

פייתון – סיכום שיעור 4

מחרוזות

מחרוזת היא אוסף של תווים. לכל תו יש מיקום במחרוזת כאשר הספירה מתחילה מ 0.

כדי להדפיס את אורך המחרוזת נשתמש בפונקציה `len`.

ניתן להדפיס תת מחרוזת כלומר חלק ממחרוזת בעזרת התחביר הבא:

`str_name[from:to:step]`

כאשר `from` – מאיזה מקום להתחיל, `to` – עד איזה מקום להגיע **לא כולל** המספר המצוין, `step` – בכמה לקפוץ כל פעם.

נראה דוגמא:

<pre>1 str = "Hello World" 2 print(str[0:5:1]) 3</pre>	Hello >
--	------------

בדוגמא אנו מדפיסים את תת המחרוזת אשר מתחילה מהמיקום 0 עד המיקום 5 **לא כולל** בקפיצות של 1.

הפונקציות `upper` & `lower`

הפונקציה `upper` הופכת כל אות קטנה לאות גדולה.

הפונקציה `lower` הופכת כל אות גדולה לאות קטנה.

הפונקציה `find`

הפונקציה `find` מחפשת תת מחרוזת בתוך מחרוזת.

הערך שהפעולה מחזירה הוא מיקומה של האות הראשונה של תת המחרוזת במחרוזת המפעילה, אם תת המחרוזת לא קיימת הערך שיוחזר יהיה -1, אם תת המחרוזת מופיעה פעמיים יוחזר מיקומה של האות הראשונה של המופע הראשון של תת המחרוזת.

נראה דוגמא:

<pre>1 str = "Hello World" 2 print(str.find("or")) 3</pre>	7 >
--	--------

בדוגמא אנו מחפשים את תת המחרוזת `or` במחרוזת `Hello World` ואכן תת המחרוזת הנ"ל נמצאת במחרוזת המפעילה ומיקום האות הראשונה – `o` במחרוזת המפעילה – `Hello World` הוא 7.

פונקציות נוספות

הפונקציה **replace** מבצעת החלפה של תו אחד באחר בעזרת התחביר הבא:

str.replace(from,to,times)

כאשר *from* – איזה תו להחליף, *to* – באיזה תו להחליף, *times* – כמה פעמים להחליף.

הפונקציה **isupper** מחזירה *True* אם כל האותיות במחרוזת הן אותיות **גדולות**, אחרת *False*.

הפונקציה **islower** מחזירה *True* אם כל האותיות במחרוזת הן אותיות **קטנות**, אחרת *False*.

הפונקציה **count** מחזירה את כמות המופעים של תו במחרוזת בעזרת התחביר הבא:

str.count(char,from,to)

כאשר *char* – איזה תו לספור, *from* – מאיזה מיקום להתחיל, *to* – עד איזה מיקום להגיע **לא כולל**.

הפונקציה **ord** מקבלת תו ומחזירה את המספר המתאים לו בטבלת ה *ASCII*.

הפונקציה **chr** מקבלת מספר ומחזירה את התו המתאים לו בטבלת ה *ASCII*.