



תרגיל הבנק - טסטים

המשימה: בדוק את מערכת הבנק באמצעות טסטים

הכן מבנה נתונים לטובת הטסטים:

```
bank_accounts = {
    1001: {
        "first_name": "Alice",
        "id_number": "123456789",
        "balance": 2500.50,
        "transactions_to_execute": [
            ("2024-08-17 14:00:00", 1001, 1002, 300), ("2024-08-17 15:00:00", 1001, 1003, 200)],
        "transaction_history": [
            ("2024-08-15 09:00:00", 1001, 1002, 500, "2024-08-15 09:30:00")]
    },
    1002: {
        "first_name": "Bob",
        "last_name": "Johnson",
        "id_number": "987654321",
        "balance": 3900.75,
        "transactions_to_execute": [],
        "transaction_history": []
    }
    ...
}
```



הכן פונקציית **perform trx** המקבלת כקלט את המילון, מספר חשבון ומבצעת את הטרנזקציות הממתנות לביצוע באותו החשבון

- כתוב טסט אשר מפעיל את הפונקציה ולאחר מכן בודק (assert) שרשימת הטרנזקציות לביצוע ריקה
- כתוב טסט אשר מפעיל את הפונקציה ולאחר מכן בודק (assert) שכל הטרנזקציות לביצוע נוספו לרשימת ההסטוריה
- כתוב טסט אשר מפעיל את הפונקציה ולאחר מכן בודק (assert) שהיתרה בחשבון הופחתה בהתאם לטרנזקציות לביצוע
- כתוב טסט אשר מפעיל את הפונקציה ולאחר מכן בודק (assert) שהיתרה בחשבונות היעד גדלה בהתאם לטרנזקציות לביצוע

הכן פונקציית **create trx** המקבלת כקלט את המילון, מספר חשבון, מספר חשבון יעד, כמות להעברה ומוסיפה את הטרנזקציה לרשימת הטרנזקציות הממתנות לביצוע באותו החשבון כ tuple

- כתוב טסט אשר מפעיל את הפונקציה ולאחר מכן בודק (assert) שאכן נוספה הטרנזקציה לרשימת הטרנזקציות הממתנות עם הפרטים הנכונים (אתגר: בדוק שגם נוסף הזמן הנוכחי)

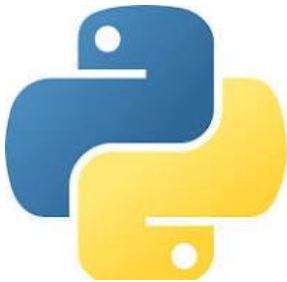
המשך בעמוד הבא ...

הכן פונקציית `get by name` המקבלת כקלט את המילון, שם פרטי ומחזירה את כל החשבונות המכילים את השם הפרטי שנקלט.

(התעלם מאותיות גדולות קטנות. שים לב: אם לדוגמא הוכנס השם `dan` אז השם `danny` יכלל בתשובה כי הוא מכיל את `dan`)

- כתוב טסט אשר מפעיל את הפונקציה ולאחר מכן בודק שחזרו החשבונות הנכונים לדוגמא שלח את החשבון שלמעלה בדוגמא ואת השם "bo", ובדוק שחזר חשבון 1002

בהצלחה!



את הפתרון יש לשלוח ל- pythonai170624+bank@gmail.com