

מבחן פנדס

לפניכם מבחן פייתון העוסק בספריית pandas השימוש ב-AI לכל צורותיו אסור בהחלט. מותר להשתמש באתרים הבאים :

<https://docs.python.org/3/>
<https://www.geeksforgeeks.org/>
<https://www.w3schools.com/>
<https://stackoverflow.com/>
<https://realpython.com/>
<https://pandas.pydata.org/docs/>

אין להשתמש בספרייה אחרת מלבד pandas וספריות מובנות של פייתון
אנא קראו היטב את כל ההנחיות בטרם התחלת ביצוע המשימה.

היעד במבחן זה: ליצור קובץ CSV המייצג את כל השינויים וההגדרות המפורטים בהמשך על אוסף הנתונים הנתון.

הנחיות הגשה:

יש להגיש למודל:

- את כתובת ה repo כמידי מבחן, יש לדאוג שכל הקבצים הבאים יעלו ל: repo:
a. קבצי המקור - קובץ פונקציות [utils.py](#) וקובץ [main.py](#), ללא הפרדה לתיקיות.
b. קובץ [readme](#)
c. קובץ הנתונים המקורי [order_sample.json](#)
2. יש להעלות למטלה במודל את קובץ התוצאה בשם [clean_orders_\[ID_NUMBER\].csv](#)

יש לתכנן את הקוד בחלוקה מודולרית לפונקציות (אין חובה למחלקות ואובייקטים), יש להפריד לקובץ פונקציות [utils.py](#) וקובץ [main.py](#), ללא הפרדה לתיקיות.
יש להקפיד על תעוד, שמות משתנים משמעותיים, פונקציות קצרות ונכונות.

בכל שאלה תתבקשו לבצע את הפעולות הבאות או חלק מהן, כפי שיפורט בסוגריים בכל שלב:

1. שינוי עמודה קיימת ב-DF (החלפת עמודה לאחר ניקוי או שינוי הנתונים)
2. הוספת עמודה על סמך נתונים מעמודה קיימת ב-DF
3. שינוי סדר הרשומות ב-DF (מיון)
4. מחיקת חלק מהרשומות (פילטר)

תוצר נדרש (פלט):

- יש להגדיר ולבנות DF שעבר את כל פעולות העיבוד שלנו (למעט פעולות הדפסה למסך)
- הטבלה הסופית תכיל את כל העמודות מהטבלה המקורית, חלקן לאחר עיבוד, ובנוסף עמודות נוספות שיצרנו, לאחר שהסרנו רשומות שפילטרנו החוצה ובסדר המיון כפי שהוגדר במהלך המטלה.
- ה DF יישמר בסוף התהליך כקובץ במבנה טבלאי מסוג CSV.

קלט:

- אוסף הנתונים למבחן מבוסס על קובץ json בשם [order_sample](#), קובץ ה json הוא דאטה-סט עסקי יומיומי של עסק אינטרנטי.
- כל אובייקט (רשומה) מייצג הזמנה אחת באתר ומכיל פרטים רבים אודותיה.

שלב 0 - (טעינת קובץ)

טען את קובץ ה-json לצורת DataFrame.

בשלב זה עליכם להבין מול איזה נתונים אתם עובדים, איך הנתונים נקלטו ומבנה כללי של הרשומה (יש לוודא שמספר השורות והעמודות תואם לזה שבקובץ).

המלצה (רשות): בכדי לבצע את המטלה שלפניכם, מומלץ לבדוק את הפרטים הבאים (בכל מקרה אסור להכניס את הנתונים הנל לקובץ הסופי)

10 שורות ראשונות

מספר שורות ועמודות

סוגי העמודות כפי שנקלטו בתהליך טעינת הקובץ

בדקו עבור כל עמודה כמה ערכים ריקים/חסרים

בדקו עבור כל עמודה כמה ערכים ייחודיים

בדקו עבור כל עמודה סטטיסטיקות מספריות

שלב 1 - (החלפת עמודות קיימות)

במידת הצורך המר את העמודות הבאות לסוגי נתונים המתאימים (בצעו פעולות ניקיון והמרה):

Total_amount: float (\$ לסימן ה)

Shipping_days: int

Customer_age: int

Rating: float

Order_date: datetime

שלב 2 - (החלפת עמודה קיימת)

הסר את סימוני ה-HTML מעמודת items_html כך שתכיל טקסט נקי בלבד (תוך הקפדה שיישאר הרווחים בין מילים שונות).

שלב 3 - (החלפת עמודה קיימת)

בעמודת coupon_used החלף ערך ריק ("") ב"coupon no" אחרת השאר ערך קיים.

שלב 4 - (יצירת עמודה חדשה בסוף הטבלה)

צור עמודה חדשה בשם order_month המכילה את חודש ההזמנה כערך מספרי בין 1 ל-12 על פי order_date.

שלב 5 - (יצירת עמודה חדשה בסוף הטבלה ומיון הטבלה)

- צור עמודה חדשה בשם high_value_order המכילה True אם ה-total_amount באותה שורה גבוה מהממוצע של העמודה total_amount, ו-False במידה ולא.

- מיון את הטבלה מה-total_amount הגבוה לנמוך.

שלב 6 - (יצירת עמודה חדשה בסוף הטבלה)

צור עמודה חדשה הכוללת את ממוצע הדירוג rating בעבור כל מדינה country.

שלב 7 - (מחיקת חלק מהשורות - פילטר)

סן את ההזמנות המופיעות בטבלה, כך שהרשומות הנותרות יעמדו בתנאים הבאים:
הערך של total_amount גבוה מ-1000 וגם Rating גבוה מ-4.5

שלב 8 - (יצירת עמודה חדשה בסוף הטבלה)

צור עמודה חדשה בשם delivery_status המכילה את הערכים הבאים:
אם shipping_days גדול מ-7 יהיה הערך "delayed" ובמידה ולא, יהיה הערך "on time".

שלב 9 - (שמירת הטבלה כקובץ)

שמרו את ה DF שיצרתם כקובץ CSV בשם clean_orders_[ID_NUMBER].csv.

בהצלחה!