



לולאות ב-PYTHON

הילה קדמן
דצמבר 2019

לולאות

○ מה מבצע קטע הקוד שלהלן?

```
for n in range (50):  
    print (n, end = ", ")  
    if n % 7 == 0:  
        print (" boom !")  
print ()
```

לולאות

○ מה מבצע קטע הקוד שלהלן?

```
print ("7 boom")
```

```
for n in range (50):  
    print (n, end = ", ")  
    if n % 7 == 0:  
        print ("boom")  
print ()
```

הוראת for

גוף הלולאה

○ לולאה: הוראה המאפשרת לבצע שוב ושוב קטע קוד מסוים.

○ קטע הקוד המתבצע נקרא גוף הלולאה

○ שני סוגי לולאות:

לולאה המתבצעת מספר ידוע של פעמים - לולאת for

לולאה המתבצעת עד שיקרה משהו - לולאת while

לולאת FOR

- מבנה ההוראה:
- `for num in range (מספר שלם):`
הוראה
- `for` - כותרת הפעולה.
- `num` - משתנה מטיפוס מספר שלם. נקרא אינדקס הלולאה.
השם המקובל למשתנה זה הוא `i` (התחלה של המילה `index`)
ניתן להדפיס את האינדקס, לשאול על ערכו אבל **אסור** לשנותו.
- `range(n)` - קובע את תחום המספרים, יש לבצע את גוף הלולאה `n` פעמים,
`n` מספר שלם.
- `in range(n)` בִּתחום. מ-0 ועד (לא כולל) הערך המספרי שבסוגריים.
- השורה מסתיימת בסימן נקודותיים : שמגדיר שיש קטע הוראות לבצע
- כל ההוראות של גוף הלולאה נרשמות בהזחה של `tab` ביחס לשורת ה-`for`.
- ההוראות שבגוף הלולאה יכולות להיות כל אחת מההוראות בשפת פייתון
(כולל הוראות לולאה חדשות - קינון לולאות).

FOR לולאת

```
for i in range (10):  
    print (i)
```

```
for i in range (10):  
    print (i, end = " ")  
print()
```

0 1 2 3 4 5 6 7 8 9

0
1
2
3
4
5
6
7
8
9

○ קטע הקוד הבא יגרום להדפסת כל המספרים מ-0 ועד (לא כולל) 10, כל מספר בשורה חדשה.

○ קטע הקוד הבא יגרום להדפסת כל המספרים מ-0 ועד (לא כולל) 10, באותה שורה

end = " " יגרום להדפסת תו רווח אחרי כל ערך הדפסה, ללא ירידת שורה
end = ", " יגרום להדפסת פסיק ורווח אחרי כל ערך הדפסה, ללא ירידת שורה
end = "\t" יגרום לקפיצת tab אחרי כל ערך הדפסה, ללא ירידת שורה
תפקיד ה- print() שאחרי הלולאה - לגרום לירידת שורה.

לולאות

7 בום

- קטע הקוד מדפיס את כל המספרים בין 0 ו-50 (לא כולל)
- עבור כל מספר שמתחלק ב-7, ידפיס !boom וירד שורה

```
def main():  
    print ("7 boom")  
  
    for n in range (50):  
        print (n, end = ", ")  
        if n % 7 == 0:  
            print (" boom !")  
    print()
```

main()

```
7 boom  
0, boom !  
1, 2, 3, 4, 5, 6, 7, boom !  
8, 9, 10, 11, 12, 13, 14, boom !  
15, 16, 17, 18, 19, 20, 21, boom !  
22, 23, 24, 25, 26, 27, 28, boom !  
29, 30, 31, 32, 33, 34, 35, boom !  
36, 37, 38, 39, 40, 41, 42, boom !  
43, 44, 45, 46, 47, 48, 49, boom !
```

לולאות

7 בום

7 בום מודפס בקפיצות של tab בין הערכים

```
def main():  
    print ("7 boom")  
  
    for n in range (50):  
        print (n, end = "\t ")  
        if n % 7 == 0:  
            print (" boom !")  
    print()
```

end = "\t"
קפיצת tab בין ערכי הפלט

main()

```
7 boom  
0 boom !  
1 2 3 4 5 6 7 boom !  
8 9 10 11 12 13 14 boom !  
15 16 17 18 19 20 21 boom !  
22 23 24 25 26 27 28 boom !  
29 30 31 32 33 34 35 boom !  
36 37 38 39 40 41 42 boom !  
43 44 45 46 47 48 49 boom !
```

FOR לולאת

`for i in range (from, to, step):`

הוראה

הוראה

- `from` - הערך הראשוני שיקבל `num`
- `to` - ערך הסיום. עד איזה מספר, לא כולל ערך `to`
- `step` - גודל הקפיצה.
- פרמטרים אלו הם אופציונאליים.

לולאת FOR

○ דוגמאות:

○ הדפסת כל המספרים מ- 3 ועד 10 (לא כולל) בדילוג של 2:

```
for i in range(3,10, 2):  
    print(i, end = " ")  
print()
```

○ 3 5 7 9

○ הדפסת כל המספרים מ- 4 ועד 10 (לא כולל)

```
for i in range(4, 10):  
    print(i, end = " ")  
print()
```

○ 4 5 6 7 8 9

לא נקבע גודל הקפיצה
לכן בדילוגים של 1

לולאת FOR – לולאה יורדת

○ דוגמאות ללולאה יורדת:

○ הדפסת כל המספרים מ- 30 ועד 10 בדילוגים של -3:

```
for i in range(30,10, -3):  
    print(i, end = " ")  
print()
```

○ 30 27 24 21 18 15 12

שימו לב לערך
האחרון שיודפס

○ הדפסת כל המספרים העוקבים מ- 10 ועד 1

```
for i in range(10, 0, -1):  
    print(i, end = " ")  
print()
```

○ 10 9 8 7 6 5 4 3 2 1

לולאת FOR

○ דוגמאות:

○ הדפסת כל המספרים החיוביים הזוגיים הקטנים מ-20:

```
for i in range(2, 20, 2):  
    print(i, end=" ")  
print()
```

○ 2 4 6 8 10 12 14 16 18

○ ובדרך נוספת:

```
for i in range(1, 10):  
    print(i*2, end=" ")  
print()
```

○ 2 4 6 8 10 12 14 16 18

לולאות FOR

○ מה יהיה הפלט של הלולאה הבאה?

```
for i in range(10):  
    print (i, end = " ")  
print ()
```

○ 0 1 2 3 4 5 6 7 8 9

○ סה"כ 10 מספרים

○ איך תיראה לולאה שמבקשת להדפיס את כל המספרים
מ- 1 ועד 10 (כולל) ?

לולאות FOR

○ הדפסת המספרים מ- 1 ועד 10 (כולל): **דרך I**

```
for i in range(1,11):  
    print (i, end = " ")  
print ()
```

○ 1 2 3 4 5 6 7 8 9 10

○ הדפסת המספרים מ- 1 ועד 10 (כולל): **דרך II**

```
for n in range(10):  
    print (i+1, end = " ")  
print ()
```

○ 1 2 3 4 5 6 7 8 9 10

לולאות FOR

○ הדפסת כל המספרים שבין n1 ו-n2 (לא כולל):

```
def main():  
    n1 = int(input("enter 1st number --> "))  
    n2 = int(input("enter 2nd number --> "))  
  
    for i in range (n1, n2):  
        print (i, end=" ")  
    print()  
  
main()
```

```
enter 1st number --> 4  
enter 2nd number --> 12  
4 5 6 7 8 9 10 11
```

ומה יקרה אם ייקלט ל-n1
מספר גדול יותר מ-n2 ?

לולאות FOR

○ הדפסת כל המספרים שבין n1 ו-n2 (לא כולל):

```
def main():  
    n1 = int(input("enter 1st number --> "))  
    n2 = int(input("enter 2nd number --> "))  
  
    if n1 > n2:  
        temp = n1  
        n1 = n2  
        n2 = temp  
  
    for i in range (n1, n2):  
        print (i, end=" ")  
    print()
```

main()

○ אם המספר הראשון גדול מהשני,
החלף ביניהם

```
enter 1st number --> 12  
enter 2nd number --> 4  
4 5 6 7 8 9 10 11
```

```
enter 1st number --> 4  
enter 2nd number --> 12  
4 5 6 7 8 9 10 11
```

מחרוזת ולולאה

str [from : to : step]

○ גישה לאיברים במחרוזת:

```
def main():  
    print ("הצגת איברי המחרוזת")  
  
    st = "Hello World !"  
    n = len(st)  
  
    print ("as a string:", st[0:n:1])    # print (st)    :א
```

```
הצגת איברי המחרוזת  
as a string: Hello World !
```

main()

מחרוזת ולולאה

○ גישה לאיברים במחרוזת: `str [from : to : step]`

○ ניתן לגשת בלולאה לכל אחד מתווי המחרוזת:

```
def main():  
    print ("הצגת איברי המחרוזת")  
  
    st = "Hello World !"  
    n = len(st)  
  
    print ("as a string:", st[0:n:1])    # print (st) :א  
  
    print ("for loop:  ", end=" ")  
    for i in range (n):  
        print (st[i], end = " ")  
    print()
```

גישה לתו במיקום i

```
הצגת איברי המחרוזת  
as a string: Hello World !  
for loop:  H e l l o   W o r l d   !
```

main()

מחרוזת ולולאה

○ גישה לאיברים במחרוזת: `str [from : to : step]`

○ ניתן לגשת בלולאה לכל אחד מתווי המחרוזת:

```
def main():  
    print ("הצגת איברי המחרוזת")  
  
    st = "Hello World !"  
    n = len(st)  
  
    print ("as a string:", st[0:n:1])    # print (st) :א  
  
    print ("for loop:  ", end=" ")  
    for i in range (n):  
        print (st[i], end = " ")  
    print()  
  
    print ("by letters: ", end=" ")  
    for letter in st:  
        print (letter, end=" ")  
    print()  
  
main()
```

עבור כל אות במחרוזת

```
הצגת איברי המחרוזת  
as a string: Hello World !  
for loop:   H e l l o   W o r l d   !  
by letters: H e l l o   W o r l d   !
```

מחרוזת ולולאה

○ החלפת האות 'l' באות 'L' במחרוזת :hello world

```
def main():  
    print ("change 'l' to 'L' in string")  
    print()  
  
    st = "hello world"  
    print (st)  
  
    for letter in st:  
        if letter == 'l':  
            print ('L', end=" ")  
        else:  
            print (letter, end=" ")
```

```
main()
```

```
change 'l' to 'L' in string
```

```
hello world
```

```
h e L L o   w o r L d
```

```
^^^
```