

**תרגיל בית – דאטה סיינס/ genAI**  
**חברת ONE**

## 1. רקע ומטרה

בתהליך ניתוח מסמכי מכרז, אחת הבעיות המרכזיות היא זיהוי מהיר ומדויק של מידע רלוונטי מתוך עשרות/מאות עמודים. תהליך ידני כזה לוקח בין שעות לבין שבועות כתלות במורכבויות שונות (אורך המכרז, פרמטרים מחולצים ועוד..)

כדי לעזור להם לענות על זה אוטומטית רצינו לפתור זזאת על ידי מודלי שפה. במקום לשלוח את כל המסמך למודל שפה (בפעם אחת) – פעולה שמובילה לעלויות גבוהות, ירידה בדיוק ובזבזז משאבים – אנחנו מפרקים את המסמך לעמודים (או מקטעים), מתייגים את התוכן של כל עמוד מראש, ושולחים רק את החלקים הרלוונטיים עבור כל פרמטר.

מדובר בגישה מקבילה לתהליך (Retrieval-Augmented Generation) RAG, אך במקום לבצע חיפוש וקטורי על דאטה חתוך מראש, כאן התהליך כולל חיתוך, תיוג והתאמה בשלב הריצה (ולא האימון/הכנת הדאטה בייס) – על בסיס מטא-דאטה, כללים וניתוח מבני של המסמך.

מטרת התרגיל היא:

- לעבור על המסמך ולתייג אותו לפי עמודים/פרמטרים רצויים לחילוץ
- לזהות נושאים ותכנים לפי רשימת פרמטרים נתונה
- ולהתאים בין כל פרמטר לעמודים הרלוונטיים בלבד – כך שהמודל יעבוד רק על התוכן הנחוץ

## 2. קלט התרגיל

לצורך ביצוע התרגיל יסופקו שני קבצים:

1. קובץ PDF של מכרז אמיתי לדוגמה – tender\_sample.pdf : המסמך כולל מידע הרבה מידע על דברים שונים כולל על המזמין, תיאור השירות, תנאי סף, מועדים, קריטריונים להערכה, ערבות ועוד.
2. קובץ JSON ובו רשימת פרמטרים שעליך לאתר במסמך. המטרה היא לזהות אילו עמודים במסמך רלוונטיים עבור כל פרמטר. אם פרמטר לא מופיע במסמך – יש לציין במפורש "לא נמצא". יש פרמטרים/ים שאינם קיימים במסמך שנכללו במכוון, כדי לבדוק טיפול במקרים כאלה.

### 3. משימות

עליך לפתח רכיב תוכנה שמקבל כקלט את קובץ המכרז ואת רשימת הפרמטרים, ומחזיר עבור כל פרמטר את רשימת העמודים במסמך שבהם סביר להניח שנמצא המידע הרלוונטי או שכדאי לשלוח אליהם את הפרמטר ובאמצעות זה את ערך הפרמטר עם תוספות לפלט שנפרט בהמשך.

אנחנו מאוד מעריכים חשיבה מודלרית ופירוק המשימה לפונקציות ולא רק סקריפט אחד ארוך!!

#### שלבים:

##### חיתוך המסמך לעמודים

קריאה וחיתוך לעמודים של המסמך - כל עמוד יטופל כיחידה נפרדת לניתוח.

##### תיוג כל עמוד לפי תוכנו

- יש לזהות עבור כל עמוד אילו פרמטרים (אם בכלל) מופיעים בו בפועל.
- בנוסף, ניתן (ואף רצוי) להוסיף תיוג אחד (מהבאים) נוסף לפי שיקולך:
  - נושא משוער של העמוד לפי מילות מפתח
  - מבנה טקסטואלי ייחודי (כותרות, רשימות, טפסים)
  - תיוג אחר שיכול לסייע בהתאמת עמודים לפרמטרים
- מומלץ להשתמש במודל שפה כלשהו, או רעיון אחר אם תרצו.

##### התאמה בין פרמטרים לעמודים

לאחר מכן, עבור כל פרמטר, יש למפות את העמודים הרלוונטיים במסמך. למשל:

```
my_parameter_pages=[2,3,7,8,40]
```

אם פרמטר לא נמצא כלל - יש לציין זאת במפורש בפלט כ"לא נמצא"

יש לוודא איתור איכותי גם במקרים של: מילים נרדפות, הטיות לשון, ניסוחים עקיפים, תרגום או שימוש מונחים שונים למסר דומה

הפלט צריך להתאים לפורמט הצנרת הקיימת (בהמשך תופיע דוגמה).

##### בניית פרומפט בסיסי לכל פרמטר

- כתוב פרומפט עבור כל פרמטר, שמטרתו להנחות את מודל השפה לחלץ את המידע מהעמודים שנבחרו.

- רצוי לבנות את הפרומפטים בצורה מודולרית שתאפשר שימוש חוזר על פני מספר פרמטרים דומים (חשוב!!).
- כדי לחסוך לכם בזמן – חשוב להבהיר שהפרומפט לא חייב להיות הכי מדויק ועמוק שיש. בעיקר חשוב לנו מבנה הקוד ותיוג הדאטה.

### שליחת הפרומפט למודל ושמירת התגובה

- הפעל את הפרומפטים מול מודל שפה שוב – הפעם כדי לקבל את הערך המחולץ (עדיף דרך api).
- בעצם לכל פרמטר לשלוח את הפרומפט ואת העמודים הרלוונטיים לו
- לשם פשטות וחיסכון בזמן – אפשר להתעלם מטיפול בטעויות ובעיות בחזרה ב api
- שמור את הפלטים במבנה קבוע:
  - ראו דוגמא בהמשך
  - אפשר בתוך קובץ או שאם הקידוד בעברית לוקח זמן (מבחינת לשמור בוורד או פידיאף) אפשר להדפיס משתנה בקונסול/דיבאגר/notebook

### רשימת הפרמטרים עם הסברים

להלן 7 פרמטרים שנבחרו לתרגיל. עליכם לזהות עבור כל אחד מהם את העמודים הרלוונטיים במסמך. אם פרמטר אינו מופיע – יש לציין זאת במפורש בפלט כ"לא נמצא". בבקשה לא להעתיק מפה אלא לקרוא את הקובץ ג'ייסון ולקחת את הרשימה משם.

שם הפרמטר (עברית)	שם בפרויקט (אנגלית)	הסבר
שם המזמין	client_name	שם הגוף שפרסם את המכרז (למשל: רשות, תאגיד, עירייה)
שם המכרז	tender_name	שם מלא של המכרז, כולל מספר וסוג (פומבי, דו-שלבי וכו)
תנאי סף	threshold_conditions	דרישות חובה להשתתפות – לדוגמה: ניסיון, עמידה בחוקים, רישוי.
תקופת ההתקשרות	contract_period	כמה זמן תימשך ההתקשרות, והאם קיימות אופציות להארכה.
שיטת ההערכה (מפ"ל)	evaluation_method	כיצד נשקלל את ההצעות – מחיר מול איכות, קריטריונים לניקוד.
ערבות מכרז	bid_guarantee	סכום הערבות, סוג הערבות, תנאים לפירעון, תוקף.
הוגה הרעיון	idea_author	פרמטר לא רלוונטי שנוסף בכוונה – אין לצפות שימצא במסמך. יש לציין "לא נמצא". לא לרשום זאת למודל, אלא אמורים לשלוח פרומפט ודפים כרגיל

## 5. מבנה הפלט המצופה

עליך להחזיר עבור כל פרמטר את המידע הבא:

- שם הפרמטר
- ערך הפרמטר - הערך המרכזי שחולץ מהמסמך
- פרטים - פירוט נוסף, ניסוח מורחב או פרשנות
- מקור - העמוד או המיקום ממנו נלקח המידע
- ציון - ציון כולל (1-5) לאיכות החילוף - לבקש מהמודל שיעשה זאת -

## דוגמה לפורמט הפלט: (JSON)

```
{
  "client_name": {
    "answer": "רשות המיס והביוב",
    "details": "תאגיד עירוני שהוקם לפי חוק תאגידי מים וביוב",
    "source": "עמוד 2, פסקה ראשונה",
    "score": 5
  },
  "tender_name": {
    "answer": "מכרז מס' 28/2024 - שירותי ניטור תחזוקה",
    "details": "מכרז DBA פומבי דו-שלבי לשירותי",
    "source": "עמוד 1, כותרת",
    "score": 4
  },
  "idea_author": {
    "answer": "",
    "details": "",
    "source": "לא נמצא",
    "score": 0
  }
}
```

## הנחיות:

- הפלט חייב להתבסס על **תוכן אמיתי מתוך המסמך**.
- אם פרמטר לא מופיע:
  - – השאר את answer ו־ details ריקים
  - Source: לא נמצא
  - ציון – לפי רמת הוודאות של המודל שהוא לא מצא באמת
- בספריית זיפ ששלחנו (מבחן בית – one) יש כמה קבצים:
  - המסמך הזה
  - פרמטרים לחילוף
  - מכרז ללימוד עם תשובות

- תשובות של מכרז
- מכרז נוסף לבדיקת הקוד (סוג של test)
- מודלי שפה לשימוש: אתם יכולים להוריד מודל פתוח/open source או לחילופין למי שיש ורוצה יכול להשתמש ב api.
- הדגש הוא על מבנה הפיתרון ולאוו דווקא על הדיוק ושימוש במודלים יקרים/נגישים/חכמים יותר.
- מודולריות זה חשוב – בנו מספר פונקציות שקל להחליף להם פרמטרים.
- חשוב מאוד!!
- למען הפשטות שלחנו מסמכים קצרים. במצב כזה אפשר גם בקלות לשלוח כל פעם את כל המסמך למודל שפה ולקבל תוצאות – זו לא כוונת המשורר – עשינו זאת כדי להקל – ראו בשליחת מינימום עמודים לכל פרמטר כאחת המטרות. במציאות מכרז גם יכול להיות 500 עמודים

## דרישות טכניות:

שפת תכנות: פייתון

ספריות/מודולים מותרים: כל מודל

מודלי שפה: אתם יכולים להוריד מודל פתוח/open source או לחילופין למי שיש ורוצה יכול להשתמש ב api. הדגש הוא על מבנה הפיתרון ולאוו דווקא על הדיוק ושימוש במודלים יקרים/נגישים/חכמים יותר.

דרישות קוד:

- יש להגיש סקריפט אחד או מחברת Jupyter אחת שמריץ את כל התהליך מקצה לקצה
- הקוד צריך לכלול:
  - קריאת המסמך
  - ניתוח לפי עמודים
  - תיוג
  - התאמת פרמטרים לעמודים
  - יצירת פלט לפי המבנה שנקבע

או לחילופין סרטון אמיתי ולא ערוך של הכל רץ אצליכם עם הסברים על הפונקציות השונות

- חשוב מאוד!!!
- תיהיה עדיפות (כמו שתואר כבר) לחשיבה מודלרית עם פונקציות. משמע איזו פונקציית main שקוראת לפונקציות קטנות/בינוניות. למשל:

```
def main():
    print("starting... ")
    init_time = time.time()

    load_data(file_name=file_name,...)
    ○ more functions here
```