

פונקציית zip מאפשרת לנו לבצע לולאה במקביל על מספר משתנים שהם מסוג iterables

```
A = ['eran', 'sharon', 'david', 'ross', 'shlomo']  
  
B = (2, 6, 9, 12, 15)  
  
C = {6.9, 7.8, 8.7, 9.6, 10.5}  
  
for tup in zip(A, B, C):  
    print(tup) # Output : ('eran', 2, 6.9) ('sharon', 6, 7.8) ...
```

```
for tup in zip(A, B, C):  
    for item in tup:  
        pass
```

## enumerate

פונקציית enumerate מקבלת משתנה מסוג iterable ומוסיפה counter (מונה) לכל אחד מהאיברים. ומחזירה tuple עם אותו אינדקס והמשתנה. האינדקס מתחיל מ-0 (אבל אפשר להחליט מאיפה)

```
names = ['Eran', 'Sharon', 'Yael', 'Kobi']  
for name in enumerate(names):  
    print(name)
```

```
(11, 'Eran')  
(22, 'Sharon')  
(33, 'Yael')  
(44, 'Kobi')
```

פלט:

פונקציית `join` עובדת בעיקר עם משתני iterables שמכילים מחרוזות אך יכולה לעבוד עם משתנים מסוג iterables שמכילים משתנים אחרים ולא רק מחרוזות (בעזרת `map`, נראה בהמשך)

מה שהפונקציה עושה, זה לאחד בין כל האיברים מאותו משתנה iterable למחרוזת אחת.

```
numbers = ['jpg', 'png', 'gif', 'bmp', 'jpeg']  
num_ste = '-'.join(numbers)
```

המחרוזת שתפריד  
בין כל מילה ומילה  
המחרוזת היא חובה

קריאה לפונקציה

משתנה מסוג  
iterable

הדפסה של `num_ste` תתן לנו את הפלט הבא:

```
jpg-png-gif-bmp-jpeg
```