# פייתון – מטלה 2

משקל המטלה = 4 נקודות. כל שאלה = נקודה אחת. אפשר לענות על חלק מהשאלות ולקבל ניקוד חלקי.

## שאלה 1: כתיבת תיעוד למודול

כתבו תוכנית, המקבלת כקלט שם של קובץ פייתון המכיל מודול מסויים, ושם של קובץ פלט, ויוצרת קובץ HTML המתעד את המודול: שמות הפונקציות שיש במודל, והתיעוד של כל פונקציה. דוגמה לשימוש:

python doc\_to\_html.py mymodule.py mydoc.html

לאחר ביצוע הפקודה, הקובץ mydoc.html יכיל את תיעוד המודול שבקובץ mymodule.

אפשר להניח שהמודול mymodule.py תקין.

[רמז: אין צורך לנתח ידנית את הטקסט שבקובץ mymodule.py. אפשר להשתמש בשיטות אחרות, כגון eval או import. אפשר לקבל רשימה של כל הפונקציות במודול ע"י הפקודה dir.]

הדגימו את פעולת הפונקציה שלכם.

## שאלה 2: קשטן שזוכר את הקלט האחרון

כתבו קשטן בשם @lastcall, המקשט פונקציה כלשהי עם פרמטר אחד. הקשטן בודק האם הקלט הנוכחי זהה לקלט הקודם. אם כן – הוא כותב הודעה מתאימה. אם לא – הוא מריץ את הפוקנציה כרגיל. דוגמה:

@lastcall

def f(x: int):

return x\*\*2

f(2) # returns 4

f(2) # prints “I already told you that the answer is 4!”

f(2) # prints “I already told you that the answer is 4!”

f(3) # returns 9

f(3) # prints “I already told you that the answer is 9!”

בדקו את הקשטן שלכם על כמה פונקציות.

## שאלה 3: מערך רב-ממדי

כתבו מבנה בשם List, שהוא זהה לרשימה של פייתון (list), אבל מאפשר לגשת לפריטים בתחביר של מערך רב-ממדי. לדוגמה:

mylist = List([

    [[1,2,3,33],[4,5,6,66]],

    [[7,8,9,99],[10,11,12,122]],

    [[13,14,15,155],[16,17,18,188]],

    ]

)

print(mylist[0,1,3])

אמור להדפיס 66 (שורה 0, מערך 1, איבר 3).

בכל שאר הדברים, המבנה אמור להתנהג כמו רשימה של פייתון, למשל:

print(mylist[0])

אמור להדפיס:

[[1,2,3,33],[4,5,6,66]]

[רמז: השתמשו בירושה ובהעמסת אופרטורים].

## שאלה 4: משחק תיכנות

בחרו שאלה אחת ברמה בינונית (medium): <https://www.codingame.com/training>

ופתרו אותה בעזרת פייתון.