מטלה מסכמת: קורס כרייה וניתוח נתונים מתקדם + קורס למידת מכונה

במטלה זו נבנה אפליקציה לחיזוי מחירים מבוקשים לדירות.

את המטלה יש לבצע בזוגות.

**שלב 1**: במטלה 4 כתבנו crawler שאסף דאטה על דירות למכירה ומחירים מבוקשים מאתר מדל"ן באמצעות scraping

הדאטה כולל:

* type – סוג הנכס – דירה/בית פרטי/קוטג'
* room\_number – מספר חדרים
* floor\_out\_of - קומה מתוך כמה קומות
* Area – שטח הדירה
* num\_of\_images
* Street – שם הרחוב
* number\_in\_street – מספר ברחוב
* city\_area – האיזור בעיר
* hasElevator – האם יש מעלית
* hasParking – האם יש חניה
* hasBars – יש סורגים
* hasStorage – יש מחסן
* condition – מצב הנכס (משופץ/דורש שיפוץ/שמור... (
* hasAirCondition = יש מזגן
* hasBalcony = יש מרפסת
* hasMamad = יש ממד
* handicapFriendly – נגיש לנכים
* entranceDate – תאריך כניסה
* furniture – כולל ריהוט (חלקי/כן/לא)
* publishedDays – מספר ימים שההודעה מפורסמת
* description – תיאור הנכס
* price – מחיר הנכס (המשתנה התלוי)

שימו לב: יתכן שלחלק מהנכסים רק חלק מהשדות יהיו מלאים.

**שלב 2:** על הדאטה מהשלב הראשון יש לבצע הכנת נתונים:

* יש למחוק נכסים שבהם אין מחיר
* יש לוודא שהמחיר הוא מטיפוס מספרי (להשאיר רק את הערך המספרי)
* יש לוודא שהשטח הוא מטיפוס מספרי (להשאיר רק את הערך המספרי)
* יש להוריד מהטקסטים בכתובת, איזור בעיר, תאור הנכס וכו' פסיקים או סימני פיסוק מיותרים
* יש להוסיף עמודה של floor - קומה (מתוך עמודת floor\_out\_of)
* יש להוסיף עמודת – total\_floors – סך כל הקומות בבניין
* יש ליצור עמודת entrance\_date עדכנית שהיא קטגוריאלית:
  + less\_than\_6 months / months\_6\_12/ above\_year / flexible / not\_defined
* יש לייצג את כל השדות הבוליאנים (hasBalcony, hasMamad וכו') כאפסים ואחדים.

**שלב 3**: יש לבנות מודל חיזוי מסוג elastic net של מחיר הנכס על פי העמודות שיצרתם.

ניתן לבחור באילו עמודות רוצים להשתמש ולבצע feature engineering כדי להשיג את המודל המדויק ביותר.

יש לבדוק את ביצועי המודל תוך שימוש ב – 10-fold-cross-validation

יש לדווח על ביצועי המודל באמצעות מדדי ביצועים רלוונטיים

**שימו לב: 90% מהציון על סעיף זה, שמהווה את הציון בקורס של ד"ר חן חג'ג', יינתן על כתיבת הקוד ובניית המודל שכתבתם, 10% נוספים ינתנו לפי ביצועי המודל (יכולתו לחזות את המחירים על נכסי TEST שהשארנו בצד)**. כלומר המודל שישיג את הביצועים הכי טובים (MSE), יקבל 10 נקודות, המודל שישיג את הביצועים הכי פחות טובים יקבל 0 נקודות.

**שלב 4**: יש ליצור שירות Flask ודף html שיאפשר להזין את הנתונים הרלוונטיים למודל ולהריץ את המודל , כך שלבסוף יודפס על הדף המחיר החזוי

**שלב 5**:

את כל הקוד יש להעלות ל – repository ייעודי ב – Github

הקוד יכלול:

* קובץ html בשם **index.html** שמכיל טופס למילוי פרטי נכס וכפתור "חיזוי מחיר"
* קובץ פייתון בשם **madlan\_data\_prep.py**. הקובץ יכיל:
  + **פונקציה בשם prepare\_data().** הפונקציה תקבל dataframe בדיוק באותו מבנה של נתוני האימון (הנתונים ממדלן) ותבצע עיבוד מוקדם על השדות ובחירה של שדות עליהן יבוצע האימון. הפלט של הפונקציה יהיה DataFrame חדש של הנתונים המעובדים הכוללים רק את השדות הרלוונטיים. הפונקציה תשמש להכנת הנתונים לאימון המודל וגם להכנת נתוני ה TEST לפני הזנתם למודל.
* קובץ בשם **model\_training.py** , שבו:
  + מפעילים על ה – Train set שהעלינו לכם את הפונקציה prepare\_data(), שמחזירה data מעובד
  + מבצעים את אימון המודל
  + שומרים את המודל כקובץ PKL.
* קובץ PKL בשם **trained\_model.pkl** , שיכיל את המודל המאומן
* קובץ **api.py** שיכיל את אפליקציית ה – Flask
  + קובץ **requirements.txt** אותו תצרו באמצעות הפקודה pip freeze בסביבה הוירטואלית שבה יצרתם את האפליקציה.