

# דף עבודה – משתנים, פעולות חשבון ואופרטורים

שם התלמיד/ה: \_\_\_\_\_

תאריך: \_\_\_\_\_

בכל תרגיל:

1. צרו פרויקט חדש בשם המבוקש.
2. תכנו את מסך האפליקציה ב-**Designer**.
3. כתבו את הלוגיקה ב-**Blocks**.
4. הריצו ובדקו שהאפליקציה עובדת לפי הדרישה.

---

## תרגיל 1 – סכום של שני משתנים קבועים

שם הפרויקט: **SumTwoVars**

מטרה: תרגול ייצרת משתנים ושימוש בהם לחישוב.

בנו אפליקציה שמחזיקה שני מספרים קבועים (למשל 35 ו-45), מחשבת את הסכום שלהם, ומציג את התוצאה כאשר לוחצים על כפתור.

שאלה בדיקה:

אם המספרים הם 35 ו-45 – מה הסכום שיופיע על המסך?

תשובה: \_\_\_\_\_

---

## תרגיל 2 – חישוב בסיסי לשני מספרים

שם הפרויקט: **MiniCalcVars**

מטרה: חיבור, חיסור, כפל וחילוק בעזרת משתנים.

בנו אפליקציה שבה המשתמש מכניס שני מספרים, יוכל לבחור פעולה:

- כפתור לחישוב חיבור
- כפתור לחישוב חיסור

- כפטור לחישוב כפל
- כפטור לחישוב חילוק

בכל לחיצה יש להציג את התוצאה ב-Label.

**שאלה בדיקה:**  
מה התוצאה שתוצג לחישוב כפל אם המשתמש הכנס 3 ו-7?  
תשובה: \_\_\_\_\_

---

### תרגיל 3 – ממוצע של שלושה מספרים

**שם הפרויקט:** Average3

מטרה: שימוש בכמה משתנים ובחישוב ממוצע.  
המשתמש מכניס שלושה מספרים.  
האפקטיצה מחשבת את הממוצע שלהם ומציגה אותו על המסך.

**טיפ לחישיבה:**  
ממוצע של שלושה מספרים =  
$$(\text{מספר 1} + \text{מספר 2} + \text{מספר 3}) / 3$$

**שאלה בדיקה:**  
בדוק את עצמן:  
אם הכנסת את המספרים 4, 10, 7 – מה הממוצע?  
(כתב את התשובה כפי שהאפקטיצה מציגה)  
תשובה: \_\_\_\_\_

---

### תרגיל 4 – המרת שניות לדקות ושניות

**שם הפרויקט:** TimeConverter

מטרה: שימוש במנה שלמה (//) ובשארית (mod).  
בנו אפקטיצה שבה המשתמש מכניס מספר של **שניות** (למשל 134), והאפקטיצה מציגה:

- כמה **דקות** **שלמות** יש במספר זהה
- כמה **שניות** **נשארות** לאחר הורדת הדקות

לדוגמה:  
134 שניות → 2 דקות ו-14 שניות.

שאלה בדיקה:  
הכנס 200 שניות.  
כמה דקות וכמה שניות האפליקציה מציגה?  
דקות: \_\_\_\_\_ שניות: \_\_\_\_\_

---

## תרגיל 5 – מי גדול יותר?

שם הפרויקט: WhoIsBigger

מטרה: השוואת שני מספרים.

המשתמש מכניס שני מספרים.  
האפליקציה בודקת מי גדול יותר ומציגו אחת מהאפשרויות:

- "המספר הראשון גדול יותר"
- "המספר השני גדול יותר"
- "שניהם שוים"

שאלה בדיקה:  
מה תציג האפליקציה אם הכנסות:  
מספר ראשון: 12  
מספר שני: 5  
תשובה: \_\_\_\_\_

---

## תרגיל 6 – בודק גיל לסרט

שם הפרויקט: MovieAgeChecker

מטרה: שימוש בהשוואות ואופרטור לוגי (AND).

יש סרט שמיועד **גיל 14** ומעלה.  
המשתמש מכניס את הגיל שלו, והאפליקציה מציגו:

- "מותר לך להיכנס לסרט" – אם הוא עומד בתנאי
- " אסור לך להיכנס לסרט" – אחרת

אפשר להוסיף תנאי שגם גיל גדול מדי (למשל מעל 120) נחשב לא הגיוני.

**שאלה בדיקה:**

מה תציג האפליקציה אם הכנסת גיל 13?

תשובה: \_\_\_\_\_

---

## תרגיל 7 – זוגי או אי-זוגי?

**שם הפרויקט:** EvenOrOdd

מטרה: שימוש ב-`mod` כדי לבדוק אם מספר זוגי.

המשתמש מכניס מספר.  
האפליקציה בודקת האם המספר זוגי או אי-זוגי ומציגה:

- "המספר זוגי"
- "המספר אי-זוגי"

**رمز רעוני:**  
סקול את השארית של חלוקת המספר ב-2.

**שאלה בדיקה:**  
מה תציג האפליקציה אם הכנסת את המספר 17?  
תשובה: \_\_\_\_\_

---



## תרגיל 8 – תרגיל אתגר בלי גלאי עזר

**שם הפרויקט:** GuessMyNumber (אפשר לבחור שם אחר)

בתרגיל זהה אין הוראות מפורנות איך לבנות – רק תיאור כללי של האפליקציה.  
עליכם לתקן בעצמכם:

- איזה רכיבים לשימר ב-`SharedPreferences` Designator
- אילו משתנים תצטרכו
- איך להשתמש בפונקציות של מספר רנדומלי, השוואות, וכו'

### תיאור האפליקציה

אפליקציה של משחק ניהול מספר.

1. האפליקציה בוחרת באופן רנדומלי מספר סודי בין 1 ל-50.
2. המשתמש מנסה לנחש את המספר (על ידי כתיבה ב-`TextBox` ולהি�צה על כפתור).
3. בכל ניחוש, האפליקציה אומרת האם המספר:

גדול מדי

קטן מדי

או מדויק ("כל הכבוד! ניחשת נכון!")

4. אפשר להוסיף ספירת ניסיונות ולהציג כמה ניסיונות לקח עד לפגיעה.

תכננו לבדוק את המספר ואת הבולוקים.  
אין "מתכוון" – נסו להשתמש בכלים שלמדתם: משתנים, מספר רנדומלי, השוואות, ואולי גם OR/AND אם אתם רוצים תנאים מורכבים יותר.