st1 + str2 + "aaa" ;</code>	<code>str := str1 + str2 + 'aaa' ;</code> <code>str := concat (str1, str2, 'aaa' );</code>		שרשור
<code>boolean equal = str1.equals(str2) ;</code> <code>int n = str1.compareTo (str2) ;</code> <code>str1 &gt; str2 ⇒ n &gt; 0</code> <code>str1 &lt; str2 ⇒ n &lt; 0</code> <code>str1 == str2 ⇒ n = 0</code>	<code>&gt; , &gt;= , &lt; , &lt;= , &lt;&gt; , =</code> סימני השוואה:		השוואה
<code>ch = str.charAt (i) ;</code> $0 \leq i < \text{str.length}()$	<code>ch := str [i];</code> $1 \leq i \leq \text{length}(\text{str})$		תו i במחרוזת
<code>int place = str.indexOf (ch) ;</code>	<code>place := pos (ch, str) ;</code> (place : integer)		חיפוש תו
<code>int place = str.indexOf (subStr);</code>	<code>place := pos (subStr, str) ;</code>		חיפוש תת-מחרוזת