מטלה 3

Python Algorithms for Industrial Engineers

by Hadas Lapid, PhD

סטודנטים.ות יקרים.ות,

במטלה זו תתבקשו לנתח בסיס נתונים בשם "employment.csv".

בסיס הנתונים מכיל נתוני העסקה של 1000 עובדים ממחלקות שונות.

בסיס הנתונים מכיל את העמודות הבאות:

Gender – משתנה בינארי (Male/Female)

Salary – משתנה רציף (float)

Bonus % - משתנה רציף (float)

Team – משתנה קטגורי (בן 11 קטגוריות)

בכל השאלות עליכם להטמיע את הפונקציות הכתובות בשלד.

המטלה מכילה 6 שאלות ו-7 פונקציות למימוש (פונקציה אחת היא פונקציית עזר).

יש לממש את הפונקציות **במקום** המילה pass, כולל הקלט והפלט של כל אחת ואחת מהן, ולבדוק את מימושן באמצעות שורות הבדיקה התואמות ב-main.

באם צלח המימוש, השאירו את הבדיקות עובדות. באם לא צלח המימוש, יש להסתירו בכדי לאפשר לקובץ לרוץ באופן תקין.

5 נקודות מתוך 100 שמורות לתקינות ריצת הפתרון. (לפתרון לא רץ יורדו 5 נק' באופן אוטומטי)

במידה ולא הצלחתם לממש שאלה כלשהי, ויש תלות בעיבוד קודם לצורך הפתרון, ישנם קבצי עזר, אותם ניתן להעלות ע"י שימוש בשורות המוסתרות לפני הבדיקות של שאלות 2, 5 ו-6.

בכל מקרה אין לשנות את שמות הפונקציות ו/או את הבדיקות המוצעות ב-main.

העלאת בסיס הנתונים לטבלה (df), שהיא מאפיין של המחלקה (Data_Analyzer), ממומשת עבורכם ב-main. כאשר מדפיסים את אובייקט המחלקה ((print(data) , ניתן לצפות בראשית הטבלה:

	Gender	Salary	Bonus %	Team
0	Male	97308	6.945	Marketing
1	Male	61933	4.170	NaN
2	Female	130590	11.858	Finance
3	Male	138705	9.340	Finance
4	Male	101004	1.389	Client Services

1. ממשו את הפונקציה omit_zeros

הפונקציה מקבלת רשימת עמודות, colnames, מחליפה את החוסרים (nans) בעמודות אלו באפסים, ומוחקת את השורות בהן מופיעים אפסים. העיבוד מתייחס לטבלה df של האובייקט data. הפונקציה לא מחזירה אף פרמטר.

הערה: לצורך הפתרון ניתן לעשות שימוש בפונקציות fillna, drop או בחיתוך תואם.

בדיקות:

גודל df אחרי העיבוד: (855, 4).

בבדיקת סכום האפסים בעמודות "Gender" , "Salary" ו- "Bonus %", עליהן מופעלת הפונקציה, יופיע 0 בכל אחת מהעמודות בהתאמה.

2. ממשו את הפונקציה calc_total

הפונקציה מקבלת שם של עמודה ראשונה (col1), שם של עמודה שניה (col2) ושם של עמודה חדשה (newcol).

באמצעות פונקציית עזר, f(), המופיעה בתוכה, הפונקציה יוצרת עמודה חדשה בשם newcol בטבלת df, כך שבעמודה החדשה מופיע החישוב של המשכורת הכוללת:

שכר הבסיס + הבונוס באחוזים משכר הבסיס.

או בנוסחא col1+col1*(col2/100)

כאשר col1 מייצגת בבדיקה את עמודת השכר ("Salary"), col2 מייצגת את עמודת הבונוס באחוזים ("Bonus %"), והשכר הכולל מיוצג על ידי העמודה החדשה newcol ("total").

הערה: חובה להשתמש בפונקציה apply של dataframe, ולעשות שימוש בפונקציית העזר f. פתרונות שלא יעשו שימוש בפונקציית העזר f לא יקבלו ניקוד.

בדיקות:

אחרי הפעלת data.calc_total("Salary", "Bonus %", "total"), ראשית הטבלה df תראה כך:

	Gender	Salary	Bonus %	Team	total
0	Male	97308	6.945	Marketing	104066.04060
1	Male	61933	4.170	0	64515.60610
2	Female	130590	11.858	Finance	146075.36220
3	Male	138705	9.340	Finance	151660.04700
4	Male	101004	1.389	Client Services	102406.94556

בבדיקת תוכן העמודה "total", שהתווספה בעקבות ההפעלה, יתקבל שוויון לחישוב מפורש של הנוסחא:

```
np.all(np.round((data.df.loc[:,"Salary"]*(1+(data.df.loc[:,"Bonus %"]/100))),1) ==
np.round(data.df.loc[:,"total"],1))
```

יחזיר True.

3. ממשו את הפונקציה select_group

הפונקציה מקבלת שם של עמודה קטגורית (group_col), ושם של קטגוריה (group).

הפונקציה מחזירה את השורות שבהן מופיעה הקטגוריה group בעמודה group_col.

הערה: לצורך הפתרון יש לעשות שימוש בפונקציה groupby ובאחת מהפונקציות שלה, על מנת לשלוף את תת-הקבוצה המתאימה.

הערה 2: הסוג המוחזר מהפונקציה הוא dataframe גם כן.

בדיקות:

בשליפת כל ה-female מעמודה "Gender", תוחזר טבלה שמידותיה (431, 5)

בשליפת כל ה-male מעמודה "Gender", תוחזר טבלה שמידותיה (424, 5)

4. ממשו את הפונקציה summ_by_group

הפונקציה מקבלת טבלת נתונים מסוג DataFrame (d), רשימת שמות עמודות קטגוריות (group_cols), רשימת שמות עמודות ספרתיות (val_col) ורשימת פונקציות לחישוב (funcs).

הפונקציה מחזירה טבלה מסכמת ובה חישוב הפונקציות funcs על העמודות הספרתיות val_col, פר קטגוריה מתוך העמודות הקטגוריאליות group_cols.

הערה: לצורך החישוב יש לעשות שימוש בפונקציה groupby ובה יש לעשות שימוש בפונקציה agg.

בדיקות:

בהפעלת הפונקציה על רשימת משכורות הנשים (females_salaries), עם חלוקה קטגורית לפי מחלקות (["Team"]), בניתוח השכר הכולל (["total"]), ובחישוב הממוצע פר קטגוריה ([np.mean]), תוחזר סדרה בת 11 איברים שמכילה את הממוצעים הבאים:

Team	mean
0	105917.546855
Business Development	101896.768858
Client Services	94966.007711
Distribution	89376.796850
Engineering	98979.220388
Finance	100339.476944
Human Resources	102738.704419
Legal	99583.600013
Marketing	104794.469473
Product	94768.875927
Sales	98960.892847

בהפעלת הפונקציה על רשימת משכורות הגברים (males_salaries), עם חלוקה קטגורית לפי מחלקות (["Team"]), בניתוח השכר הכולל (["total"]), ובחישוב הממוצע פר קטגוריה ([np.mean]), תוחזר סדרה בת 11 איברים שמכילה את הממוצעים הבאים:

Team	mean
0	96159.029114
Business Development	98397.259503
Client Services	102831.303804
Distribution	102056.535255
Engineering	108632.566072
Finance	105473.926771
Human Resources	100529.674708
Legal	94901.839660
Marketing	94712.626207
Product	97751.262883
Sales	102603.816227

5. ממשו את הפונקציה concat_dfs

הפונקציה מקבלת רשימה של שתי סדרות (dfs) עם ערך ברירת מחדל רשימה ריקה [ ], שם של עמודה ראשונה (col1) עם ערך ברירת מחדל "", שם של עמודה שניה (col2) עם ערך ברירת מחדל "", ושם של עמודה חדשה (newcol) עם ערך ברירת מחדל "diff".

הפונקציה מחברת את הסדרות בציר העמודות בהתאם לשמות השורות כולל כל הערכים הקיימים בכל אחת מהסדרות.

הפונקציה משנה את שמות העמודות כך שהעמודה הראשונה נקראת col1 והעמודה השניה נקראת col2.

הפונקציה מייצרת עמודה חדשה בשם newcol ובה ההפרש בין הערכים של עמודה col1 לערכים של עמודה col2.

לסיום, הפונקציה מאפסת את מספרי השורות (index) כך שיהיו בסדר עולה מ-0 ועד למספר השורות פחות 1.

הפונקציה מחזירה את הטבלה המאוחדת בעלת עמודת ההפרש החדשה.

אם לא הצלחתם לממש את שאלה 4, ניתן להעלות את הקבצים "females_salaries.csv", ואת "males_salaries.csv" לבצע השמה לסדרות females_salaries ו-males_salaries בהתאמה, כפי שממומש בשורות הנסתרות בקוד הבדיקה של שאלה זו.

בדיקה:

כאשר מפעילים את concat_dfs עם הסדרות [females_salaries,males_salaries], "females" בתור col1, "males" בתור col2 ו-"diff" בתור newcol, מתקבלת טבלה בת (11,4) שנראית כך:

	Team	females	males	diff
0	0	105917.546855	96159.029114	9758.517740
1	Business Development	101896.768858	98397.259503	3499.509355
2	Client Services	94966.007711	102831.303804	-7865.296093
3	Distribution	89376.796850	102056.535255	-12679.738405
4	Engineering	98979.220388	108632.566072	-9653.345684
5	Finance	100339.476944	105473.926771	-5134.449827
6	Human Resources	102738.704419	100529.674708	2209.029712
7	Legal	99583.600013	94901.839660	4681.760353
8	Marketing	104794.469473	94712.626207	10081.843266
9	Product	94768.875927	97751.262883	-2982.386956
10	Sales	98960.892847	102603.816227	-3642.923381

6. ממשו את הפונקציות merge_sort ואת פונקציית העזר merge

הפונקציה הרקורסיבית merge_sort מקבלת:

- טבלה-DataFrame בשם L
- שם של עמודה (col) בעלת ערך ברירת מחדל "diff".

הפונקציה ממיינת בצורת פיצול ומיזוג (merge sort) את טבלת הקלט L, בהתאם לערכים המופיעים בעמודה col בטבלה.
בתחילה, merge_sort מפצלת את L לשניים לפי מימד השורות, עד שמגיעה לשורות בודדות.

בתנאי העצירה, היא מחזירה את הטבלאות המכילות שורה בודדת.
לאחר הפיצול merge_sort ממזגת את הטבלאות המפוצלות על ידי הפעלת הפונקציה merge.
פונקציית המיזוג, merge, מקבלת כקלט:

- תת-טבלה מסוג DataFrame בשם left, ממויינת לפי העמודה col
- תת-טבלה מסוג DataFrame בשם right, ממויינת לפי העמודה col
- שם עמודה בשם col, לפיה מתבצע המיזוג. ערך ברירת המחדל של col הוא "diff"

הפונקציה merge מחברת בין השורות של left לבין השורות של right ומחזירה טבלה אחת המכילה את כל השורות של left ושל right, ממויינות בסדר עולה בהתאם לערכים המופיעים בעמודה col.

הערה: המימוש האלגוריתמי מתבצע בדיוק כפי שהודגם בכיתה עבור רשימה, רק שכאן יש לפצל ולמזג טבלאות (DataFrames) לפי הערכים המופיעים בעמודה ספציפית.

הערה 2: אם הפלט של שאלה 5 לא תואם את הדרישה, ניתן להעלות אותו באופן עצמאי מהקובץ "join_dfs.csv" על ידי הפעלת השורה join_dfs = pd.read_csv("join_dfs.csv").

בדיקה: אם הקלט של merge_sort הוא טבלת הפלט של שאלה 5, ו-col לפיה מתבצע המיון היא העמודה diff.

	Team	females	males	diff
0	0	105917.546855	96159.029114	9758.517740
1	Business Development	101896.768858	98397.259503	3499.509355
2	Client Services	94966.007711	102831.303804	-7865.296093
3	Distribution	89376.796850	102056.535255	-12679.738405
4	Engineering	98979.220388	108632.566072	-9653.345684
5	Finance	100339.476944	105473.926771	-5134.449827
6	Human Resources	102738.704419	100529.674708	2209.029712
7	Legal	99583.600013	94901.839660	4681.760353
8	Marketing	104794.469473	94712.626207	10081.843266
9	Product	94768.875927	97751.262883	-2982.386956
10	Sales	98960.892847	102603.816227	-3642.923381

אז פלט הפונקציה merge_sort לאחר אתחול האינדקסים של טבלת הפלט (reset_index) יראה כך:

	Team	diff	females	males
0	Distribution	-12679.738405	89376.796850	102056.535255
1	Engineering	-9653.345684	98979.220388	108632.566072
2	Client Services	-7865.296093	94966.007711	102831.303804
3	Finance	-5134.449827	100339.476944	105473.926771
4	Sales	-3642.923381	98960.892847	102603.816227
5	Product	-2982.386956	94768.875927	97751.262883
6	Human Resources	2209.029712	102738.704419	100529.674708
7	Business Development	3499.509355	101896.768858	98397.259503
8	Legal	4681.760353	99583.600013	94901.839660
9	0	9758.517740	105917.546855	96159.029114
10	Marketing	10081.843266	104794.469473	94712.626207

בדקו היטב את פתרונכם, ודאו שאתם מעלים קובץ רץ, עם הת.ז. שלכם, שמכיל את הפתרון ולא מכיל אף הערה בעברית או בכל שפה אחרת מלבד אנגלית / פייתון.

בהצלחה רבה,

צוות הקורס