

## { יבש 5 }

מגשים:

אראג מחיטריאן: 315031203

חוסין עזקי: 213382104

הערה כללית לגבי התשובות: לא מעט פעמים השתמשנו באופרטור \* כדי להדפיס כמו בחלק מהקודים שדורשים. זה עלול להיות טיפה מבלבל לקריאה 😊.

1. לצורך קטע הקוד הבא, מוגדר עבורכם משתנה בשם my\_str המכילה מחרוזת, למשל:

```
my_str = "The first one is the easiest!"
```

הסבירו מה הקוד הבא עושה וכתבו oneliner שקול:

```
res = []
my_str = my_str.split()
first, second, third = 1, len(my_str), 2
while first < second:
    res.append(my_str[first])
    first += third
```

הקוד לעיל שם ברשימה res את כל המילים במקום אי-זוגי במחרוזת my\_str.

על מנת לקבל את אותה תוצאה ב Oneliner, נבצע את הפקודה הבאה:

```
res = [my_str.split()[i] for i in range(1, len(my_str.split()), 2)]
```

הפרמטרים של range הם 1 לאינדקס ההתחלה len(my\_str.split()) לאינדקס אחד אחרי הסוף, ו 2 כגודל צעד.

2. לצורך קטע הקוד הבא, מוגדר עבורכם המשתנה n (למשל n=42), הסבירו מה הקוד הבא עושה וכתבו oneliner שקול:

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

שימו לב – אין משמעות לסדר בו מודפסים האיברים, הניחו שבסוף קטע הקוד מסתיימת ריצת התכנית (ולכן אין צורך שהמשתנה my\_dictionary יכיל מידע), השתמשו בשיטת ההדפסה בקטע הקוד המצורף וכן שימו לב שניתן לפצל את השורה עם היא ארוכה לצורך קריאות).

הקוד לעיל מדפיס את הערכים שנטענו למילון my\_dictionary, הערכים שנטענו הם או הודעה שמתארת את שארית החלוקה של המפתח x ב n, או ההודעה הראשונה במקרה והשארית אפס שבה נכתב ש n מחלק את x.

המספרים הנבחרים בתור מפתחות הם המספרים מ 100 עד 0 בקפיצות יורדות של 3 לא כולל את 0 (בכללי לא כולל, כאן במקרה נעצור ב 1). Oneliner שקול לקוד לעיל

```
print(*{x: (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)).values())
```

3. קראו על הפונקציות `ord`, `chr` ב `python`. הסבירו מה הקוד הבא עושה וכתבו oneliner שקול:

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

הקוד לעיל אפקטיבית עובר על כל הערכי אסקי מ 0 עד 122 שהוא ערך האסקי של "z" קטן, ומודפסים הערכי אסקי והתווים שלהם אם הם מספרים או אותיות אנגלית.

הקוד הבא זהה ל Oneliner הבא:

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

4. לצורך קטע הקוד הבא, מוגדר עבורכם משתנה בשם `list_c` המכיל רשימה של מספרים שלמים, למשל:

```
list_c = [80, 121, 116, 104, 111, 110, 32, 105, 115, 32, 102, 117, 110, 33]
```

הסבירו מה הקוד הבא עושה וכתבו oneliner שקול:

```
tmp_chr = ""
for num in list_c:
    tmp_chr += chr(num)
print(tmp_chr)
```

רמז – קראו על המתודה `join` לערכים מסוג `string`, תוכלו להשתמש בה לצורך הפתרון.

הקוד לעיל בונה מחרוזת מתוך רשימת אינדקסי אסקי של התווים שלה ואז מדפיס אותה, המחרוזת היא

"Python is fun!"

Oneliner שקול לקוד לעיל :

```
print(''.join([chr(num) for num in list_c]))
```