

דוח פרויקט בסיסי נתונים:

בפרויקט שה נעסוק בעיצוב ובניית בסיס נתונים למוזיאון, בסיס הנתונים כולל עובדים, מבקרים, מוצגים, יוצרים, סיורים, סדנאות, תערוכות, מוצגים מושאלים, ציוד נלווה, בניינים, חדרים, מחלקות, אגפים, תקלות, וחנות מזכרות.

החלק שלנו בפרויקט הוא בעיצוב ובניית הישויות

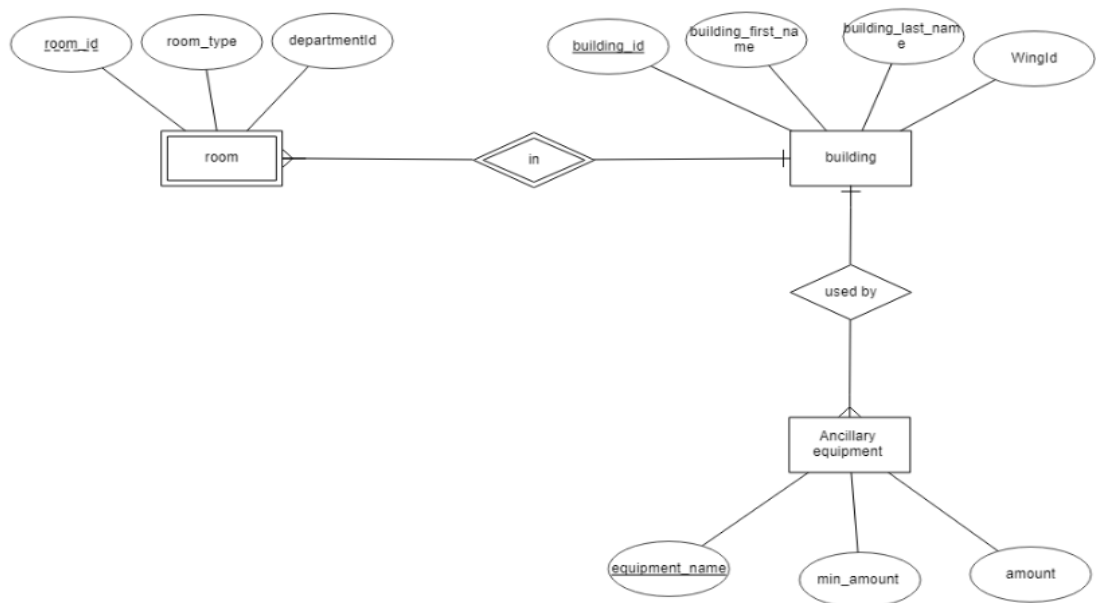
(1 בניין

(2 חדר

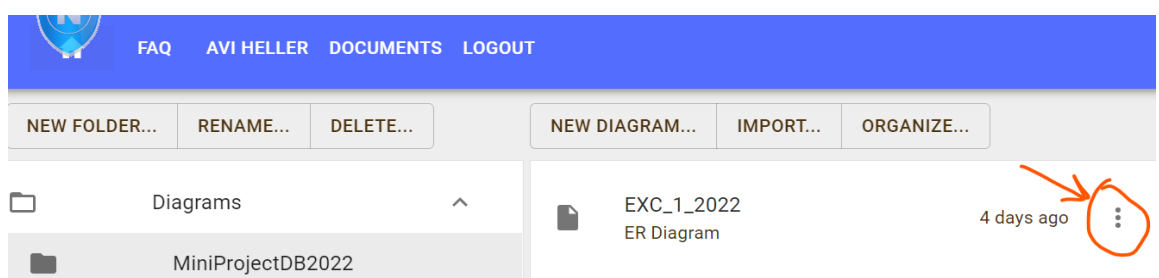
(3 ציוד היקפי

ראשית נתחיל את המסע על ידי כך שנבנה את תרשים ERD עבור הישויות הנ"ל.

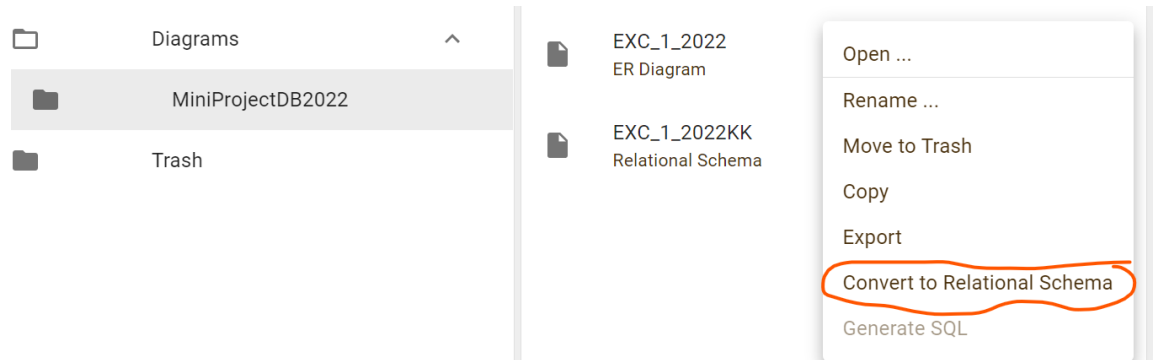
עבור בניית התרשים ניעזר באתר <https://erdplus.com/>



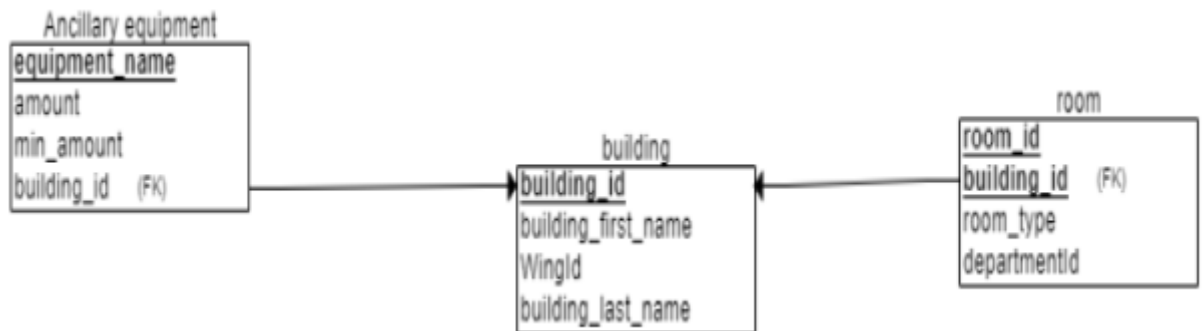
לאחר בניית התרשים נלחץ על שלושת הנקודות שבצד הקובץ.



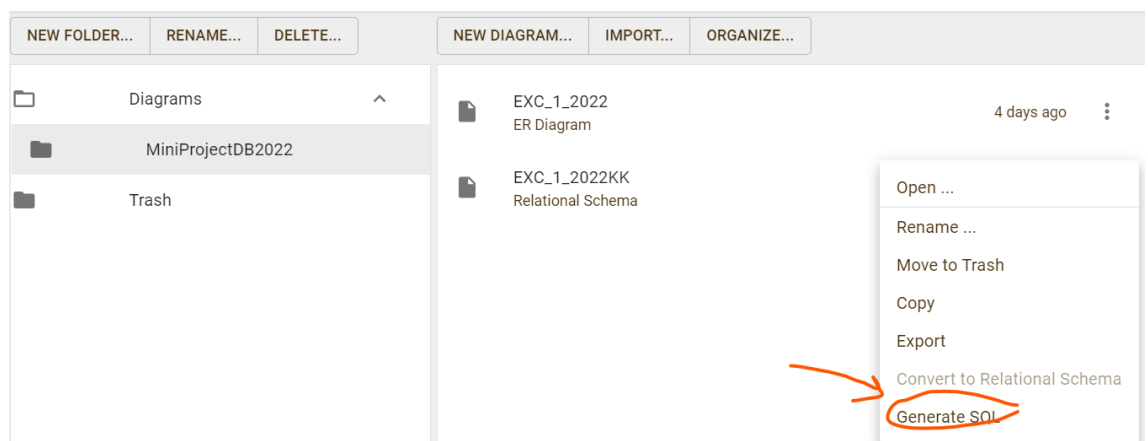
ונקבל אפשרות להמיר את התרשים שלנו לסכמה רלציונית



אחרי ההמרה נקבל את הסכמה הרלציונית הבאה:



לאחר קבלת הסכמה הרלציונית נוכל באותו הדרך לקבל אפשרות להמיר את הסכמה הרלציונית לקוד SQL ליצירת הטבלאות.



לאחר קבלת קוד ה SQL לרוב יש צורך לערוך את הקוד לקבל את הטבלאות המדויקות, סוגי הטיפוסים וגודלם ועוד מספר דברים שההמרה לא יודעת לעשות, לאחר ההתאמות קיבלנו את הקוד הבא:

```

CREATE TABLE Buildings
(
    building_id INT NOT NULL,
    building_first_name VARCHAR(20) NOT NULL,
    building_last_name VARCHAR(20) NOT NULL,
    wing_id INT NOT NULL,
    PRIMARY KEY (building_id),
    FOREIGN KEY (wing_id) REFERENCES MKLEMFNE.WING(wingid)
);

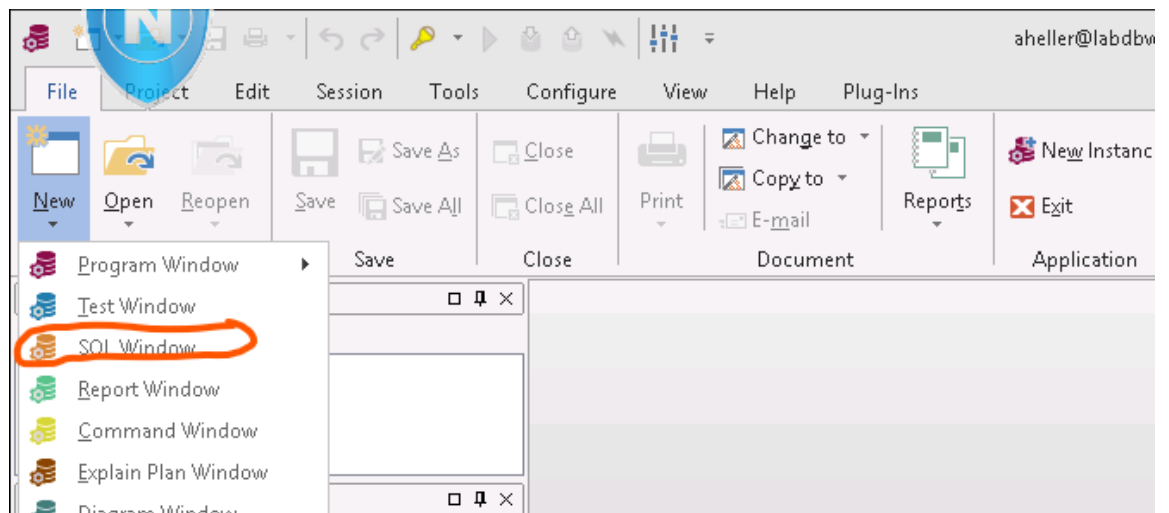
CREATE TABLE Room_types
(
    room_type_id INT NOT NULL,
    name VARCHAR(20) NOT NULL,
    PRIMARY KEY (room_type_id)
);

CREATE TABLE Rooms
(
    room_id INT NOT NULL,
    room_type_id INT NOT NULL,
    building_id INT NOT NULL,
    department_id INT NOT NULL,
    PRIMARY KEY (room_id),
    FOREIGN KEY (building_id) REFERENCES Buildings(building_id),
    FOREIGN KEY (room_type_id) REFERENCES Room_types(room_type_id),
    FOREIGN KEY (department_id) REFERENCES MKLEMFNE.DEPARTMENT(department_id)
);

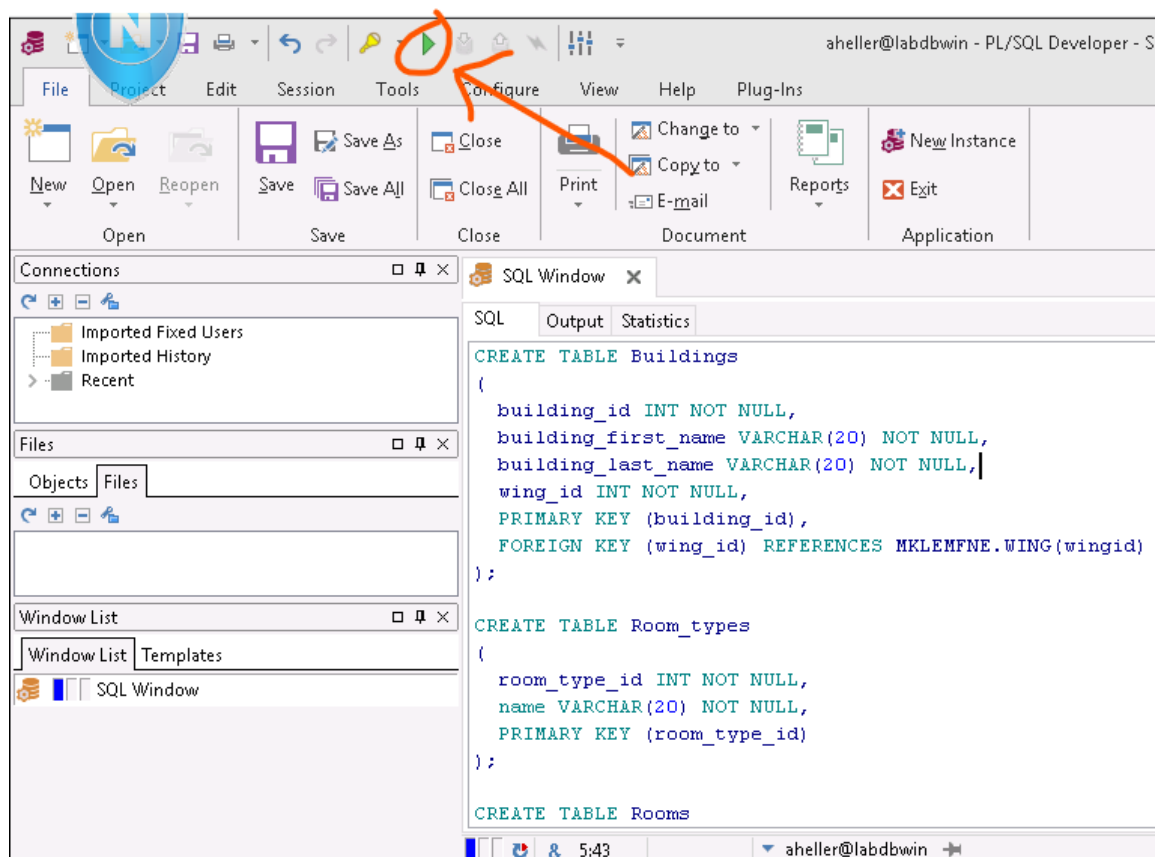
CREATE TABLE Equipments
(
    equipment_name VARCHAR(40) NOT NULL,
    amount INT NOT NULL,
    min_amount INT NOT NULL,
    building_id INT NOT NULL,
    PRIMARY KEY (equipment_name),
    FOREIGN KEY (building_id) REFERENCES Buildings(building_id)
);

```

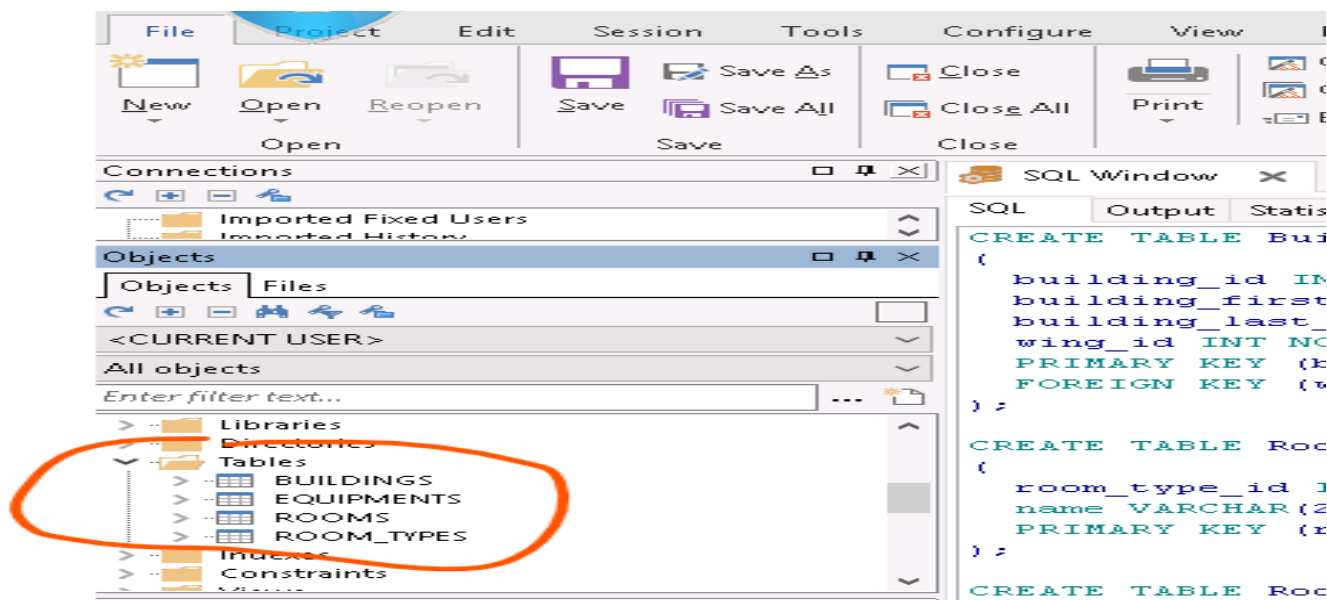
לאחר קבלת הקוד ניגש לבניית הטבלאות בתוכנה PLSQL, בצד השמאלי למעלה נבחר בחלון SQL -



נכניס את הקוד של בניית הטבלאות לקובץ:



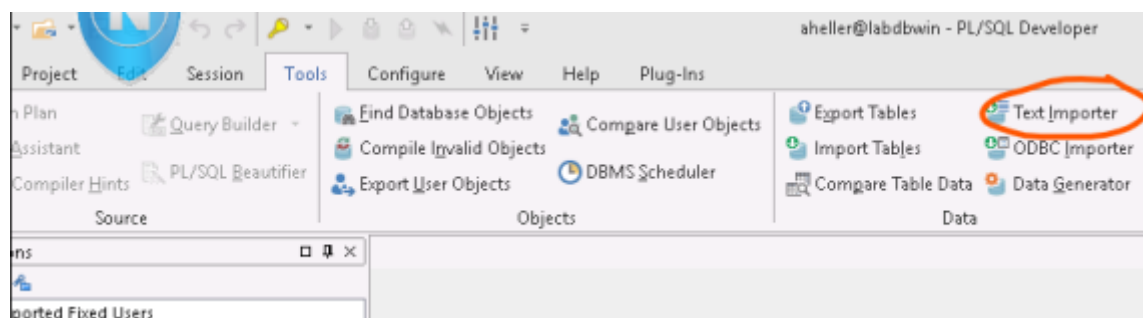
נריץ ונוכל למצוא את הטבלאות בתיקיית הטבלאות:



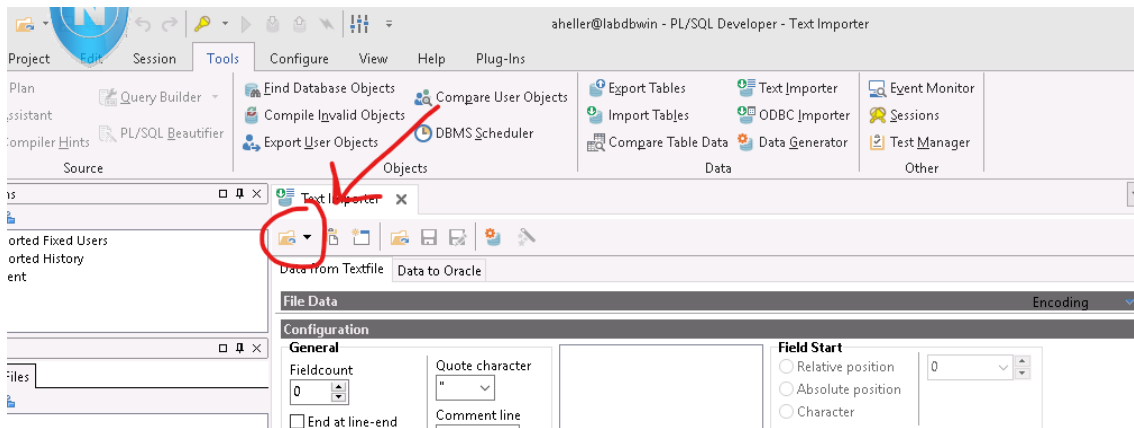
אכלוס הטבלאות:

עבור אכלוס הטבלאות כתבנו קוד פיתון לכל ישות וישות, המייצרת ערכים מתאימים בצורה רנדומלית (קבצי הקוד מאוחסנים בגיטהאב) ומייצרים קובץ CSV אותו נטען לפרויקט. תהליך הטעינה המפורט:

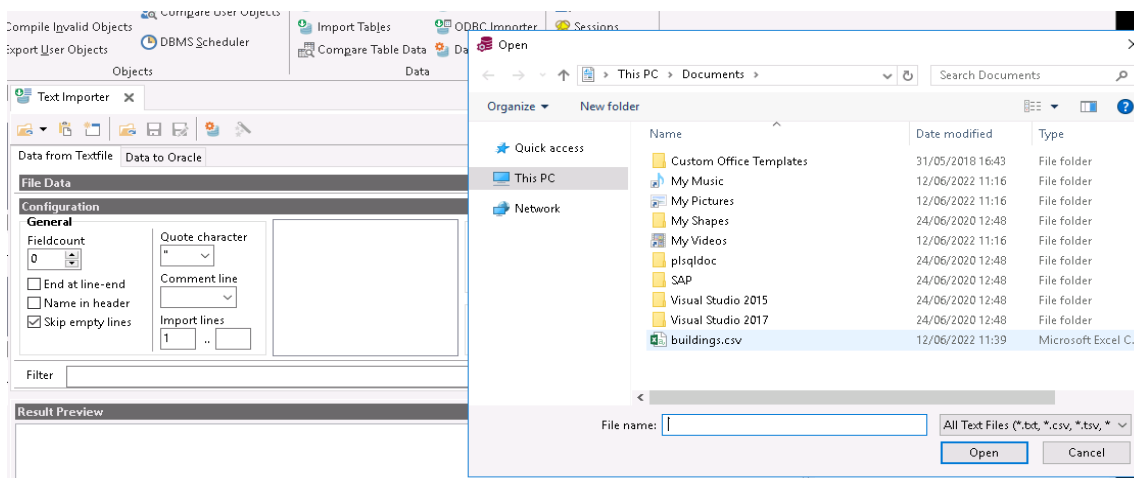
Tools -> Text Importer



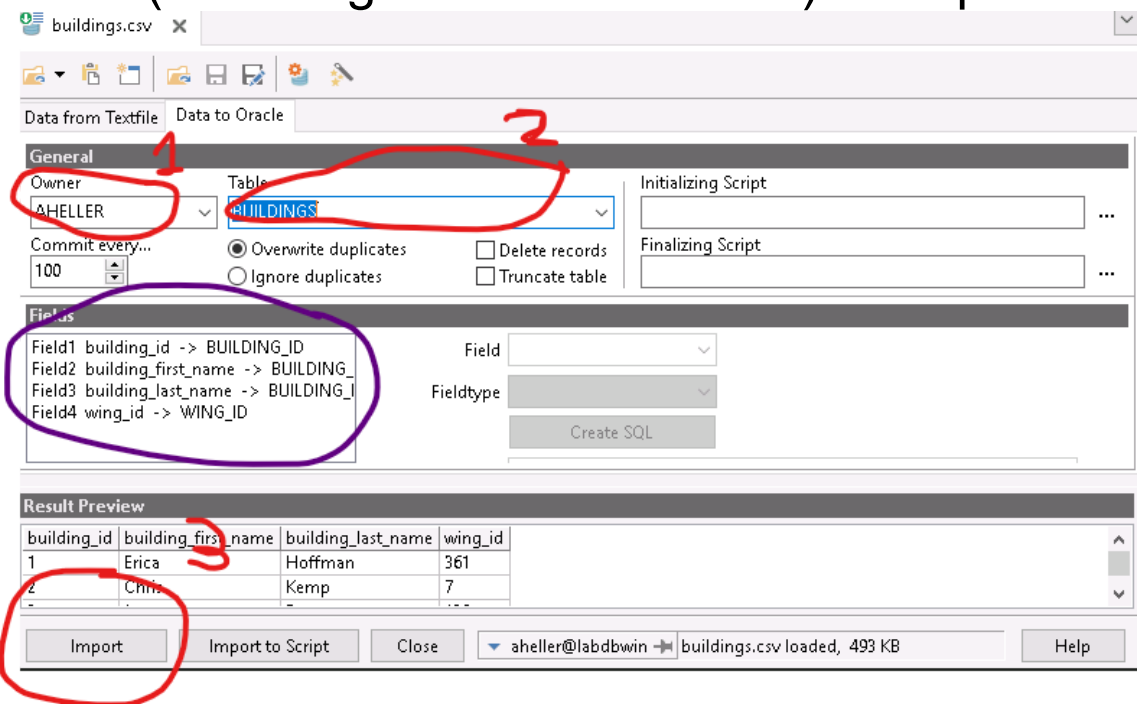
-> open data file



-> choose csv file path from explorer



data to orcal -> (choosing the relevant table) -> Import.



(להשים לב שהשדות שבטבלה מתאימות לשדות שבקובץ CSV)
(אחרת ניתן יהיה להתאים אותן ידנית)

מתן הרשאות:
נפתח חלון SQL חדש ובו נזין את פקודות ה GRANT שיאפשרו
מתן גישה לשותפי הפרויקט.

```
GRANT SELECT, INDEX, REFERENCES ON Buildings TO SORIA, ZAGURI;  
GRANT SELECT, INDEX, REFERENCES ON Rooms TO SORIA, ZAGURI;  
GRANT SELECT, INDEX, REFERENCES ON EQUIPMENTS TO SORIA, ZAGURI;  
GRANT all ON buildings TO SILON;  
GRANT all ON Rooms TO SILON;  
GRANT all ON room_types TO SILON;  
GRANT all ON EQUIPMENTS TO SILON;
```

אחרי ההרצה נוכל לראות את ההרשאות בצורה הבאה:

שאלות:

(1) מספר הבניינים שתרמה משפחה מסוימת.

```
SELECT b.building_last_name, COUNT(*)  
FROM Building b  
GROUP BY b.building_last_name
```

(2) כל הבניינים שיש בהם חדרי תצוגה של המחלקה שמספר
המזהה שלה נתון כקלט בידי המשתמש

```
SELECT b.building_id  
FROM Buildings b  
WHERE EXISTS (SELECT *  
              FROM ROOMS r  
              WHERE r.department_id = &<name="id" hint="Department ID">  
                    AND r.room_type_id = 1  
                    AND r.building_id = b.building_id)
```

(3) כל הבניינים שיש להם יותר מ - 7 חדרי תצוגה
+ חדר מנהלה אחד ולא חסר בהם ציוד (עבור אירוח תערוכה
נדרשים מספר בניינים מתאימים)

```

SELECT b.building_id
FROM Buildings b
WHERE EXISTS (SELECT *
              FROM Room_by_type r
              WHERE r.room_type_id = 5)

INTERSECT

SELECT b.building_id
FROM Buildings b
WHERE b.building_id IN (
    SELECT x.building_id
    FROM (SELECT r.building_id, COUNT(r.room_id) c
          FROM Room_by_type r
          WHERE r.room_type_id = 1
          GROUP BY r.building_id ) x
    WHERE x.c > 7)

INTERSECT

SELECT b.building_id
FROM buildings b
WHERE b.building_id NOT IN (SELECT e.building_id
                           FROM Equipments e
                           WHERE e.amount < e.min_amount)

```

- (4) שאילתה עבור כל רשימת משפחות התורמים לפי אגפים, וממוינת -
 (א) לפי האגפים
 (ב) לפי כמות הבניינים שהמשפחה תרמה.

```

SELECT b.wing_id, b.building_last_name, COUNT(*) c
FROM Buildings b
GROUP BY b.wing_id, b.building_last_name
ORDER BY b.wing_id, c DESC

```

- (5) שאילתה עבור כל חדרי התצוגה מכל הבניינים

```

select r.room_id
from rooms r inner join buildings b on r.building_id = b.building_id
where r.room_type_id = 5

```

- (6) שאילתה עבור כל הבניינים שיש בהם תצוגה אבל אין מנהלה (בניינים אלו משמשים כבנייני עזר למחלקת ההוראה)


```

SELECT b1.building_id
FROM Buildings b1
WHERE EXISTS( SELECT *
               FROM Room_by_type r
               WHERE r.room_type_id = 1
               and r.building_id = b1.building_id)

INTERSECT

SELECT b2.building_id
FROM Buildings b2
WHERE NOT EXISTS( SELECT *
                  FROM Room_by_type r
                  WHERE r.room_type_id = 5
                  and r.building_id = b2.building_id)

```

(7) שאילתה עבור כל הציוד שיש להזמין לבניינים.

```

SELECT e.equipment_name e.min_amount - e.amount
FROM Equipment e
WHERE e.amount < e.min_amount

```

(8) שאילתה של כל הציוד שיש צורך להזמין לפי אגפים ואת הכמות להזמנה.

```

SELECT t.wing_id, t.equipment_name, SUM(t.min_amount - t.amount)
FROM Equipments e JOIN Wing w AS t
WHERE t.min_amount < t.amount
GROUP BY t.wing_id, t.equipment_name

```

(9) שאילתה לעדכון כמות מוצר, כאשר שם המוצר מוזן כקלט מהמשתמש.

```

update Equipments e
set amount = &<name="amount" type="integer" required="true">
where e.equipment_name = &<name="name"
      list= "select equipment_name from Equipments"
      restricted="yes"
      type = "string">

```

טבלאות וירטואליות (views):

1) טבלת החדרים חדרים שמשפחה מסוימת תרמה (היא תרמה את הבניין כולל את חדריו נרצה להציע למשפחת

התורם לתחזק ולשפץ את החדרים כשצריך)

```
create view Room_by_founder_family AS
SELECT b.building_last_name, r.room_id, b.wing_id,r.department_id
FROM Buildings b INNER JOIN Rooms r ON b.building_id = r.building_id
ORDER BY b.building_last_name
```

2) טבלת החדרים לפי הענף שהן נמצאים

```
create view ROOM_BY_WING as
SELECT r.room_id, w.wingid
FROM MKLEMFNE.WING w
INNER JOIN BUILDINGS b on b.wing_id = w.wingid
INNER JOIN ROOMS r on r.building_id = b.building_id
order by r.room_id;
```