

# ניהול מידע מבוזר - פרויקט חלק א'

## חלק 1

### שאלה 1.1

בשביל לסנן את העמודות שעומדות בתנאים כדרוש, בחרנו להוסיף את העמודות הבאות:

- בטבלה Demographic data נוסיף את עמודת age\_gap - הפרש גילאים בין המבוגר הראשון למבוגר השני. נשתמש בעמודה זו בשביל התנאי השלישי, שכן נצטרך לבדוק בו האם הפרש הגילאים הוא גדול/שווה ל6, לכן נוכל פשוט לבצע filter על העמודה החדשה.
- בטבלה Daily program data נוסיף את עמודת weekday – היום בשבוע (בפורמט string מקוצר) שבו התוכנית עלתה לאוויר. נשתמש בעמודה זו בשביל התנאי הרביעי, שכן נצטרך לסנן רק את הרשומות שהתקיימו ביום שיש.
- בטבלה Daily program data נוסיף את עמודת monthday – היום מתוך התאריך (1-31) שבו התוכנית עלתה לאוויר. נשתמש בעמודה זו בשביל התנאי הרביעי, שכן נצטרך לסנן רק את הרשומות שהתקיימו ביום ה-13.
- בטבלה Demographic data נוסיף את עמודת numeric\_income – נמיר את האותיות A-D שהופיעו בעמודה הזו למספרים נומריים כפי שהגודר בתרגיל למספרים 10-13. נצטרך את כל ההכנסות בתור מספרים על מנת לחשב ממוצע הכנסה, שכן בתנאי החמישי נצטרך לסנן רק את המשפחות להן הכנסה מתחת לממוצע.
- בטבלה Daily program data נוסיף את עמודת genre\_array – המטרה היא להפריד בין המחרוזת שהופרדה ב", למערך של ז'אנרים, כדי שנוכל לבדוק בצורה נוחה את תנאי 6.
- בעת עיבוד הנתונים והוספת הטבלה רצינו לחלץ את הרשומות "הרלוונטיות" כלומר, רשומות אשר לפחות אחד מהז'אנרים שלהן שייך לרשימה הנתונה. בעת ביצוע עיבוד זה, נעזרנו בצ'אט GPT.

## שאלה 1.2 a

נציג כעת יתרונות וחסרונות של השיטה שהוצעה בשאלה (כלומר איחוד 4 הטבלאות לטבלה אחת):

יתרונות :

- פשטות – מנהלים Data Frame יחיד המאפשר פשטות בעבודה עם תנאים מורכבים המשלבים שדות ממספר טבלאות שונות.
- בדיקת התנאים על הרשומות היא נוחה יותר, שכן לכל רשומה ניתן לבדוק "בבת אחת", את כל 7 התנאים.

חסרונות :

- בזבוז זיכרון ומשאבים – הטבלה המאוחדת תהיה מאוד גדולה ותכיל הרבה מידע, ייתכן שהרבה מהמידע יהיה מיותר.
- כפילויות מיותרות – לאחר איחוד כל 4 הטבלאות, יישמר הרבה מידע כפול (לדוגמא עבור כל שידור, יישמרו רשומות כמספר מופעי הצפייה, והמידע על השידור יופיע המון פעמים).

