**אנליזה של ביג דאטה - תרגיל גדול:**

1. **קישור למקור נתונים:**

<https://www.kaggle.com/datasets/PROPPG-PPG/hourly-weather-surface-brazil-southeast-region/data>

תיאור המאגר: המאגר שבחרנו מציג נתוני מזג אוויר לפי שעה מ-623 תחנות מזג אוויר אינמט בברזיל, התמקדנו ב-central west של ברזיל.

1. **תיאור גודלו של המאגר:**

* שורות: 11,427,120 שורות.
* עמודות: 27
* נפח המידע: 1.9GB

1. **תוכן כללי של המאגר:**

* תאריך (YYYY-MM-DD)
* זמן (HH:00)
* כמות המשקעים במילימטרים (שעה אחרונה)
* לחץ אטמוספרי ברמת התחנה (mb)
* לחץ אוויר מקסימלי בשעה האחרונה (MB)
* לחץ אוויר מינימלי בשעה האחרונה (MB)
* קרינת שמש (KJ/m2)
* טמפרטורת אוויר (מיידית) (°c)
* טמפרטורת נקודת הטל (מיידית) (°c)
* טמפרטורה מקסימלית בשעה האחרונה (°c)
* טמפרטורה מינימלית בשעה האחרונה (°c)
* טמפרטורת נקודת הטל המרבית בשעה האחרונה (°c)
* טמפרטורת נקודת הטל המינימלית בשעה האחרונה (°c)
* טמפרטורת לחות יחסית מקסימלית בשעה האחרונה (%)
* טמפרטורת לחות יחסית מינימלית בשעה האחרונה (%)
* לחות יחסית (% מיידי)
* כיוון הרוח (רדיוס מעלות (0-360)).

נתונים חשובים:

* מקור: INMET (מכון המטאורולוגי הלאומי - ברזיל).
* קטגוריה: מזג אוויר.
* פורמט: CSV.

1. **תיאור הקובץ db\_file.duckdb:**

הקובץ db\_file.duckdb הוא מסד נתונים גדול שנוצר בפורמט DuckDB המיועד לטיפול בכמויות נתונים גדולות בצורה יעילה. הוא מכיל:

* נתוני מזג אוויר של 623 תחנות באזור Central West Brazil הכוללים מעל 11 מיליון שורות.
* עמודות עם מדדים שונים כמו:
  + טמפרטורה .(Temperature)
  + משקעים .(Precipitation)
  + מהירות וכיוון הרוח .(Wind Speed & Direction)
  + קרינת שמש (Solar Radiation).
  + לחות .(Humidity)
  + ועוד.

מה מטרתו?

* אחסון מרכזי: לשמש מאגר נתונים ראשי שמרכז את כל הנתונים הגולמיים בפורמט דחוס ויעיל.
* עיבוד נתונים מהיר: ביצוע שאילתות SQL מורכבות על כמויות נתונים גדולות, תוך שמירה על ביצועים גבוהים.
* הפקת טבלאות קטנות: יצירת טבלאות מסוננות וסיכומיות שהועברו למסד נתונים קטן יותר (SQLite) לצורך שימוש בדאשבורד.

**תהליך העלאת הדאטה גדול ל-DuckDB:**

בשלבים הבאים נתאר כיצד חיברנו את בסיס הנתונים DuckDB יצרנו טבלה, טענו לתוכה נתונים מקובץ CSVואימתנו את התהליך.

1. חיבור ל-DuckDB:

השלב הראשון הוא יצירת חיבור לבסיס הנתוניםDuckDB באמצעות הפונקציה duckdb.connect(). במקרה זה, חיברנו את קובץ מסד הנתונים db\_file.duckdb שבו ישמרו הנתונים.

הפקודה שהרצנו:

import duckdb

con = duckdb.connect('db\_file.duckdb')

------

1. יצירת טבלה וטעינת הנתונים:

בשלב זה, השתמשנו בפקודתread\_csv\_auto המאפשרת ל DuckDBלטעון את הנתונים ישירות מקובץ CSV באופן אוטומטי, ללא צורך בהגדרת מבנה הטבלה מראש.

הפקודה CREATE TABLE ... AS SELECT \* FROM read\_csv\_auto(...) יוצרת את הטבלה central\_west\_data שאליה נטענו כל הנתונים מקובץ הCSV.

הפקודה שהרצנו:

con.execute(f"""

CREATE TABLE central\_west\_data AS

SELECT \* FROM read\_csv\_auto('{csv\_file\_name}');

""")

------

1. אימות מבנה הטבלה:

כדי לוודא שהנתונים נטענו כראוי, הפעלנו את הפקודה PRAGMA table\_info המציגה את מבנה הטבלה ואת סוגי הנתונים בכל עמודה. כך יכולנו לבדוק שהעמודות והמבנה תואמים לציפיות.

הפקודה שהרצנו:

table\_info = con.execute("PRAGMA table\_info('central\_west\_data');").fetchdf()

print("Columns in the table:")

print(table\_info)

------

1. ספירת שורות הטבלה:

לאחר טעינת הנתונים בדקנו כמה שורות נטענו בפועל לטבלה central\_west\_data באמצעות הפקודה SELECT COUNT(\*). הפלט הצביע על כך שנטענו בהצלחה 11,427,120 שורות.

הפקודה שהרצנו:

row\_count = con.execute("SELECT COUNT(\*) AS row\_count FROM central\_west\_data;").fetchone()[0]

print(f"Row count in central\_west\_data: {row\_count}")

------

1. סגירת החיבור:

לאחר סיום התהליך, סגרנו את החיבור ל-DuckDB כדי לשמור על ביצועים מיטביים ולשחרר משאבים.

הפקודה שהרצנו:

con.close()

1. **+ (7) הסיפור שאנחנו רוצות לספר:**

בפרויקט זה התעסקנו בנתונים על מזג אוויר מהמרכז המערבי של ברזיל, תוך התמקדות ב-623 תחנות מזג אוויר. המטרה שלנו היא להבין את דפוסי מזג האוויר באזור, ובמיוחד את השפעת המשתנים השונים (כמו טמפרטורה ומשקעים) על האקלים המקומי.

השאלות ששאלנו על הדאטה, מה אנחנו יכולות ללמוד מהן והתובנות שהסקנו:

שאלה 1:

**מהי הטמפרטורה הממוצעת לאורך כל תקופה?**

* + **השאילתה:**

SELECT DISTINCT

strftime('%Y-%m', Data) AS YearMonth,

AVG("TEMPERATURA DO AR - BULBO SECO, HORARIA (\u00b0C)")

OVER(PARTITION BY strftime('%Y-%m', Data)) AS AvgTemp

FROM central\_west\_data

ORDER BY YearMonth;

* + **מה השאלה שנענית באמצעות שאילתה זו?**

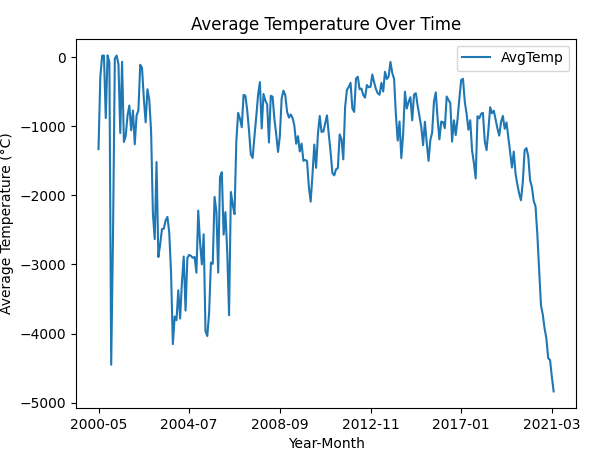
השאילתה מחשבת את הטמפרטורה הממוצעת עבור כל חודש ושנה מתוך הנתונים בטבלה central\_west\_data. היא מאפשרת לקבוע את ממוצע הטמפרטורה לכל חודש מבלי לשמור נתונים פרטניים של מדידות בודדות.

* + **מה ניתן ללמוד מהשאילתה?**

השאילתה מספקת מידע על מגמות טמפרטורה חודשיות לאורך התקופה הנתונה. באמצעותה ניתן לזהות חודשים חמים או קרים יחסית, ולהסיק מסקנות על שינויים עונתיים או מגמות אקלימיות אזוריות.

* + **איך השאילתה מצמצמת את הנתונים?**

במקום להחזיק כל קריאה בודדת של הטמפרטורה, השאילתה מייצרת ערך חודשי יחיד של ממוצע טמפרטורה, מה שמקטין את כמות הנתונים המאוחסנים או המוצגים.

* + **המחשה בצורה ויזואלית:**

שאלה 2:

**מהי כמות המשקעים הכוללת בכל תחנה ובחודש?**

* **השאילתה:**

SELECT

strftime('%Y-%m', Data) AS YearMonth,

station,

SUM("PRECIPITA\u00c7\u00c3O TOTAL, HOR\u00c1RIO (mm)") AS TotalPrecipitation

FROM central\_west\_data

GROUP BY CUBE (YearMonth, station)

HAVING YearMonth IS NOT NULL AND station IS NOT NULL

ORDER BY YearMonth, station;

* **מה השאלה שנענית באמצעות שאילתה זו?**  
  השאילתה סוכמת את כמות המשקעים הכוללת לפי תחנה ומועד, ומספקת את הסכום הכולל של המשקעים לכל חודש בתחנות השונות.
* **מה ניתן ללמוד מהשאילתה?**  
  השאילתה מאפשרת להבין את התפלגות המשקעים בכל תחנה ובכל חודש. בעזרתה ניתן לזהות אזורים שבהם נמדדו כמויות משקעים גבוהות או נמוכות, ולעקוב אחרי מגמות גשם עונתיות.
* **איך השאילתה מצמצמת את הנתונים?**  
  השאילתה מבצעת סיכום (SUM) של המשקעים עבור כל חודש ותחנה, ומספקת נתון מרוכז שמקטין את כמות הנתונים והמידע המוצג.
* **המחשה בצורה ויזואלית:**

תמונה שמכילה טקסט, צילום מסך, קו, עלילה

התיאור נוצר באופן אוטומטי

שאלה 3:

**כמה מדידות יש בכל תחנה?**

* **השאילתה:**

SELECT

station,

COUNT(\*) AS DataCount

FROM central\_west\_data

GROUP BY station

ORDER BY DataCount DESC

* **מה השאלה שנענית באמצעות שאילתה זו?**  
  השאילתה סופרת את מספר השורות (מדידות) עבור כל תחנה ומציגה את מספר השורות הכולל שקשור לכל תחנה בטבלה central\_west\_data.
* **מה ניתן ללמוד מהשאילתה?**  
  השאילתה מאפשרת להבין כמה נתונים קיימים עבור כל תחנה, ובכך ניתן לזהות תחנות עם יותר או פחות מדידות, מה שעשוי להעיד על דיוק ותחום הנתונים לכל תחנה.
* **איך השאילתה מצמצמת את הנתונים?**  
  השאילתה מציגה את מספר השורות בלבד לכל תחנה, במקום להציג את כל הנתונים, דבר שמקטין את כמות המידע המוצג ומאפשר ניתוח מהיר.
* **המחשה בצורה ויזואלית:**

תמונה שמכילה טקסט, צילום מסך, עלילה, תרשים

התיאור נוצר באופן אוטומטי

שאלה 4:

**מהן הטמפרטורות המקסימליות והמינימליות ומהירות הרוח המקסימלית בכל תחנה?**

* **השאילתה:**

SELECT

station,

MAX("TEMPERATURA DO AR - BULBO SECO, HORARIA (\u00b0C)") AS MaxTemp,

MIN("TEMPERATURA DO AR - BULBO SECO, HORARIA (\u00b0C)") AS MinTemp,

MAX("VENTO, VELOCIDADE HORARIA (m/s)") AS MaxWindSpeed

FROM central\_west\_data

GROUP BY station

ORDER BY MaxTemp DESC

* **מה השאלה שנענית באמצעות שאילתה זו?**  
  השאילתה מחשבת את הטמפרטורה המקסימלית והמינימלית שנמדדו בכל תחנה, יחד עם מהירות הרוח המקסימלית בכל תחנה.
* **מה ניתן ללמוד מהשאילתה?**  
  השאילתה מאפשרת לזהות את תנאי מזג האוויר הקיצוניים בכל תחנה, כגון הטמפרטורות החמות ביותר והקרות ביותר, ומהירות הרוח הגבוהה ביותר.
* **איך השאילתה מצמצמת את הנתונים?**  
  השאילתה מחזירה רק את הערכים הקיצוניים ביותר עבור כל תחנה (מקסימום ומינימום), כך שהיא מצמצמת את כמות הנתונים על ידי הצגת מידע ממוזג, ולא כל קריאה נפרדת.
* **המחשה בצורה ויזואלית:**

תמונה שמכילה טקסט, צילום מסך, מקביל, קו

התיאור נוצר באופן אוטומטי

* 1. **שאילתות שבוחרות שורות לדוגמא מכל טבלה:**

1. SELECT \* FROM avg\_temp\_by\_month LIMIT 5;
2. SELECT \* FROM total\_precipitation\_by\_month LIMIT 5;
3. SELECT \* FROM total\_precipitation\_by\_station LIMIT 5;
4. SELECT \* FROM avg\_wind\_speed\_by\_station LIMIT 5;
   1. **תהליך העלאת הדאטה הקטן ל- SQLITE:**

1. יצירת טבלאות קטנות:

* head(500):   
  פקודה זו משמשת כדי לבחור את 500 השורות הראשונות מכל DataFrame. עבור כל אחד מה-DataFrame-ים (avg\_temp\_over\_time, monthly\_precipitation, data\_count\_by\_station, ו-extreme\_conditions), פקודת head(500) שומרת את 500 השורות הראשונות ומשמרת אותן במשתנים חדשים.
* table\_mappings = {}:  
  מבנה נתונים מסוג מילון שבו נשמרים שמות הטבלאות כמפתחות וה-DataFrame-ים כערכים. זה מאפשר גישה קלה לכל טבלה מאוחסנת.

2. שמירת הנתונים בבסיס נתונים SQLite:

* sqlite3.connect('central\_west\_data.db'):  
  פקודה זו יוצרת חיבור למסד נתונים SQLite בשם central\_west\_data.db. אם מסד הנתונים לא קיים, הוא ייווצר אוטומטית.
* df.to\_sql(table\_name, sqlite\_conn, if\_exists='replace', index=False):  
  פקודה זו משמשת לשמירת ה-DataFrame כטבלה בבסיס הנתונים.
* table\_name: שם הטבלה שתשמר בבסיס הנתונים.
* sqlite\_conn: חיבור למסד הנתונים.
* if\_exists='replace': אם טבלה בשם זה כבר קיימת, היא תימחק ותוחלף בטבלה החדשה.
* index=False: לא לשמור את עמודת האינדקס של ה-DataFrame בטבלה.
* print(f"Table {table\_name} saved to SQLite."):

לאחר שמירת כל טבלה, מודפסת הודעה שמאשרת את השמירה.

3. אימות הנתונים בבסיס הנתונים:

* sqlite\_conn.execute(f"SELECT COUNT(\*) FROM {table\_name}"):  
  פקודה זו מבצעת שאילתת SQL שמחזירה את מספר השורות בטבלה בשם {table\_name}.
* fetchone()[0]:  
  fetchone() מחזיר את התוצאה הראשונה של השאילתה (במקרה הזה מספר השורות), ו- [0] נועד לגשת למספר השורות בתוצאה.
* print(f"Table {table\_name} in SQLite has {rows} rows."):  
  מדפיס את מספר השורות בטבלה כדי לוודא שהנתונים נשמרו כראוי.

**(11) ב' - רשימת הטבלאות שנמצאות בדאטא בייס:**

1. טבלה המייצגת את השאילתה – מהי הטמפרטורה הממוצעת לאורך כל תקופה?  
   מכילה: 500 שורות ו-2 עמודות.  
   הטבלה מראה לנו את הטמפרטורה הממוצעת בכל חודש בעזרת העמודות YearMonth ו- AvgTemp..
2. טבלה המייצגת את השאילתה – מהי כמות המשקעים הכוללת בכל תחנה ובחודש?  
   מכילה: 500 שורות ו-3 עמודות.  
   הטבלה מראה לנו את סכום המשקעים הכוללים לכל תחנה בעזרת העמודות YearMonth, station ו- TotalPrecipitation.
3. טבלה המייצגת את השאילתה – כמה מדידות יש בכל תחנה?  
   מכילה: 500 שורות ו-3 עמודות.  
   הטבלה מראה לנו את מספר המדידות בכל תחנה בעזרת העמודות station ו- DataCount.
4. טבלה המייצגת את השאילתה – מהן הטמפרטורות המקסימליות והמינימליות ומהירות הרוח המקסימלית בכל תחנה?

מכילה: 500 שורות ו-4 עמודות.  
הטבלה מציגה נתונים ממספר תחנות מדידה כאשר העמודות כוללות את שם התחנה ,(station) הטמפרטורה המרבית שנמדדה ,(MaxTemp) הטמפרטורה המינימלית ,(MinTemp) ומהירות הרוח המרבית .(MaxWindSpeed)

1. **דאשבורד - הסיפור שאנחנו רוצות לספר:**

אנחנו מנתחות נתוני מזג אוויר מאזור Central West Brazil כדי להבין דפוסי מזג אוויר עונתיים והשפעותיהם. האזור מאופיין במגוון אקלימי רחב והוא קריטי מבחינה חקלאית וכלכלית. בעזרת הנתונים, אנו שואפות לספק תובנות על עונות גשומות, טמפרטורות קיצוניות והשפעת רוחות חזקות.

איך נספר אותו?

1. הסבר רקע בעמוד "Story": תיאור מקורות הנתונים, תחומי המחקר והמטרות.

2. מעבר לשאלות בעמוד "Questions": נציג את השאלות המרכזיות שהמחקר עונה עליהן.

3. הצגת תובנות בעמוד "Summary and Insights": פירוט התובנות, כגון מגמות עונתיות בטמפרטורה ובמשקעים, והשפעת תנאי מזג קיצוניים.

4. המחשה ויזואלית בעמודים "Sample Rows" ו-"Visualizations": טבלאות לדוגמה מציגות את הנתונים וויזואליזציות מדגישות מגמות ודפוסים חשובים.

1. **תיאור סכמתי:**

1. Story: הסבר רקע, מטרות.

2. Questions: רשימת שאלות מחקר.

3. Summary and Insights: סיכום התובנות המרכזיות.

4. Sample Rows: טבלאות לדוגמה מכל אחת מהטבלאות הקטנות.

5. Visualizations: ויזואליזציות גרפיות ואינטראקטיביות.

1. **הוויזואליזציות שנבחרו:**
2. Average Temperature Over Time (Matplotlib):

מה רואים: גרף קווי המציג מגמות טמפרטורה ממוצעת לאורך זמן.

למה נבחר: עוזר לזהות מגמות עונתיות בטמפרטורה ומעברים בין תקופות חמות לקרות.

1. Monthly Precipitation by Station (Seaborn)**:**

מה רואים: ברים המייצגים את כמויות המשקעים החודשיות לפי תחנה.

למה נבחר: מדגיש תחנות עם כמויות משקעים חריגות ועונות גשומות.

1. Weather Stations Word Cloud:

מה רואים: ענן מילים המציג את תחנות מזג האוויר בגודל המתאר את פעילותן.

למה נבחר: ויזואליזציה יצירתית שמספקת מבט מהיר על תחנות פעילות.

1. Interactive: Temperature by Station:

מה רואים: טבלה אינטראקטיבית המציגה ערכי טמפרטורה לפי תחנה נבחרת.

למה נבחר: מאפשר למשתמש לחקור מידע מותאם אישית על תחנה מסוימת.

1. Interactive: Filter Precipitation by Month:

מה רואים: נתוני משקעים מסוננים לפי חודש נבחר.

למה נבחר: מספק למשתמש שליטה לחקור מגמות על פי חודש ספציפי.

**(18) מדריך מפורט להרצת הדאשבורד:**

1. וודא ש Python - מותקן במערכת שלך:

* python --version

אם אינך רואה גרסה, התקן Python מהאתר הרשמי: [https://www.python.org](https://www.python.org/).

**במהלך ההתקנה סמן את האפשרות "Add Python to PATH".**

------

1. הורדת התלויות:

ודא שיש לך את כלי pip מותקן. בדוק עם:

* python -m pip --version

אם הוא לא מותקן, הרץ:

python -m ensurepip --upgrade

* התקן את כל התלויות הנדרשות לדאשבורד. מהתיקייה שבה נמצאים הקבצים, הרץ: python -m pip install -r requirements.txt

-------

.3 בדוק שהקבצים הדרושים נמצאים בתיקייה:

וודא שכל הקבצים הבאים נמצאים בתיקייה אחת:

1. streamlit\_dashboard.py - קובץ הקוד של הדאשבורד.
2. central\_west\_data.db - מסד הנתונים.
3. requirements.txt - קובץ התלויות.

-------

.4 הרצת הדאשבורד:

נווט לתיקייה שבה נמצאים הקבצים בעזרת הפקודה:

1. cd PATH\_TO\_DIRECTORY

לדוגמה:

cd C:\Users\yaelk\PycharmProjects\pythonProject\big\_data\_analysis

1. הרץ את הדאשבורד:
2. python -m streamlit run streamlit\_dashboard.py

------

.5 פתיחת הדאשבורד:

* לאחר ההרצה Streamlit יפתח דפדפן אוטומטית. אם זה לא קורה Streamlit יציג בטרמינל כתובת URL כמו: Local URL: <http://localhost:8501> העתק את הכתובת ופתח אותה בדפדפן.

תוכל לנווט בין הדפים השונים באמצעות התפריט הצדדי.

------

6. ניווט בדאשבורד:

הדאשבורד מכיל כמה דפים:

- Questions: מציג את השאלות המרכזיות שהניתוח מנסה לענות עליהן.

- Story: מספק סקירה כללית של הסיפור והתובנות העיקריות מהניתוח.

- Sample Rows: מציג שורות דוגמה מכל הטבלאות שנוצרו.

- שאר הדפים מציגים וויזואליזציות שונות ותובנות ממאגר הנתונים.

-------

7. עצירת הדאשבורד:

כדי לעצור את הדאשבורד, השתמש בקיצור המקלדת "Ctrl + C" בחלון ה-Terminal שבו הופעל הפקודה של הרצת הדאשבורד.

-------

8. פתרון בעיות:

* שגיאה של מודול חסר:

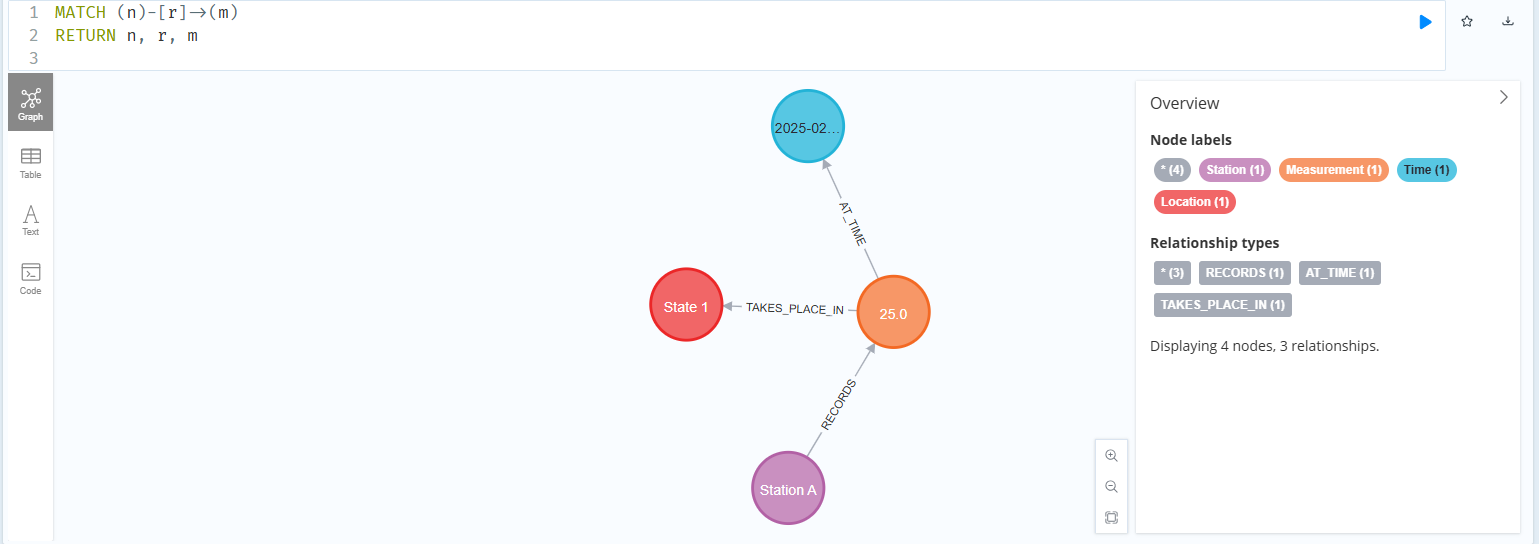
אם Streamlit או אחת התלויות חסרים, התקן אותם ידנית עם:

* python -m pip install streamlit
* python -m pip install matplotlib seaborn pandas wordcloud
* שגיאת :PATH אם Python או pip לא מזוהים, ודא ש - Python נוסף לנתיב המערכת.(PATH)

לחלופין, הפעל עם הנתיב המלא:

C:\Path\To\Python\python.exe -m streamlit run streamlit\_dashboard.py

ולאחר מכן תחזור על שלבים 4-7.

**(19) הצגת סכמה:**

תמונה שמכילה טקסט, תוכנה, סמל מחשב, תרשים

התיאור נוצר באופן אוטומטילאחר הרצת שורת הקוד שהתבקשנו להריץ:

פירוט טכני של הסכמה:

צמתים (ישויות):

* **Station**: מייצג את תחנת מזג האוויר.

מאפיינים:

station\_id, name, latitude, longitude, height, region, state.

* **Measurement**: מייצג כל רשומת מדידה של מזג האוויר.

מאפיינים**:**

timestamp, precipitation, atmospheric\_pressure, max\_atmospheric\_pressure, min\_atmospheric\_pressure, global\_radiation, temperature\_air, dew\_point\_temperature, max\_temperature, min\_temperature, max\_dew\_point\_temperature, min\_dew\_point\_temperature, max\_humidity, min\_humidity, wind\_direction, max\_gust\_speed, wind\_speed.

* **Location**: אופציונלי, משמש לייצוג מיקומים כלליים לצורך קיבוץ תחנות או נתונים אזוריים.

מאפיינים:

region, state.

* **:Time** מייצג תקופות זמן או חותמות זמן בודדות של מדידות.  
  מאפיינים:

Timestamp.

יחסים (קשתות):

* **RECORDS**: קשר בין Station ל-Measurement.  
  תחנה "מתעדת" מדידות בזמנים ספציפיים.  
  **מאפיינים:**

measurement\_time (timestamp), measurement\_id.

* TAKES\_PLACE\_IN: קשר בין **Measurement** ל- **Location.**

מדידה מתבצעת במקום מסוים (אזור או מדינה).

* AT\_TIME: קשר בין **Measurement** ל- **Time**.  
  מדידה מתבצעת בזמן מסוים.
* HAS\_MEASUREMENT: קשר בין **Station** ל- **.Measurement**כל תחנה מכילה מספר מדידות המחוברות אליה.
  + 1. **תיאור הסכמה המצוירת:**

התרשים מציג גרף נתונים מתוך מסד נתונים גרפי (Neo4j), שבו מופיעים צמתים (Nodes) וקשרים (Relationships) ביניהם.

* **תחנת מזג אוויר (Station)** מתעדת **מדידות (Measurement)** בזמנים שונים**.**
* **כל מדידה** מתרחשת **באזור גאוגרפי (Location)** ספציפי.
* **לכל מדידה** יש חותמת זמן והיא מתבצעת **בזמן מסוים .(Time)**
* **לכל קשר** יש מאפיינים משלו, כגון חותמת זמן למדידה או מזהה מיקום.

מרכיבי הסכמה בתרשים:

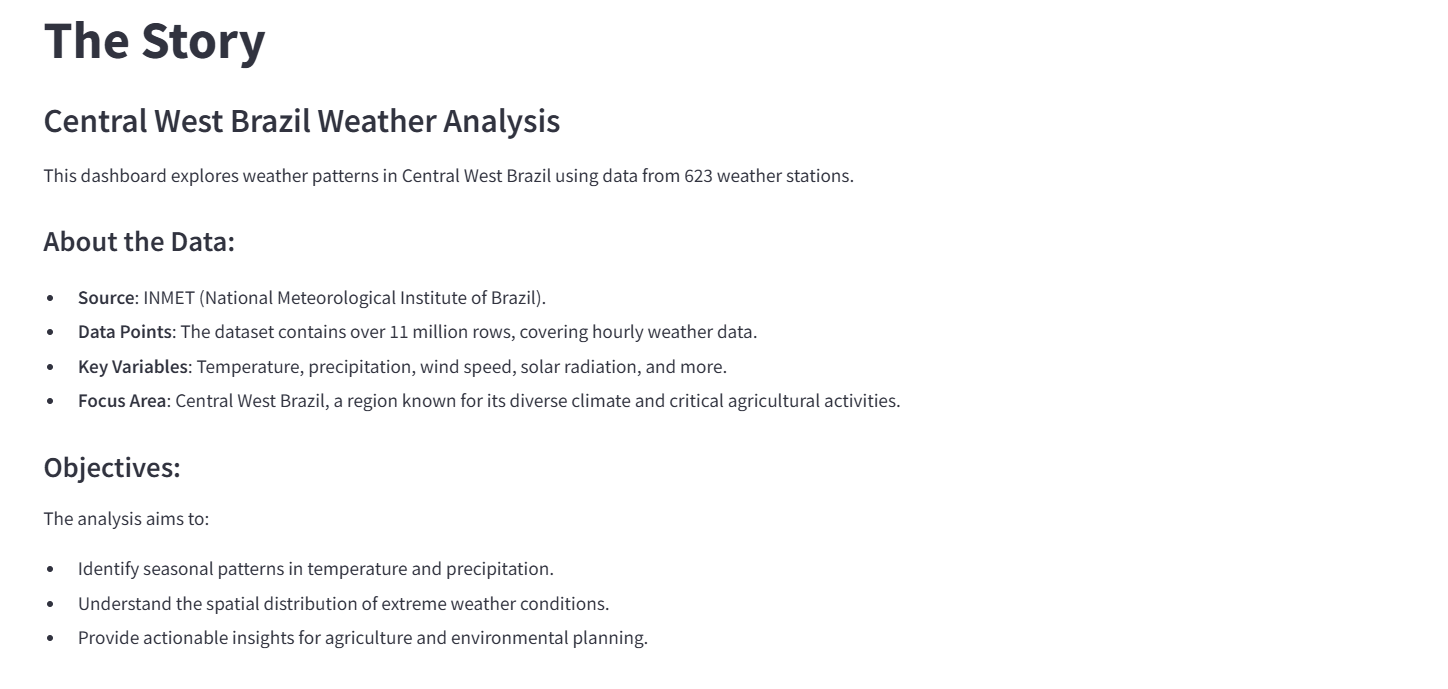
* Station (תחנה): מיוצגת בצבע סגול, למשל "Station A".
* Measurement (מדידה): מיוצגת בצבע כתום, עם ערך מדידה (למשל 25.0).
* Location (מיקום): מיוצגת בצבע אדום, למשל "State 1".
* Time (זמן): מיוצגת בצבע כחול, ומציינת חותמת זמן (למשל "2025-02...").
* הקשרים (Relationships):
* RECORDS (מתעד): הקשר בין תחנה ל-מדידה, מציין כי תחנה מסוימת ביצעה את המדידה.
* TAKES\_PLACE\_IN (מתבצע ב): הקשר בין מדידה ל-מיקום, מציין כי המדידה התבצעה באזור מסוים.
* AT\_TIME (בוצע בזמן): הקשר בין מדידה ל-זמן, מציין מתי המדידה בוצעה.

מסקנות מהתרשים:

* תחנה מסוימת ביצעה מדידה של ערך 25.0.
* המדידה התקיימה במדינה או אזור בשם "State 1".
* המדידה התרחשה בזמן מסוים בפברואר 2025.

התרשים ממחיש היטב את המבנה הגרפי של הנתונים וכיצד ישויות שונות מקושרות זו לזו במסד הנתונים.

**(25) צילומי מסך שכל דפי הדאשבורד:**

****

**תמונה שמכילה טקסט, צילום מסך, גופן

התיאור נוצר באופן אוטומטי**

**תמונה שמכילה טקסט, צילום מסך, גופן

התיאור נוצר באופן אוטומטי**

**תמונה שמכילה טקסט, צילום מסך, גופן, עיצוב

התיאור נוצר באופן אוטומטי**

**תמונה שמכילה טקסט, צילום מסך, גופן, מספר

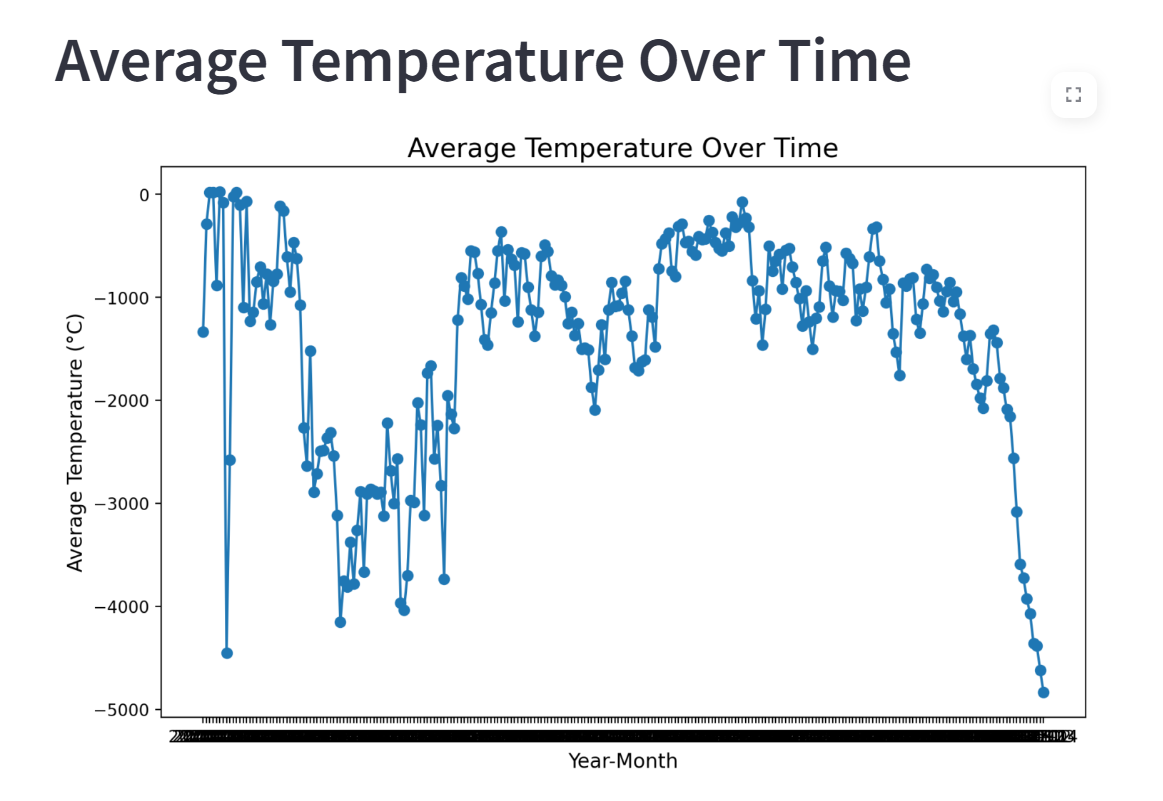
התיאור נוצר באופן אוטומטי**

**תמונה שמכילה טקסט, צילום מסך, מספר, גופן

התיאור נוצר באופן אוטומטי**

**תמונה שמכילה טקסט, צילום מסך, מספר, גופן

התיאור נוצר באופן אוטומטי**

**** **תמונה שמכילה טקסט, צילום מסך, תרשים, עלילה

התיאור נוצר באופן אוטומטי** **תמונה שמכילה טקסט, גופן, גרפיקה, עיצוב גרפי

התיאור נוצר באופן אוטומטי** **תמונה שמכילה טקסט, צילום מסך, גופן

התיאור נוצר באופן אוטומטי**

**(26) סיכום – תיאור הארכיטקטורה והקוד:**

ארכיטקטורת הפרויקט:

הפרויקט בנוי במספר שלבים הכוללים עיבוד נתונים, אחסון, ניתוח והצגה באמצעות ממשק אינטראקטיבי.

1. טעינת הנתונים ואחסון
   * הנתונים נטענים מקובץ CSV לתוך מסד הנתוניםDuckDB המספק יכולות עיבוד מהירות ואפקטיביות.
   * לאחר ביצוע עיבודים וחישובים ראשוניים, הנתונים הרלוונטיים נשמרים במסד הנתונים SQLite כדי לאפשר אינטגרציה נוחה עם ה - .Dashboard
2. עיבוד הנתונים וניתוח
   * בוצעו שאילתות SQL על גביDuckDB לניתוח מגמות מרכזיות כגון ממוצעי טמפרטורה, משקעים חודשיים, כמות הנתונים בתחנות שונות ותנאי מזג אוויר קיצוניים.
   * הנתונים עובדו והומרו ל DataFrame -של Pandas לצורך ניתוח נוסף.
3. ויזואליזציה והצגת הנתונים
   * שימוש בספריות Matplotlib ו Seaborn -המשמשות ליצירת גרפים של מגמות טמפרטורה, משקעים והתנהגויות קיצוניות.
   * שימוש בספריית WordCloudהמשמשת להצגת תחנות מזג האוויר בתצורת ענן מילים.
   * ממשק אינטראקטיבי שנבנה באמצעותStreamlit המאפשר למשתמשים לנווט בין מסכים שונים ולחקור את הנתונים בעצמם.

תיאור הקוד המרכזי:

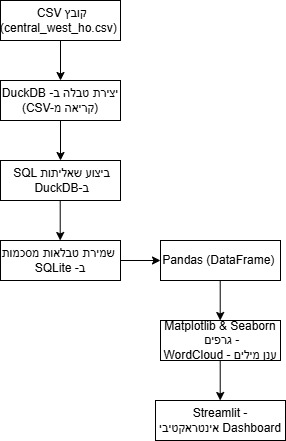
* טעינת הנתונים :קובץ ה- CSVנטען למסד DuckDB ולאחר ביצוע השאילתות נשמרות טבלאות מסכמות במסד .SQLite
* ביצוע שאילתות :SQL מבוצעות שאילתות הכוללות חישובי ממוצעים, סכומים וחיתוכי נתונים לפי תחנות ושנים.
* שימוש ב- :Streamlit נבנה Dashboardהכולל מסכים שונים עם הסברים, תובנות, ודוגמאות נתונים.
* יצירת ויזואליזציות: שרטוטים הכוללים גרפים ליניאריים, תרשימי עמודות וענן מילים כדי לספק המחשה חזותית ברורה של הנתונים.

**מסקנות עיקריות:**

* זוהו דפוסי טמפרטורה ומשקעים לאורך השנים באזור.
* נמצאו תחנות עם תנאים קיצוניים כגון טמפרטורות גבוהות במיוחד או רוחות חזקות.
* הוצעו המלצות לניהול משאבי מים ומעקב אחר אזורים רגישים למזג אוויר קיצוני.

הפרויקט משלב טכנולוגיות מודרניות לניתוח והדמיית נתונים, ומאפשר חוויית משתמש אינטראקטיבית באמצעות .Streamlit

ציור סכמתי של המערכת:



**(27) חלוקת אחריות:**

* **יעל:**
  + - בחירת הדאטה בייס ותיאורו.
    - כתיבת הקוד ליצירת הטבלאות ושמירתן.
    - כתיבת הקוד להצגת הדאשבורד.
    - כתיבת הסיפורים ופירוטיהם.
    - כתיבת המדריכים הטכניים.
* **הדר:**
  + - כתיבת השאילתות ופירוטן.
    - עריכת הקוד בהתאם לשאילתות.
    - יצירת הגרף ותיאורו.
    - כתיבת סיכום העבודה.
    - עריכת מסמך הוורד.