

- הכרות עם שפת התכנות פייתון (PYTHON)
- הכרות עם סביבת העבודה של PYTHON
- תחביר השפה:
 - ✓ משתנים
 - ✓ סוגי משתנים
 - ✓ השמה
 - ✓ הדפסה
 - ✓ אופרטורים
 - ✓ תיעוד

הכל נמצא ברשת, צריך רק לדעת איפה לחפש:

אתרים ללמוד עצמי

- The Algorithm Design Manual by Steven S Skiena (highly recommended)
- [Python Documentation](#)
- [Think Python: How to Think Like a Computer Scientist](#) by Allen B. Downey
- [Python Programming](#) WikiBook
- [The Python Tutorial](#) official tutorial

➤ תחביר השפה:

✓ אופרטורים בוליאנים

✓ משפטי תנאי

✓ לולאות

אופרטורים בוליאנים

- ניזכר במשתנים בוליאנים
- נכיר אופרטורים יחסיים ולוגיים
- ננסח ביטויים בוליאנים
- נכיר טבלאות אמת

□ משתנים בוליאניים, Boolean variables, הם משתנים

שיכולים לקבל רק אחד משני ערכים

אמת True

או

שקר False

□ שימו לב: יש משמעות לאותיות גדולות וקטנות.

אופרטורים (Variables) בוליאנים וביטויים בוליאנים

□ אופרטור, operator, היא סוג של פקודה בפייתון, שמפעילים

על משתנים

□ ביטוי בוליאני, בדומה למשתנה בוליאני, יחזיר כפלט אחד משני

ערכים True או False

□ הביטוי $5 > 3$ יחזיר את הערך True, האופרטור הוא ">"

□ הביטוי $2 > 3$ יחזיר את הערך False, האופרטור הוא ">"

אופרטורים (Variables) בוליאנים

□ ישנן שתי קבוצות של אופרטורים בוליאניים:

❖ קבוצה אחת היא הקבוצה של האופרטורים **היחסיים** כגון גדול, קטן, שווה, ושונה. עבור האופרטורים היחסיים נהוג להשתמש בכתיב המזכיר מאד את הכתיב המתמטי.

❖ קבוצה שנייה היא של האופרטורים **הלוגיים**: AND, OR,


and	&
or	

NOT


אופרטורים (Variables) יחסיים

הטבלה הבאה מכילה רשימה של כל האופרטורים היחסיים:

משמעות	האופרטור
האם x שווה ל- y?	$x==y$
האם x שונה מ- y?	$x!=y$
האם x גדול מ- y?	$x>y$
האם x גדול או שווה ל- y?	$x>=y$
האם x קטן מ- y?	$x<y$
האם x קטן או שווה ל- y?	$x<=y$

שימו 

כאשר אנו רושמים $x=3$ המשמעות היא השמה של הערך 3 בתוך משתנה שנקרא x.
כאשר אנו רושמים $x==3$ המשמעות היא אופרטור הבדוק האם הערך של x שווה למספר 3.

שימו 

כאשר בודקים אם שני מספרים עשרוניים שווים, רצוי לא להשתמש בסימן השוואה ($==$) אלא לבדוק שההפרש בין שני המספרים מאוד קטן.
השוויון יכול להיות לא מדויק בגלל ייצוג הספרות לאחר הנקודה במחשב. לדוגמא אם המספרים S ו-T הם עשרוניים, ונרצה לבדוק אם הם שווים, נשתמש בפעולת התנאי הבאה: $\text{If } (\text{abs}(S-T) < 10^{-6})$:

□ טבלאות אמת

פעולת האופרטורים הלוגיים מתוארת באמצעות טבלאות אמת.

טבלת אמת היא טבלה המשמשת להצגת הערכים השונים שביטוי בוליאני יכול לקבל.

אופרטורים (Variables) לוגים

טבלת אמת עבור אופרטור "וגם" AND

הטבלה הבאה היא טבלת אמת של אופרטור **וגם** שסימנו הוא **and**:

הכוונה היא שכדי לקבל תשובת True, שני המשתנים חייבים להיות True. לדוגמא: כדי להתקבל למגמת פיזיקה בתיכון, עליך לקבל ציון סופי מעל 90 הן במתמטיקה והן במדעים, כלומר יש לקיים את שני התנאים.

A	B	A and B
True	False	False
False	True	False
False	False	False
True	True	True

אופרטורים (Variables) לוגים


טבלת אמת עבור האופרטור "או" OR

הטבלה הבאה היא טבלת אמת של אופרטור ה-**או** שסימנו הוא **or**:

הכוונה היא שנדי לקבל תשובת True, על אחד מהמשתנים (לפחות) להיות True.

לדוגמא: כדי לזכות בפרס מעורבות חברתית, עליך להתנדב מספר שנים או להיות פעיל במועצת תלמידים מספר שנים.

A	B	A or B
True	False	True
False	True	True
False	False	False
True	True	True

שימו 

בניגוד ל'או' שאנו מכירים מחיי היום-יום, אופרטור ה'או' משמעו שלפחות אחד מהמשתנים יהיה True.

באשר שניהם True – התוצאה של A או B עדיין תהיה True!

אופרטור השלילה NOT

פעולת אופרטור השלילה היא היפוך הערך הבוליאני.
במידה וערך הביטוי שמימינו הוא True הוא יהפוך ל- False וההיפך.
הטבלה הבאה היא טבלת אמת של אופרטור השלילה שסימנו not:

A	not A
True	False
False	True

אופרטורים (Variables) לוגים - תרגול

טבלה א

	A	B	not A and B
1	True	True	False and True = False
2	False	True	True and True = True
3	True	False	
4	False	False	

אופרטורים (Variables) לוגים - תרגול

טבלה ב

	A	B	not A and not B
1	True	True	
2	False	True	
3	True	False	
4	False	False	

משפטי תנאי

IF ➤

ELIF ➤

ELSE ➤

על מנת לכתוב תוכניות שימושיות, אנחנו צריכים לבדוק תנאים ולשנות את מהלך התוכנית בהתאם

```
x=2
if x > 0:
    print("x is positive")
```

הצבה

ביטוי בוליאני

נקודתיים:

שורת קוד שתבצע אם תוצאת הביטוי הבוליאני היא : True

הדפסה למסך

שאלה
מה יודפס למסך
אם נציב $x=-1$




```
x=2
```

```
if x > 0:
```

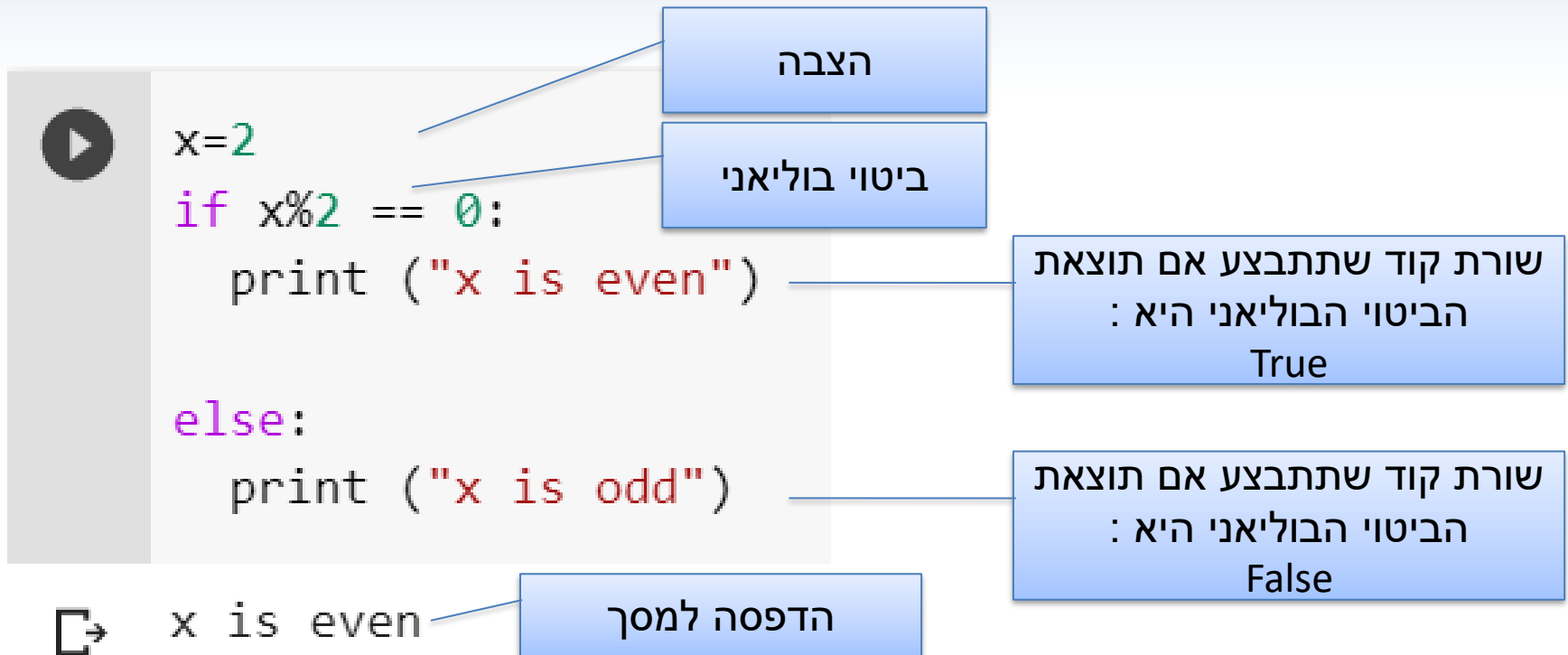
```
    print ("x is positive")
```


```
    x is positive
```

שימו 

המשפט בשורה שלאחר ה-if לא מתחיל בתחילת שורה, יש רווח קודם ("הזחה"). פייתון יבצע את רצף הפקודות שנמצאות בהזחה אם התנאי מתקיים.

משפטי תנאי if, else



מה קטע הקוד מבצע?
אם שארית החלוקה של הערך במשתנה `x` ב-2 היא 0 הביטוי יחזיר `True`
רק כאשר הערך במשתנה `x` הוא מספר זוגי התנאי יהיה `True`
אחרת (`else`) התנאי יהיה `False`
שימו  יש שני מצבים: זוגי או אי זוגי

שאלה
מה יודפס למסך
אם נציב `x=3`

כתבו תכנית המקבלת כקלט מספר **חיובי** כלשהו

אם המספר גדול מאפס וזוגי
ציירו שורה המורכבת מכוכביות כמספר שנקלט

לדוגמא עבור 8 יודפס:

* * * * *

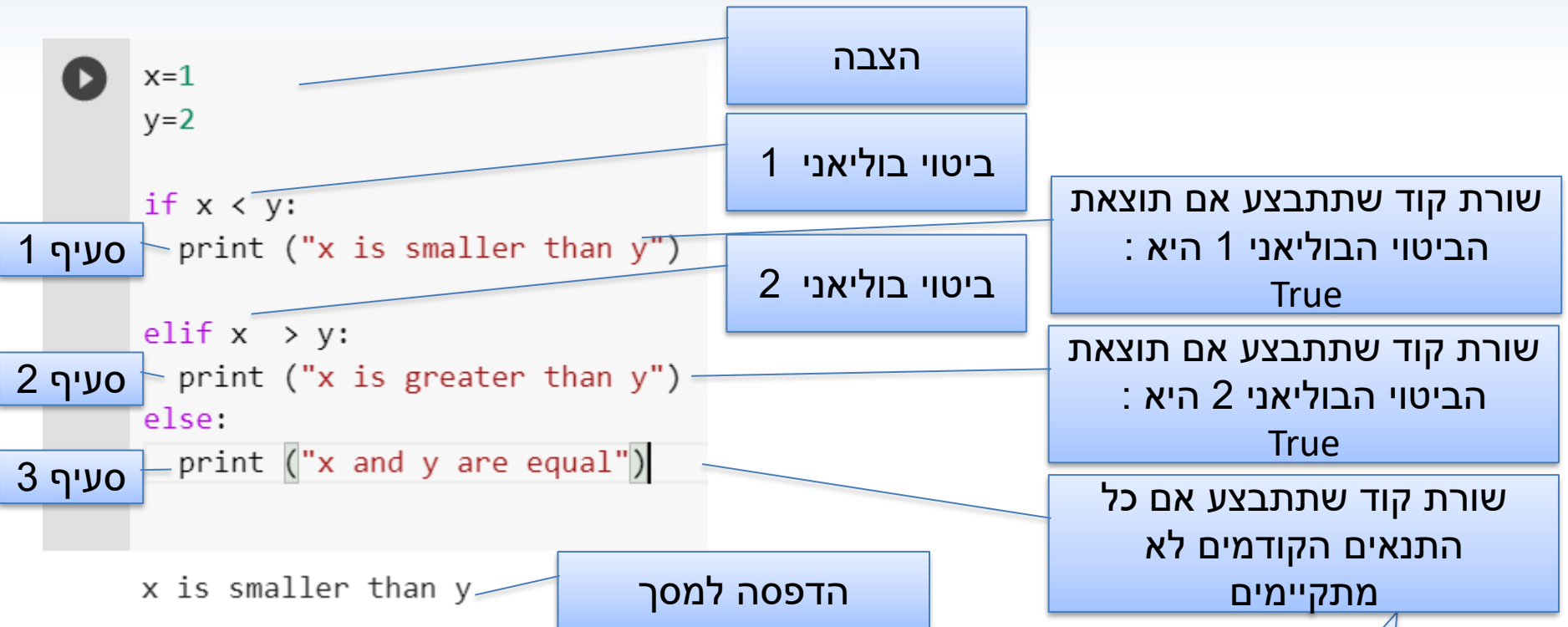
אחרת:

צייר שורה המורכבת מפלוסים כמספר שנקלט

לדוגמא עבור 5 יודפס:

+ + + + +

משפטי תנאי if, elif, else



elif זה קיצור של if else
רק סעיף אחד מתוך השלושה יתבצע.
אין כל מגבלה על מספר משפטי ה- elif
אם ישנו משפט else הוא חייב להיות אחרון, אך לא הכרחי
שהיה אחד בזה

כתוב\י תכנית המקבלת כקלט שני מספרים.
אם המספר הראשון הוא גדול מהמספר השני
התכנית תדפיס את המספר הראשון,
אחרת אם המספר השני שווה למספר הראשון
התכנית תדפיס "שווים"
אחרת התכנית תדפיס "את המספר השני"

לולאות

FOR לולאת ➤

WHILE לולאת ➤

מה קורה כאשר רוצים לחזור ולבצע מספר רב של פעמים
פקודות שחוזרות על עצמן בתוכנית?

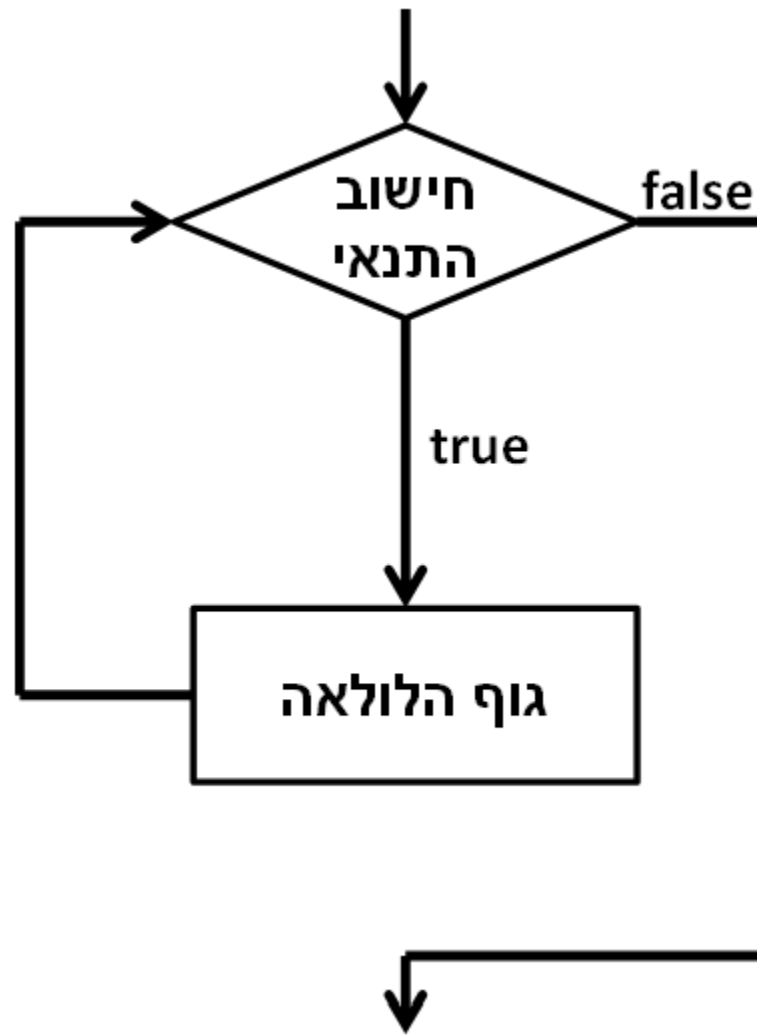
לולאות הן דבר מאוד שימושי שמאפשר לנו לחזור מספר רב של
פעמים על חלקים בתוכנית כאשר תנאי כלשהו מתקיים. כאשר
התנאי מפסיק להתקיים, החזרה על הפקודות מפסיקה.

הלולאות אותן תכירו הן: while ו- for..in.

while = כל עוד

ישנן בעיות תכנות שבהן מספר הפעמים שבהן לולאה אמורה לחזור על עצמה לא ידוע מראש, והלולאה חוזרת על עצמה כל עוד תנאי מסוים מתקיים.

בבעיות אלו משתמשים בלולאת ה- while



```
▶ countdown = 10  
  
while countdown > 0:  
  
    print (countdown)  
  
    countdown = countdown - 1  
  
print ("כל הכבוד על הספירה לאחור!")
```

הצבה

כותרת הלולאה

פקודה 1 בלולאה

פקודה 2 בלולאה

פקודה 3
מחוץ ללולאה

שימו לב

כל הפקודות שאמורות להתבצע בתוך לולאת ה-
while נכתבים בהזזה (על ידי הוספת טאב
[Tab] או 4 רווחים מצד שמאל)

העתיקו את התכנית לתוך סביבת עבודה
של פייתון והריצו אותה. מה התוצאה
שמתקבלת על המסך?

כיצד זה עובד?

1. אתחלנו את ערך המשתנה countdown ל-10.

2. בכניסה ללולאת ה-while מתבצעת הערכת התנאי: $countdown > 0$

3. האם תוצאתו true או false?

4. כל עוד תוצאת התנאי היא True בצע את הפקודות בגוף הלולאה:

- הדפסת ערך של countdown

- עדכון ערכו ע"י החסרת 1 מערכו הקודם

5. כאשר תוצאת התנאי היא False, צא מלולאת ה-while

6. לאחר יציאה מלולאת ה-while, הפקודות הבאות בתכנית

שימו לב

חשוב מאוד לשנות את ערך המשתנה/משתנים בלולאת התנאי על מנת שהתנאי בסופו של דבר יהפוך ל-False והלולאה תסתיים, אחרת הלולאה לעולם לא תסתיים ותהיה לולאה אינסופית.

```
▶ countdown = 10

while countdown > 0:

    print (countdown)

    countdown = countdown - 1

print ("כל הכבוד על הספירה לאחור")
```

```
☞ 10
9
8
7
6
5
4
3
2
1
כל הכבוד על הספירה לאחור
```

לולאות - while

```
▶ countdown = 10  
  
while countdown > 0:  
    print (countdown)  
    countdown = countdown - 1  
print ("כל הכבוד על הספירה לאחור")
```

```
☞ 10  
9  
8  
7  
6  
5  
4  
3  
2  
1  
כל הכבוד על הספירה לאחור
```

הצבה

כותרת הלולאה

פקודה 1

פקודה 2

פקודה 3

פלט

מה יודפס לאחר הרצת הקוד הבא?

```
x = 0
while x < 5:
    print("hello")
    x = x + 1
```

ד.
5

ג.
hello hello hello hello hello

ב.
hello
hello
hello
hello
hello

א.
hello
hello
hello
hello
hello
hello

מה יודפס לאחר הרצת הקוד הבא?

```
x = 0
while x < 5:
    print("hello")
    x = x + 1
```

ד. 5

ג. hello hello hello hello hello

ב. hello
hello
hello
hello
hello

א. hello
hello
hello
hello
hello
hello

כמה פעמים הלולאה בקוד הבא תתבצע?

```
x = 0
while x < 5 or x < 6:
    print("hello")
    x = x + 1
```

א.
7

ב.
5

ג.
6

ד.
לא תתבצע בכלל

כמה פעמים הלולאה בקוד הבא תתבצע?

```
x = 0
while x < 5 or x < 6:
    print("hello")
    x = x + 1
```

א.
7

ב.
5

ג.
6

ד.
לא תתבצע בכלל

כמה פעמים יודפס hello?

```
num = 10
while True:
    if num > 5:
        break
    print("hello")
    num = num + 1
```

ד.
לא יודפס בכלל

ג.
6

ב.
5

א.
7

כמה פעמים יודפס hello?

```
num = 10
while True:
    if num > 5:
        break
    print("hello")
    num = num + 1
```

ד.
לא יודפס בכלל

ג.
6

ב.
5

א.
7

מה יהיה הערך של num בסיום ההרצה?

```
num = 10
while True:
    if num > 5:
        break
    print("hello")
    num = num + 1
```

ד.
5

ג.
11

ב.
10

א.
2

מה יהיה הערך של num בסיום ההרצה?

```
num = 10
while True:
    if num > 5:
        break
    print("hello")
    num = num + 1
```

ד.
5

ג.
11

ב.
10

א.
2

for = לכל טווח ערכים

ישנן בעיות תכנות שבהן הלולאה חוזרת על עצמה לכל טווח
של ערכים. **הלולאה תתבצע מספר מוגדר של פעמים**

בבעיות אלו משתמשים בלולאת ה-for

טווח בין 5 מספרים
בתחילה $i=0$

```
for i in range(5):  
    print("hello")
```

`range(5)`

הפקודה



`range(0, 5, 1)`

והפקודה

נותנות תוצאה זהה

כמה פעמיים תוצג על המסך המילה hello?

א.
4

ב.
5

טווח בין 5 מספרים
בתחילה $i=0$

```
for i in range(5):  
    print("hello")
```

`range(5)`

הפקודה



`range(0, 5, 1)`

והפקודה

נותנות תוצאה זהה

כמה פעמיים תוצג על המסך המילה hello?

ב.
4

א.
5

טווח בין 5 מספרים
בתחילה $i=0$

```
for i in range(5):  
    print(i)
```

הפקודה `range(5)`



והפקודה `range(0, 5, 1)`

נותנות תוצאה זהה

מה יוצג על המסך?

א.
0
1
2
3
4

ב.
1
2
3
4
5

טווח בין 5 מספרים
בתחילה $i=0$

```
for i in range(5):  
    print(i)
```

הפקודה `range(5)`



והפקודה `range(0, 5, 1)`

נותנות תוצאה זהה

מה יוצג על המסך?



כלומר מתחילים
ב-0 ומסיימים
מספר אחד לפני
המספר שכתוב

ב.
0
1
2
3
4

א.
1
2
3
4
5


```
for i in range(1, 5):  
    print(i)
```

מ 1 עד 4

```
for i in range(1, 10, 2):  
    print(i)
```

מ 1 עד 9 בקפיצות של 2

```
for i in range(1, 5):  
    if i == 3:  
        continue  
    print(i)
```



מ 1 עד 4 לא כולל 3

1

כתבו לולאת for שרצה מ 0 עד 40

עבור מספרים זוגיים היא מדפיסה את המספר ולידו את המילה "זוגי"
עבור מספרים אי זוגיים היא מדפיסה את המספר ולידו את המילה "זוגי"

2

כתבו לולאת while שרצה כל עוד המשתמש לא רשם 999

עבור מספרים זוגיים היא מדפיסה את המספר ולידו את המילה "זוגי"
עבור מספרים אי זוגיים היא מדפיסה את המספר ולידו את המילה "זוגי"

3

כתבו תכנית המקבלת כקלט מהמשתמש מספר שלם בן 4 ספרות

ומדפיסה אותו בסדר הפוך

לדוגמא עבור קלט: 1234

יודפס: 4321

4

כתבו תכנית המקבלת כקלט שני מספרים שלמים מהמשתמש ומדפיסה את כל המספרים שביניהם (כולל את המספרים עצמם). לדוגמא, אם המשתמש הכניס את המספרים 3 ו-6 אז הפלט יהיה:

3

4

5

6