

# דף עבודה – תנאים (ביצוע מותנה) ב-App Inventor

שם התלמיד/ה: \_\_\_\_\_

תאריך: \_\_\_\_\_

בכל תרגיל:

1. צרו פרויקט חדש בשם המבוקש.
2. תכננו את המסך ב-Designer.
3. כתבו את הלוגיקה ב-Blocks בעזרת בלוקי `if`, `if else`, `if else if` ובלוקי Logic.
4. הריצו ובדקו שהאפליקציה עובדת לפי הדרישה.

## תרגיל 1 – מי גדול יותר? (תנאי פשוט)

שם הפרויקט: `BiggerNumberSimple`

צרו פרויקט עם שני משתנים מספריים (בתחילת התוכנית), למשל:

`num1 = 10, num2 = 12`

בנו אפליקציה שבה:

- יש רכיב `Label` אחד ריק.
- בתוכנית (ב-Blocks) תבדקו בעזרת תנאי `if`:
  - אם `num1` גדול מ-`num2` – בטקסט של ה-`Label` יופיע: "מספר 1 גדול ממספר 2".
  - אם התנאי לא מתקיים – לא קורה כלום.

שאלת בדיקה:

אם `num1 = 10` ו-`num2 = 12`, האם הטקסט ב-`Label` ישתנה?

תשובה: \_\_\_\_\_

## תרגיל 2 – מי גדול יותר? (עם `if...else`)

שם הפרויקט: `BiggerNumberElse`

שפרו את האפליקציה של תרגיל 1:

- עדיין עובדים עם שני משתנים קבועים `num1, num2`.

- הפעם השתמשו בבולוק תנאי מורכב `if ... else`:
  - אם `num1` גדול מ-`num2` → "מספר 1 גדול ממספר 2".
  - אחרת (כל מקרה אחר) → "מספר 1 לא גדול ממספר 2".

---

## תרגיל 3 – השוואה מלאה בין שני מספרים (if – else if – else)

שם הפרויקט: `CompareTwoNumbers`

בנו אפליקציה שמשווה בין שני מספרים שהשתמשו מכניס:

Designer:

- שני `TextBox` (מספר ראשון, מספר שני) – `NumbersOnly`.
- כפתור "בדוק".
- `Label` לתוצאה.

התנהגות:

בעזרת מבנה תנאי מורכב `if ... else if ... else`:

1. אם המספר הראשון גדול מהשני → הציגו ב-`Label`:  
"המספר הראשון גדול מהשני".
2. אחרת אם המספר הראשון שווה לשני →  
"שני המספרים שווים".
3. אחרת (נשאר רק שהשני גדול מהראשון) →  
"המספר השני גדול מהראשון".

---

## תרגיל 4 – בודק ציון (תנאי מורכב עם else if)

שם הפרויקט: `GradeChecker`

Designer:

- `TextBox` – הכנסת ציון (0–100).
- כפתור "בדוק ציון".
- `Label` לתוצאה.

לוגיקה:

בעזרת תנאי מורכב:

- אם הציון קטן מ-55 → "נכשל".
- אחרת אם הציון בין 55 ל-84 (כולל) → "עבר".
- אחרת (85 ומעלה) → "עבר בהצטיינות".

---

## תרגיל 5 – שני תנאים ביחד (AND)

שם הפרויקט: PassBoth

רעיון: התלמיד עובר רק אם עבר גם במתמטיקה וגם באנגלית.

Designer:

- שני `TextBox`:
  - ציון במתמטיקה.
  - ציון באנגלית.
- כפתור "בדוק מעבר".
- `Label` לתוצאה.

לוגיקה (רעיונית):

- השתמשו בבלוקי `Logic` (תתיקטגוריה `Logic`) ובאופרטור `and`.
- התנאי למעבר:
  - ציון מתמטיקה  $\leq 55$  וגם
  - ציון אנגלית  $\leq 55$ .
- אם שני התנאים מתקיימים יחד → "עברת בשתי המקצועות".
- אחרת → "לא עברת לפחות במקצוע אחד".

---

## תרגיל 6 – לפחות תנאי אחד (OR)

שם הפרויקט: DiscountChecker

רעיון: הנחה בחנות ניתנת למי שזעיר מ-18 או מבוגר מ-65.

Designer:

- `TextBox` לגיל.
- כפתור "בדוק הנחה".
- `Label` לתשובה.

לוגיקה:

- בעזרת `if` ובלוק `or`:
  - אם הגיל קטן מ-18 או גדול מ-65 → "מגיע לך לקבל הנחה".
  - אחרת → "לא מגיעה לך הנחה".

---

## תרגיל 7 – תמונה שמופיעה בלחיצה (`Image` + תנאי פשוט)

שם הפרויקט: `ShowSmiley`

:Designer

- רכיב `Image` (תמונה) ריק בהתחלה.
- כפתור "הצג סמילי".

שלבים מומלצים:

- העלו לקובץ הפרויקט תמונה של סמילי (כמו בדוגמה במצגת – רכיב `Image` ו-`Media`).  
מצגת 4- ביצוע מותנה - תנאים
- בלחיצה על הכפתור – קבעו שתכונת `Picture` של רכיב התמונה תהיה קובץ הסמילי שהעליתם.
- ניתן לקבוע את הרוחב והגובה ל-`Fill parent` כך שהתמונה תתפוס את רוב המסך.

---

## תרגיל 8 – תחזית מזג אוויר לפי טמפרטורה (תנאי מורכב + תמונה)

שם הפרויקט: `WeatherByTemp`

:Designer

- `TextBox` – הכנסת טמפרטורה (מספר מעלות).
- כפתור "עדכן מזג אוויר".
- רכיב `Image` לתמונה.

העלו כמה תמונות למערכת (שלג, גשם, מעונן, חמים, שרב) – כמו ברעיון בתרגיל שבמצגת.

מצגת 4- ביצוע מותנה - תנאים

לוגיקה:

בעזרת `if - else if - else`:

- אם המספר קטן מ-0 → תמונה של מזג אוויר מושלג.
- אם 0-15 → תמונה של גשם.
- אם 15-22 → תמונה של מעונן.
- אם 22-30 → תמונה של מזג חמים.
- אם גדול מ-30 → תמונה של מזג שרבי.

(אפשר להוסיף גם שינוי צבע רקע המסך בהתאם לטמפרטורה.)

---

## תרגיל 9 – מציאת הגדול והקטן מבין ארבעה מספרים

שם הפרויקט: `MinMax4`

מטרה: שימוש בכמה תנאים כדי למצוא מינימום ומקסימום – המשך לתרגיל הסיכום במצגת.

מצגת 4- ביצוע מותנה - תנאים

:Designer

- ארבעה `TextBox` (מספר 1, 2, 3, 4).
- כפתור "חשב".
- שני רכיבי `Label`:

○ אחד לטקסט: "המספר הקטן ביותר הוא: ..."

○ אחד לטקסט: "המספר הגדול ביותר הוא: ...".

משימה:

השתמשו במשתנים ותנאים (רוצים / לא רוצים – `if / else if / else`, אפשר גם `AND`) כדי:

- למצוא מי המספר הקטן ביותר מבין הארבעה.
  - למצוא מי המספר הגדול ביותר מבין הארבעה.
  - להציג את שניהם ב-`Label`-ים המתאימים.
-

## תרגיל 10 – תרגיל אתגר בלי גלגלי עזר 🧠

שם הפרויקט: (תבחרו שם בעצמכם)

בתרגיל הזה אין הוראות צעד-אחר-צעד – רק תיאור כללי.  
עליכם לתכנן לבד:

- את מסך האפליקציה (איזה רכיבים ב-Designer).
- את המשתנים הדרושים.
- את התנאים (if / else if / else, AND/OR).
- את בלוקי ה-Blocks המתאימים.

### תיאור כללי של האפליקציה

בנו משחק שבו:

1. יש שלושה שחקנים (או יותר) שלכל אחד יש ניקוד.
2. המשתמש מכניס את הניקוד של כל שחקן.
3. האפליקציה מציגה:
  - מי המנצח (מי קיבל הכי הרבה נקודות).
  - האם יש תיקו בין שני שחקנים או יותר.
4. אם יש מנצח אחד ברור – אפשר להציג גם תמונה (למשל גביע קטן) לידו.

נסו להשתמש בכל מה שלמדנו: משתנים, תנאים פשוטים ומורכבים, AND/OR, ואפשר גם רכיב תמונה.  
אין פתרון אחד נכון – העיקר שהאפליקציה שלכם תעבוד בצורה הגיונית ותבדוק את כל המקרים.