

חלק יבש

(1)

```
my_str = "The first one is the easiest!"  
res_oneliner = my_str.split()[1::2]
```

הfonקציה `split()` הופכת את המחרוזת `my_str` לרשימה של מילים (מחולקת לפי רווחים).
הערך "1" אומר תחיל מהמילה השנייה כלומר אינדקס אחד.
הערך "2" הוא גודל הקפיצה בחיתוך הרשימה כלומר 1,3,5
הלוואה `while` מוסיפה את המילים האלו לתוך המשתנה `res`.
התוצאה הסופית `res` מכיל כל מילה שנייה, החל מהמילה השנייה(אינדקס 1).

(2)

```
n = 42  
print(*([f"{x} is divided by {n}.\\n" if x % n == 0  
       else f"the remainder of {x} divided by {n} is: {x % n}.\\n"  
       for x in range(100, 0, -3)), sep="")
```

הלוואה רצה על $x = 100, 97, 94 \dots 3$ (מתקדמת בקפיצות של 3
עבור כל x :

- אם x מתחלק בח ללא שארית, שמור את " x is divided by n ".
- אחרת שמור את " x divided by n is: $x \% n$ ".

הערכים מודפסים לפי הסדר בצורה הבאה `"sep=`" כר שהפלטים משורשים.

(3)

```
print(*([f"The ASCII number {i} represent the char '{chr(i)}'"  
        for i in range(0, max(ord('9'), ord('z'), ord('Z')) + 1)  
        if chr(i).isalpha() or chr(i).isdigit()),  
        sep="\n")
```

לולאה שרצה על i מ-0 עד $\max(\text{ord('9')}, \text{ord('z')}, \text{ord('Z'))} + 1$.
לאחר מכןתו מומר לתו באמצעות(`i`).`chr(i)`
הלוואה מדפיסה את הפלט רק אם התו הוא אות(גדולה או קטנה באנגלית) או מספר.

(4

```
list_c = [80, 121, 116, 104, 111, 110, 32, 105, 115, 32, 102, 117, 110, 33]
print("".join(map(chr, list_c)))
```

מאתחלים משתנה tmp_chr בתוכו "ריק"

הלוולאה רצה על כל הרשימה c_list ובעזרת chr(num) ממיר כל מספר בראשימה לתוכו,
מצרף למחרצת tmp_chr.

כלומר הקוד ממיר את המספרים לאותיות ומדפיס את המשפט שנוצר.