**סקרייפינג באמצעות Playwright**

במסגרת המשימה, ביצענו Web Scraping משני אתרים לחיפוש טיסות: Momondo ו-Kayak, תוך שימוש בספריית Playwright בשפת Python. המטרה הייתה לאסוף נתוני טיסות בהתאם לקריטריונים של TTT (זמן בין ההזמנה לטיסה) ו-LOS (משך השהייה ביעד) עבור מגוון מסלולים.

טכנולוגיות וכלים

* Playwright – ספרייה לניהול דפדפן אוטומטי.
* asyncio – ספרייה לניהול ריצות אסינכרוניות.
* re – לטיפול במחרוזות באמצעות ביטויים רגולריים.
* datetime – לעבודה עם תאריכים.
* csv / json – לשמירת הנתונים.
* requests – לשליחת התראות (במקרה של כישלון סקרייפינג).

מבנה הקוד

1. טיפול בדפדפן ובטעינת הדף

* יצירת דפדפן חדש ועמוד חדש בעזרת async\_playwright.
* שימוש ב-headless=False למניעת זיהוי כמפעיל בוטים.
* הגדרת User-Agent רנדומלי כדי לדמות משתמשים שונים.
* טיפול בקפצ׳ות במידה והופיעו (ניקוי עוגיות וטעינת דף מחדש).

2. דגימת נתוני טיסות

* יצירת רשימה של מסלולים אפשריים (routes או permutations של ערים).
* חיפוש טיסות לפי המסלול, תאריך המראה ותאריך חזרה.
* לחיצה על כפתורים דינמיים למעבר בין חודשים בלוח השנה.
* חיקוי אינטראקציה אנושית עם הדף (גלילה, תנועת עכבר).
* טעינה הדרגתית של טיסות כדי לוודא שנאספו מספיק תוצאות.
* שליפת נתוני טיסה מרכזיים, כולל:
* זמני המראה ונחיתה
* שדות תעופה
* סוג הטיסה (ישירה / קונקשן)
* חברות תעופה
* משך עצירות
* מחירי טיסות
* סוג כבודה
* תאריך Snapshot של הנתונים

3. שמירת הנתונים

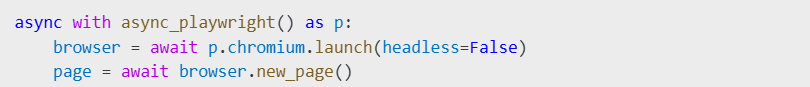
* שמירת המידע בקובץ CSV.
* ניטור התקדמות ושמירת שאילתות שכבר הושלמו בקובץ JSON למניעת חיפושים כפולים.
* התראות Telegram במקרה של כישלון הסקרייפינג.

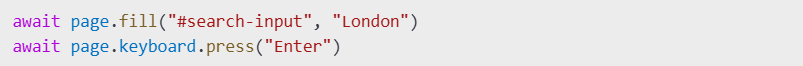
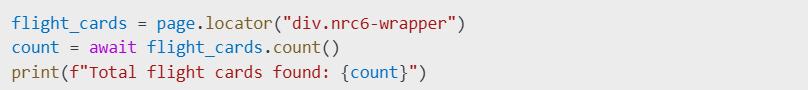
**התחביר של Playwright וכיצד מתבצע הסקרייפינג**

Playwright היא ספרייה לניהול דפדפנים אוטומטיים, וניתן להשתמש בה להפעלת דפדפן וניווט בדפים באמצעות קוד Python.

שלבי הסקרייפינג בפועל:

1. פתיחת דפדפן ועמוד חדש:



1. ניווט לעמוד המבוקש:
2. מילוי שדות וחיפוש:
3. שליפת נתונים מהעמוד:
4. אינטראקציה עם עמוד דינמי (העמסה הדרגתית של תוצאות):

תמונה שמכילה טקסט, גופן, צילום מסך

תוכן שנוצר על-ידי בינה מלאכותית עשוי להיות שגוי.

1. תמונה שמכילה טקסט, גופן, צילום מסך, קו

   תוכן שנוצר על-ידי בינה מלאכותית עשוי להיות שגוי.שמירת הנתונים לקובץ CSV:

התייחסות למשימה שקיבלנו

בקובץ המשימה שהתקבל נדרשו לבצע חיפוש טיסות תוך עמידה בקריטריונים מוגדרים מראש:

* ביצוע חיפושים לכל הקומבינציות של TTT בין 1 ל-30 ימים ו-LOS בין 1 ל-5 ימים.
* דגימת לפחות 100 טיסות לכל חיפוש והבטחת טעינת תוצאות מספקת.
* תיעוד הנתונים כולל שעות טיסה, שדות תעופה, חברות תעופה, עצירות, מחירים וזמן Snapshot.
* שמירת המידע בקובץ CSV להמשך ניתוח.

במהלך הביצוע נעשה שימוש במנגנוני הגנה מפני בוטים, כולל ניקוי עוגיות, שינוי User-Agent, חיקוי התנהגות אנושית ופתרון בעיות טעינה. הנתונים נשמרו וניטור ההתקדמות בוצע בקובץ JSON כדי למנוע כפילויות ולשפר את יעילות החיפושים.

מסקנות

* שימוש ב-Playwright אפשר לנו לגשת ולנתח דפים דינמיים בצורה אמינה תוך שמירה על אסינכרוניות לביצועים מהירים.
* נדרש טיפול זהיר בזיהוי בוטים, במיוחד באתרים פופולריים עם מנגנוני הגנה מתקדמים.
* ניהול נתונים חכם (JSON לשאילתות שהושלמו, CSV לשמירת תוצאות) עזר למנוע חיפושים כפולים ולשמור על ארגון המידע.
* שימוש נכון באינטראקציה אנושית (עכבר, גלילה, השהיות אקראיות) שיפר את ההצלחה של הסקרייפינג ומנע חסימות.