

שם הארגון: בית ספר

שם האגף: ילדים

מגישות:

ספיר פיק 326378155

sapirf1973@gmail.com

הדסה שטולמן 325462430

dasstlman@gmail.com

תוכן העניינים

3.....	מבוא
3.....	היישואיות במערכת
4.....	תרשים ERD של המערכת
4.....	תרשים DSD של המערכת
5.....	בניית בסיס הנתונים
5.....	יצירת הטבלאות
7.....	תיאור הטבלאות
8.....	הכנסת נתונים
9.....	Data Generator
11.....	Data Importer
11.....	Programming
13.....	גיבוי הנתונים ושיחזור
14.....	שאילתות
14.....	Select שאילתות
16.....	Update שאילתות
18.....	Delete שאילתות
20.....	שאילתות פרמטרים
24.....	אילוצים

מבוא

בפרוייקט זה נממש מערכת ניהול מידע עבור בית ספר. המערכת מכילה ישויות מידע שונות וקשרים ביניהם על מנת למפות את כל המידע בצורה מיטבית. מטרת הפרוייקט היא לעזור לנהל את בית הספר בצורה יעילה ומתועדת היטב.

היישויות במערכת

תלמידים: פרטים אישיים של ילדים הלומדים בבית ספר

מורה: פרטים אישיים של מורים המלמדים בבית ספר

כיתה: קבוצת תלמידים שלומדים יחד, לכל כיתה יש מחנכת אחת

חדר: החדרים הפיזיים במבנה בית הספר. לכל כיתה אם יש חדר לימוד קבוע

מקצוע: תחום לימודים כלשהו הנלמד בשיעורים, מקצוע יכול להיות חובה או רשות.

שיעור: רכיב מערכת שבו מורה מלמדת מקצוע מסויים בכיתה מסויימת

Entities:

Student = Student_ID, FirstName, LastName, BirthDate, FathersName, MothersName, Phone, Class_ID

Teacher = Teacher_ID, FirstName, LastName

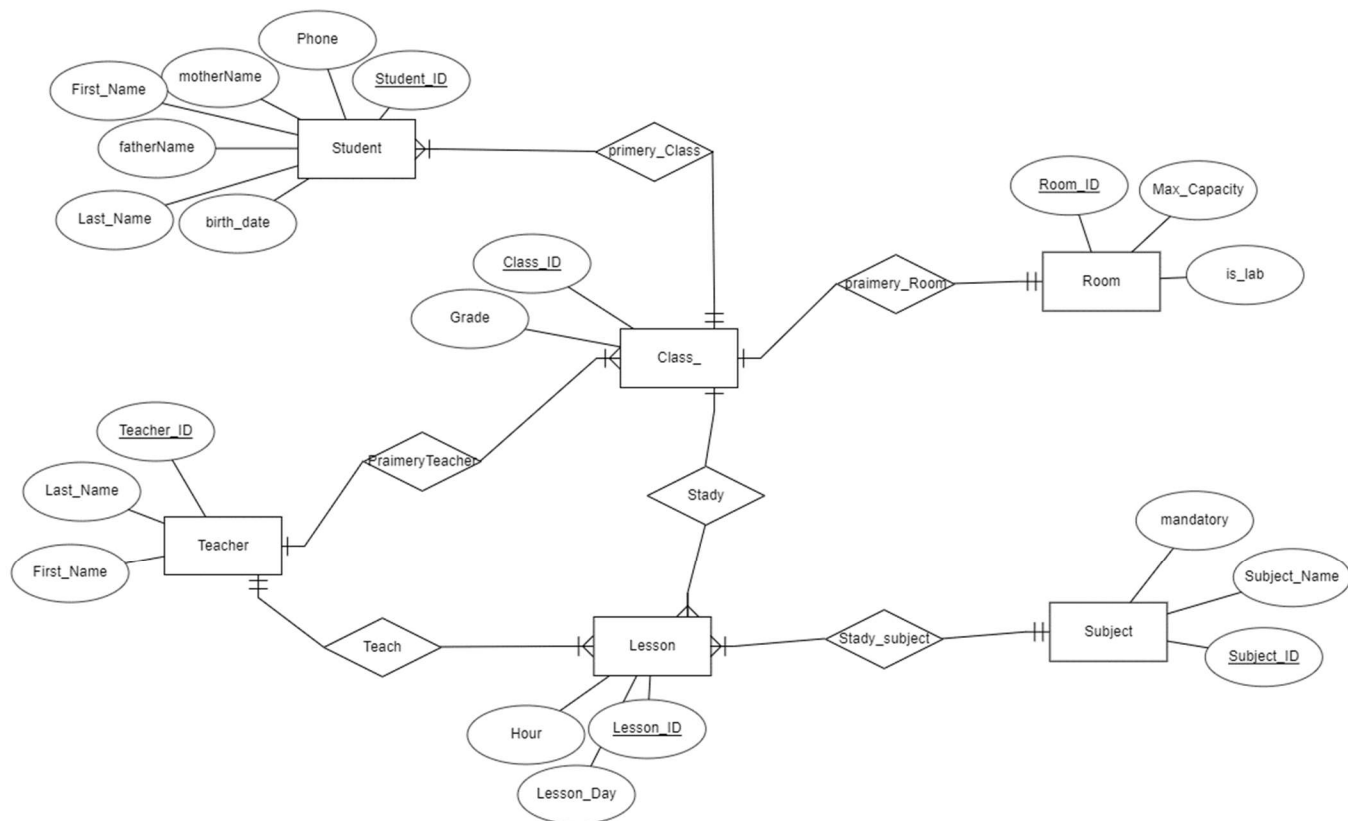
Class = Class_ID, Grade, Room_ID, Teacher_ID

Room = Room_ID, MaxCapacity, is_lab

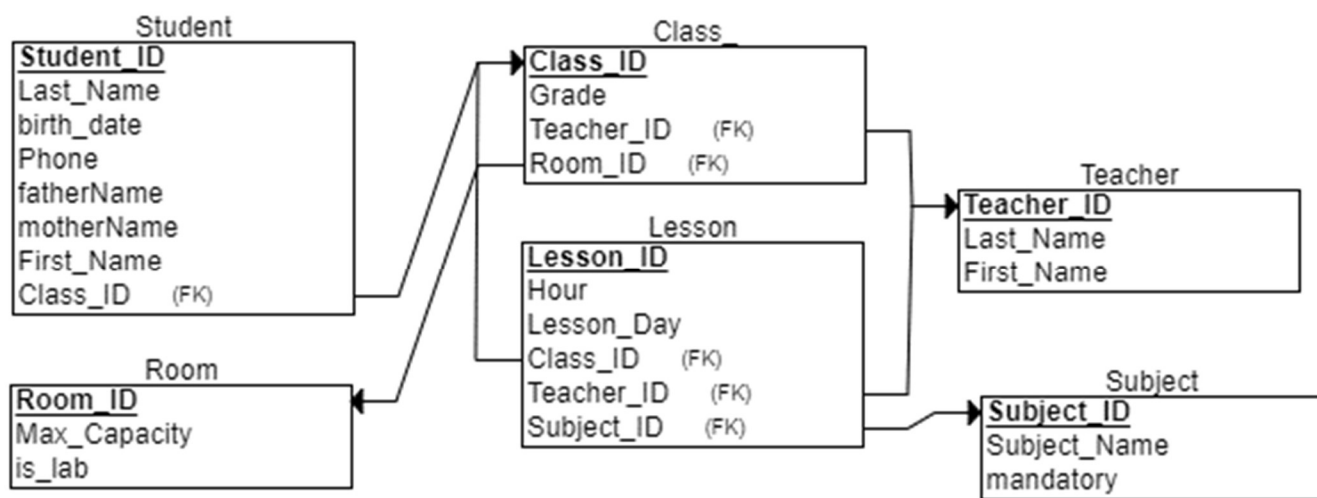
Subject = Subject_ID, SubjectName, is_mandatory

Lesson = Lesson_ID, Lesson_Day, Hour, Subject_ID, Teacher_ID

תרשים ERD של המערכת



תרשים DSD של המערכת



בניית בסיס הנתונים

יצירת הטבלאות

```

CREATE TABLE Teacher
(
  Teacher_ID NUMBER(3),
  TLast_Name VARCHAR2(15) NOT NULL,
  TFirst_Name VARCHAR2(15) NOT NULL,
  PRIMARY KEY (Teacher_ID)
);

CREATE TABLE Room
(
  Room_ID NUMBER(3),
  Max_Capacity INT NOT NULL,
  is_lab NUMBER(1),
  Last_maintenance_check DATE,
  PRIMARY KEY (Room_ID),
  CHECK (is_lab = 1 OR is_lab = 0)
);

CREATE TABLE Subject
(
  Subject_ID NUMBER(3),
  Subject_Name VARCHAR2(30) NOT NULL,
  mandatory number(1) NOT NULL,
  PRIMARY KEY (Subject_ID),
  CHECK (mandatory = 1 OR mandatory = 0)
);

CREATE TABLE Class_
(
  Class_ID NUMBER(3),
  Grade NUMBER(2) NOT NULL,
  Teacher_ID NUMBER(3) NOT NULL,
  Room_ID NUMBER(3) NOT NULL,
  PRIMARY KEY (Class_ID),
  FOREIGN KEY (Teacher_ID) REFERENCES Teacher(Teacher_ID),
  FOREIGN KEY (Room_ID) REFERENCES Room(Room_ID)
);

```

```

CREATE TABLE Lesson
(
    Lesson_ID NUMBER(3),
    Lesson_Day INT NOT NULL,
    Lesson_Hour INT NOT NULL,
    Class_ID NUMBER(3),
    Teacher_ID NUMBER(3),
    Subject_ID NUMBER(3),
    PRIMARY KEY (Lesson_ID),
    FOREIGN KEY (Class_ID) REFERENCES Class_(Class_ID),
    FOREIGN KEY (Teacher_ID) REFERENCES Teacher(Teacher_ID),
    FOREIGN KEY (Subject_ID) REFERENCES Subject(Subject_ID),
    UNIQUE (Class_ID, Lesson_Day, Lesson_Hour),
    UNIQUE (Teacher_ID, Lesson_Day, Lesson_Hour),
    CHECK (Lesson_Day BETWEEN 1 AND 6)
);

CREATE TABLE Student
(
    Student_ID NUMBER(3),
    First_Name VARCHAR(15) NOT NULL,
    Last_Name VARCHAR2(15) NOT NULL,
    birth_date DATE NOT NULL,
    Phone INT NOT NULL,
    fatherName VARCHAR(15) NOT NULL,
    motherName VARCHAR(15) NOT NULL,
    Class_ID NUMBER(3),
    PRIMARY KEY (Student_ID),
    FOREIGN KEY (Class_ID) REFERENCES Class_(Class_ID)
);

```

תיאור הטבלאות

SQL> desc room

Name	Type	Nullable	Default	Comments
ROOM_ID	NUMBER(3)			
MAX_CAPACITY	INTEGER			
IS_LAB	NUMBER(1)	Y		
LAST_MAINTENANCE_CHECK	DATE	Y		

SQL> desc subject

Name	Type	Nullable	Default	Comments
SUBJECT_ID	NUMBER(3)			
SUBJECT_NAME	VARCHAR2(30)			
MANDATORY	NUMBER(1)			

SQL> desc teacher

Name	Type	Nullable	Default	Comments
TEACHER_ID	NUMBER(3)			
TLAST_NAME	VARCHAR2(15)			
TFIRST_NAME	VARCHAR2(15)			

SQL> desc class_

Name	Type	Nullable	Default	Comments
CLASS_ID	NUMBER(3)			
GRADE	NUMBER(2)			
TEACHER_ID	NUMBER(3)			
ROOM_ID	NUMBER(3)			

SQL> desc lesson

Name	Type	Nullable	Default	Comments
LESSON_ID	NUMBER(3)			
LESSON_DAY	INTEGER			
LESSON_HOUR	INTEGER			
CLASS_ID	NUMBER(3)	Y		
TEACHER_ID	NUMBER(3)	Y		
SUBJECT_ID	NUMBER(3)	Y		

SQL> desc student

Name	Type	Nullable	Default	Comments
STUDENT_ID	NUMBER(3)			
FIRST_NAME	VARCHAR2(15)			
LAST_NAME	VARCHAR2(15)			
BIRTH_DATE	DATE			
PHONE	INTEGER			
FATHERNAME	VARCHAR2(15)			
MOTHERNAME	VARCHAR2(15)			
CLASS_ID	NUMBER(3)	Y		

הכנסת נתונים

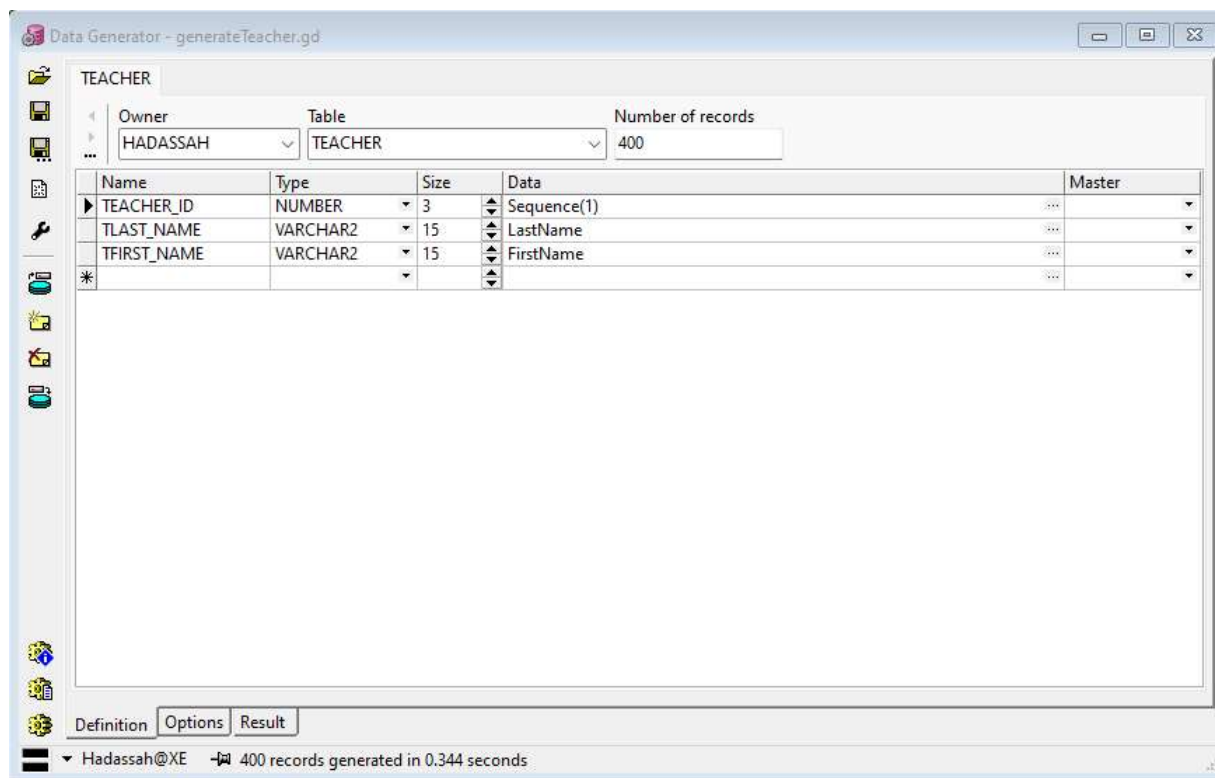
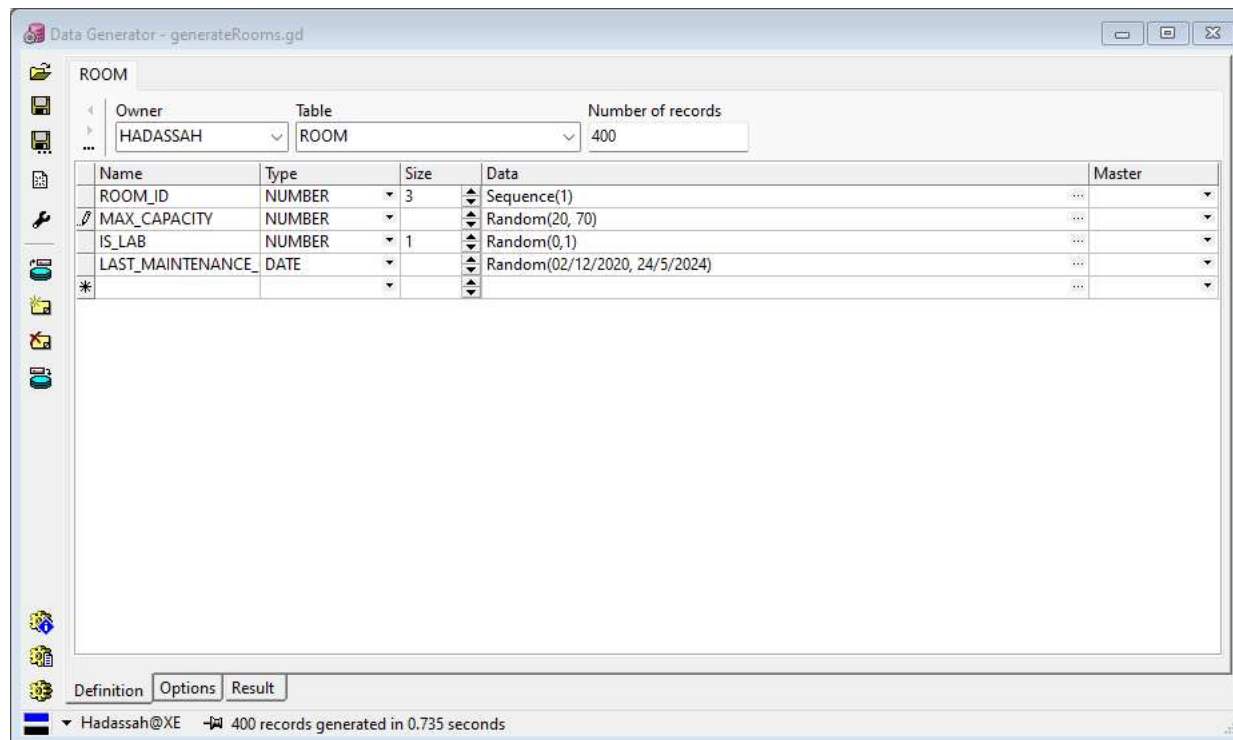
השתמשנו על פי ההנחיות ב-3 שיטות להכנסת נתונים: data generator, data importer, programming

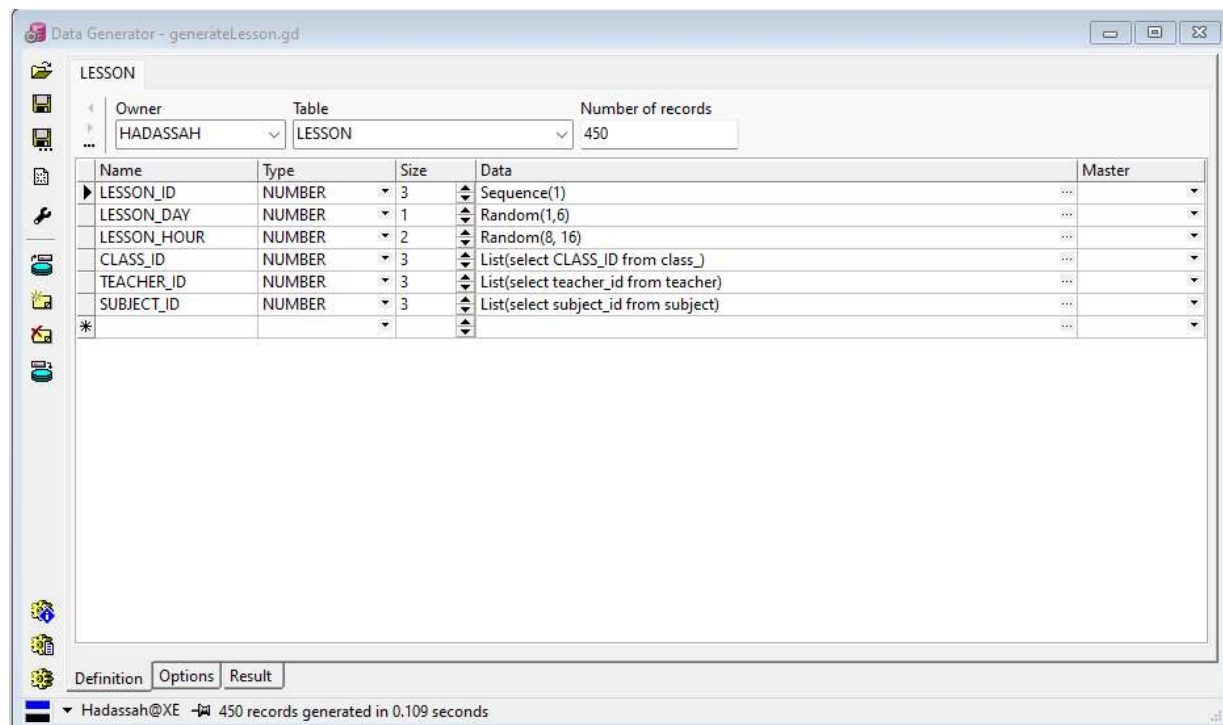
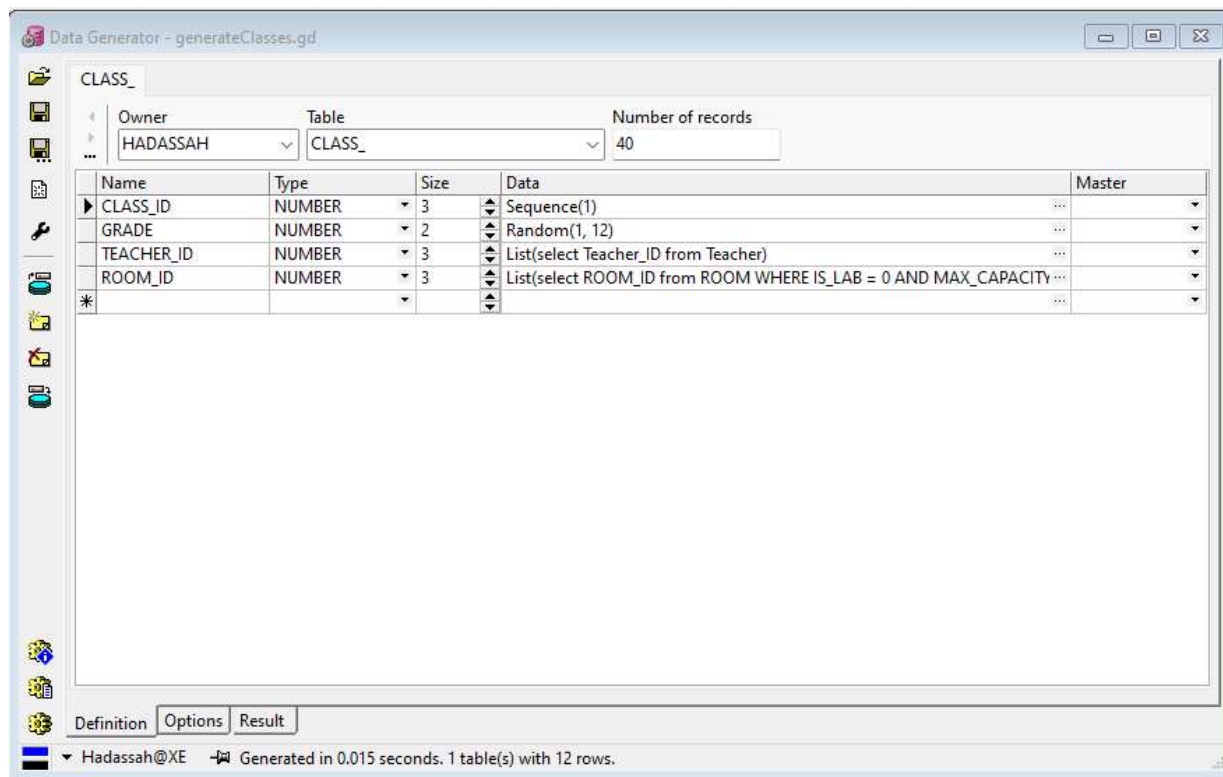
בחלק מהטבלאות קיימנו את האילוצים בשיטת "מצליח" והסתמכנו על העובדה שסטטיסטית מעט מאוד רשומות יפלו ביצירה. למשל בייצור של המידע בטבלת Lesson הסתמכנו על כך שסטטיסטית כאשר יש 6 ימים, 9 שעות לימוד ביום, 40 כיתות ו-400 מורים, הסיכוי שמורה או כיתה ישובצו לשתי שיעורים באותו יום באותה שעה, נמוך ביותר. במציאות באמת מתוך 450 רשומות שיצרנו, 400 היו חוקיות ונכנסו לטבלה.

השתדלנו לאכלס את ה-databases בנתונים ריאליסטיים ככל הניתן, ולכן:

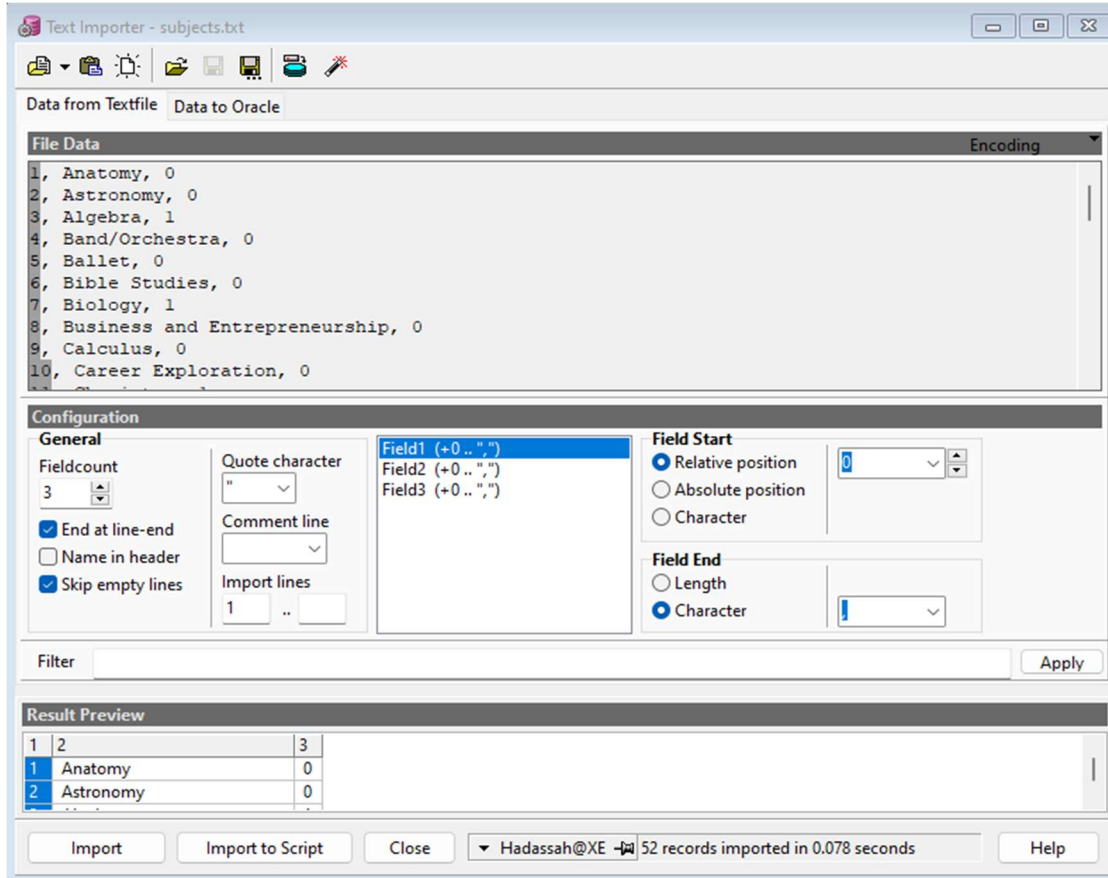
- ❖ בטבלה של Subject קיימות 52 רשומות בלבד. הכנסנו לשם כמעט כל מקצוע שניתן להעלות על הדעת שנלמד בבית ספר וגם קצת כאלו שככל הנראה לא...
- ❖ בטבלה של Class_ קיימות 40 רשומות בלבד. אם היינו רוצות למלא את הטבלה הזאת ב-400 רשומות אזי היה מתקיים תרחיש אחד מהשתיים:
 - היינו צריכות ליצור לפחות 6,000 תלמידים (ממוצע 15 בכיתה) וזה חורג מההנחיות שקיבלנו מהמרצה על מקסימום 1,000 רשומות.
 - בכל כיתה היה בממוצע תלמיד וחצי, וזה היה מקשה עלינו בהמשך הדרך להסיק מידע הגיוני ומועיל בשלבים של השאלות
- לכן לאחר שיקול מעמיק והסתמכות על הנחיות המרצה שלנו במצגת, החלטנו שעדיף ליצור מספר גדול של תלמידים ומספר קטן יחסית של כיתות.
- ❖ בשאר הטבלאות הכנסנו מעל 400 רשומות כנדרש

Data Generator

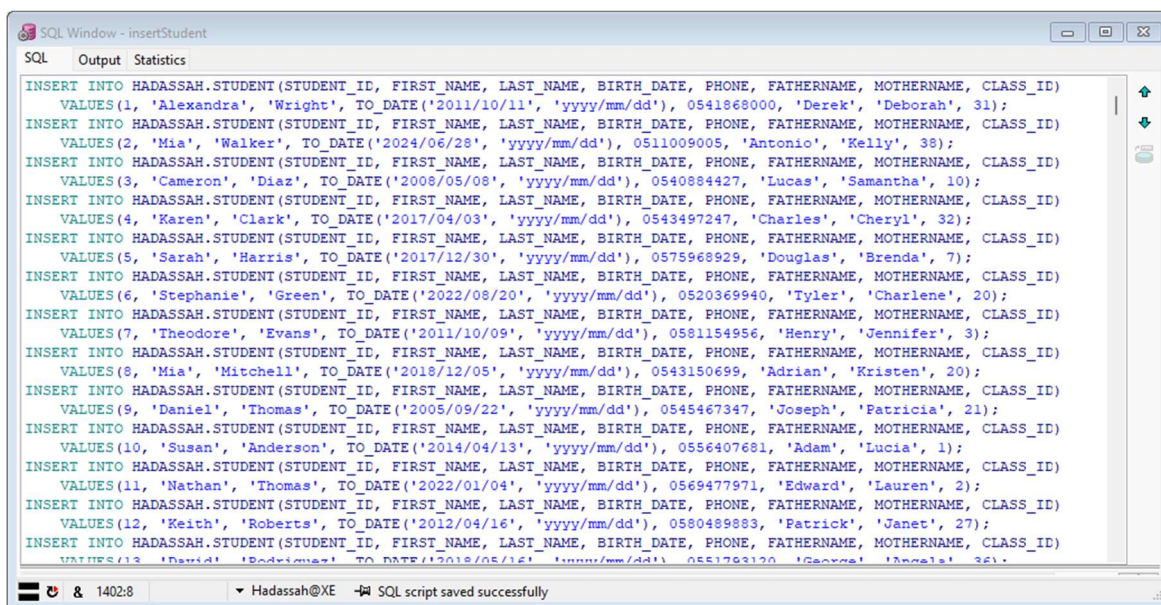




Data Importer



Programming



```

def random_first_name():
    rnd_gender = random.randint(0, 1)
    if (rnd_gender == 0):
        return male_names[random.randint(0, 96)]
    return female_names[random.randint(0, 111)]

def random_last_name():
    return last_names[random.randint(0, 169)]

def random_father_name():
    return male_names[random.randint(0, 96)]

def random_mother_name():
    return female_names[random.randint(0, 111)]

def random_b_date(start_year=2005, end_year=2024):
    """Generates a random date between start_year and end_year."""
    start_date = datetime(start_year, 1, 1)
    end_date = datetime(end_year, 12, 31)
    random_date = start_date + timedelta(days=random.randint(0, (end_date - start_date).days))
    return random_date.strftime('%Y/%m/%d')

def random_phone():
    """Generates a phone number in the format 05[1-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9]."""
    return f"05{random.randint(10000000, 99999999)}"

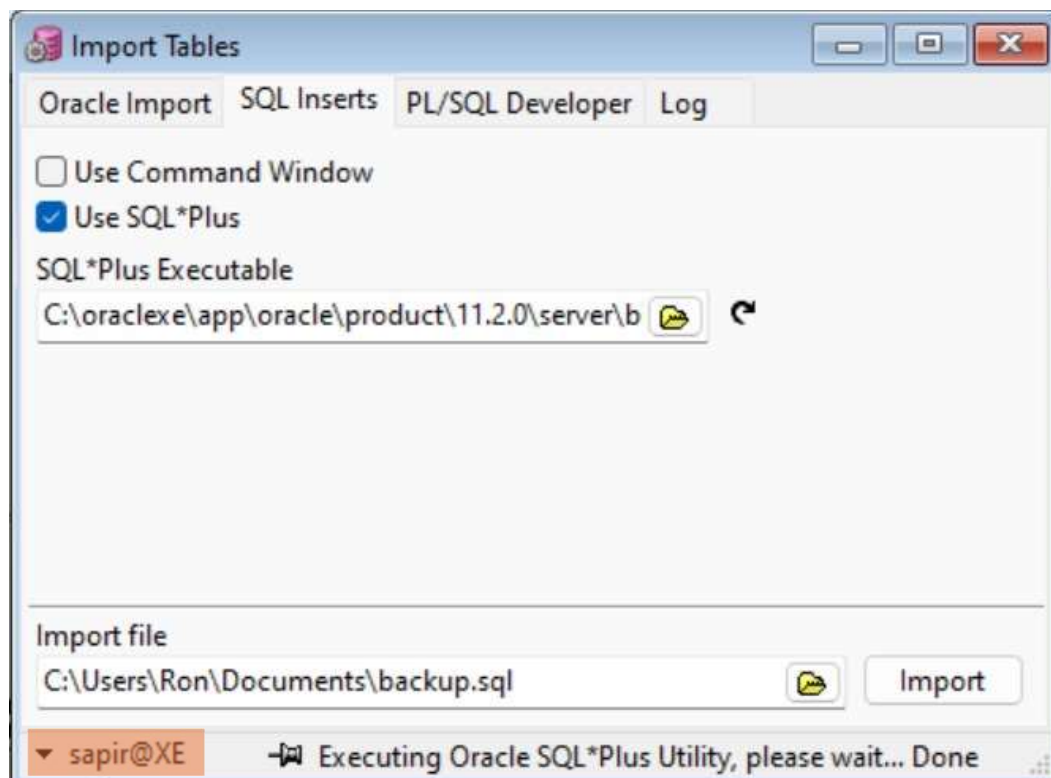
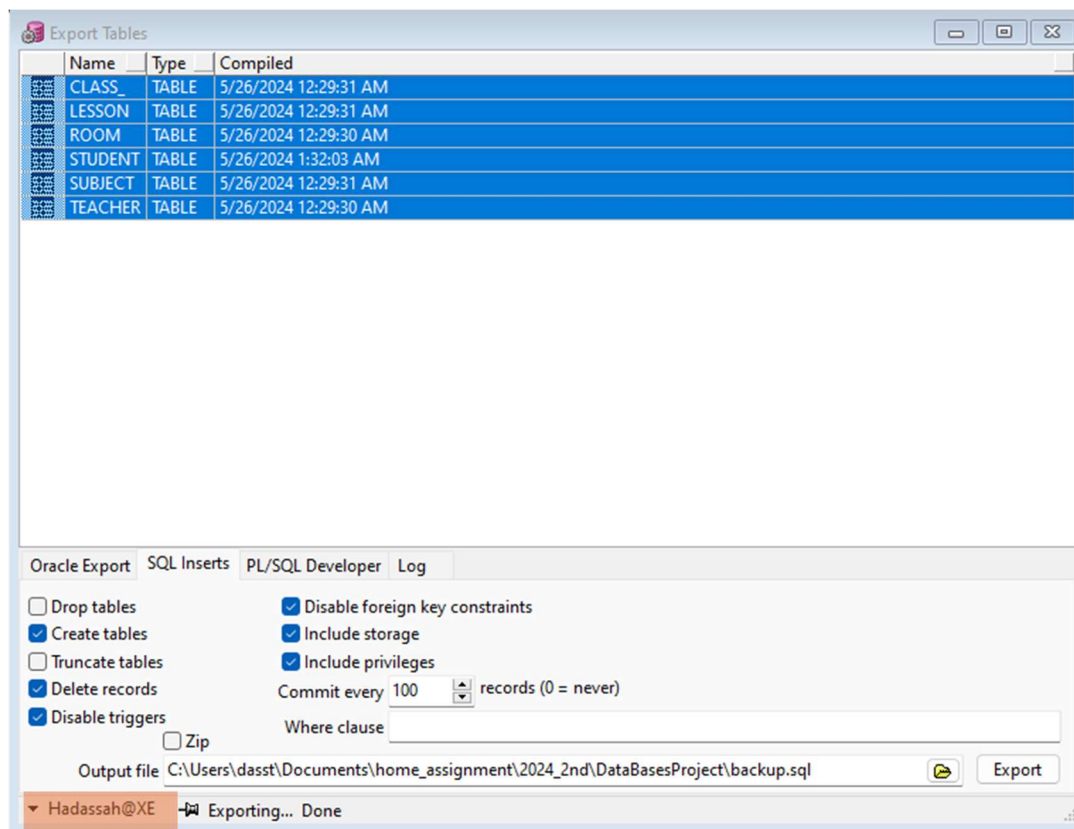
def random_class():
    return random.randint(1, 40)

def insert_statement():
    global id
    student_id = id
    id+=1
    first_name = random_first_name()
    last_name = random_last_name()
    birth_date = random_b_date()
    phone = random_phone()
    father = random_father_name()
    mother = random_mother_name()
    class_id = random_class()
    return f"""INSERT INTO HADASSAH.STUDENT(STUDENT_ID, FIRST_NAME, LAST_NAME,
    BIRTH_DATE, PHONE, FATHERNAME, MOTHERNAME, CLASS_ID)
    VALUES({student_id}, '{first_name}', '{last_name}', TO_DATE('{birth_date}', 'yyyy/mm/dd'),
    {phone}, '{father}', '{mother}', {class_id});"""

if __name__ == "__main__":
    with open("insert_commands.txt", "w") as file:
        for i in range(700):
            insert_list = insert_statement()
            file.write(f"{insert_list} \n")

```

גיבוי הנתונים ושיחזור



שאלות

שאלות Select

1. הצגת רשימת החדרים הפנויים שאין בהם כיתת אם, וגם עברו בדיקת תחזוקה אחרי 2021

SQL Window - selectEmptyRooms.sql

```
SQL
-- finds all empty rooms that had maintenance check since 2021,
select roo.Room_ID as "Empty Rooms", roo.last_maintenance_check AS "Last Check",
CASE WHEN roo.is_lab = 1 THEN 'Yes'
ELSE 'No'
END AS "Is Lab"
from Room roo left join class_cls
ON roo.Room_ID = cls.Room_ID
where cls.room_id is null
AND roo.last_maintenance_check > to_date('01/01/2021', 'dd/mm/yyyy')
```

	Empty Room	Last Check	Is Lab
1	2	11/27/2022	No
2	3	3/7/2023	No
3	4	2/28/2024	Yes
4	6	4/20/2021	Yes
5	7	6/3/2024	Yes
6	8	11/8/2021	Yes
7	9	9/15/2023	Yes
8	11	12/7/2021	Yes
9	12	8/13/2023	Yes
10	13	6/4/2023	Yes
11	15	1/18/2024	No
12	16	5/14/2024	Yes
13	17	12/3/2024	Yes

2. הצגת כל המורים וכמות המקצועות השונים שהם מלמדים, ממוינים בסדר יורד

SQL Window - how_much_lesson_teacher_teach.sql

```
SQL
-- for each teacher counts the amount of different subject he teaches and display from ost to least
SELECT t.Teacher_ID, t.TFirst_Name, t.TLast_Name , top_teacher.subject_count
FROM (SELECT Teacher_ID, COUNT(DISTINCT Subject_ID) AS Subject_Count
FROM Lesson
GROUP BY Teacher_ID )
top_teacher

JOIN Teacher t ON t.Teacher_ID = top_teacher.Teacher_ID
ORDER BY Subject_Count desc;
```

	TEACHER_ID	TFIRST_NAME	TLAST_NAME	SUBJECT_COUNT
1	41	Hal	Stevens	4
2	94	Eliza	McLachlan	4
3	147	Scarlett	Berry	4
4	100	Lauren	Dourif	4
5	239	Diane	Gandolfini	4
6	326	Larnelle	Downey	4

Hadassah@XE 249 rows selected in 0.266 seconds

3. לכל כיתה, הצגת המורים המלמדים אותה, וכמה שעות בשבוע

SQL Window - selectTeachersOfEachClass.sql

SQL Output Statistics

```
-- select teachers of each class and count amount of weekly lesson in that class
SELECT c.grade, c.class_id, t.tfirst_name||' '||t.tlast_name AS "Teacher Name", COUNT(l.lesson_id) AS lesson_count
FROM class_c
INNER JOIN lesson l ON c.class_id = l.class_id
INNER JOIN teacher t ON l.teacher_id = t.teacher_id
GROUP BY c.grade, c.class_id, t.tfirst_name, t.tlast_name
ORDER BY c.grade, c.class_id;
```

	GRADE	CLASS_ID	Teacher Name	LESSON_COUNT
18	1	15	Anthony Benet	1
19	1	15	Brian Gough	1
20	1	15	Ed Fehr	1
21	1	15	Joely LuPone	2
22	1	15	Larnelle Downey	1
23	1	15	Luis Cooper	1
24	1	15	Melba Bright	1

Hadassah@XE 388 rows selected in 0.656 seconds

4. לכל שכבה נספור ונציג כמה שיעורי חובה הם לומדים

SQL Window - selectandCountAmountOfMandatorySubjectPerClass.sql

SQL Output Statistics

```
-- query to select the grade and count of distinct mandatory subjects for each grade
SELECT Grade, COUNT(DISTINCT Subject_ID) AS Mandatory_Subject_Count
FROM (
    SELECT c.Grade AS Grade, s.Subject_ID
    FROM Lesson l
    JOIN Class_c ON l.Class_ID = c.Class_ID
    JOIN Subject s ON l.Subject_ID = s.Subject_ID
    WHERE
        s.mandatory = 1
) subquery
GROUP BY Grade
ORDER BY Grade;
```

	GRADE	MANDATORY_SUBJECT_COUNT
1	1	13
2	2	18
3	3	4
4	4	8
5	5	13
6	6	3
7	7	7

Hadassah@XE 12 rows selected in 0.093 seconds

שאלות Update

1. עדכון תאריך בדיקת תחזוקה לכל החדרים שהבדיקה האחרונה בהם הייתה בשנת 2022. תאריך הבדיקה יתעדכן לתאריך הנוכחי.

```
SQL Window - -- Update the maintenance check date for rooms that the last maintenance check was in 2022 UPDATE ...
SQL      Output      Statistics

-- Update the maintenance check date for rooms that the last maintenance check was in 2022
UPDATE Room
SET Last_maintenance_check = SYSDATE
WHERE EXTRACT (YEAR FROM Last_maintenance_check) = 2022;

commit;
```

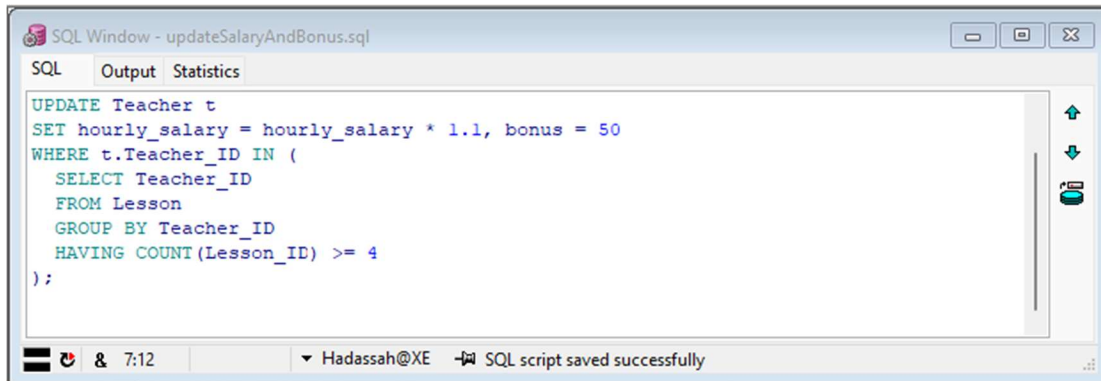
הטבלה לפני השאילתה:

Select room	Select class	Select teacher	Select subject	Select lesson	Select student
ROOM_ID	MAX_CAPACITY	IS_LAB	LAST_MAINTENANCE_CHECK		
1	1	62	1	6/22/2020	
2	2	21	0	11/27/2022	
3	3	55	0	3/7/2023	
4	4	63	1	2/28/2024	
5	5	63	0	9/13/2022	
6	6	66	1	4/20/2021	
7	7	69	1	6/3/2024	

הטבלה לאחר השאילתה:

Select room	Select class	Select teacher	Select subject	Select lesson	Select student
ROOM_ID	MAX_CAPACITY	IS_LAB	LAST_MAINTENANCE_CHECK		
1	1	62	1	6/22/2020	
2	2	21	0	6/16/2024	
3	3	55	0	3/7/2023	
4	4	63	1	2/28/2024	
5	5	63	0	6/16/2024	
6	6	66	1	4/20/2021	
7	7	69	1	6/3/2024	
8	8	64	1	11/8/2021	

2. עדכון משכורת ובונוס למורה על פי כמות שיעורים שהוא מוסר בשבוע. מורה שמוסר 4 שיעורי ומעלה, יקבל בונוס של 50 שח והעלאה של 10% לשעת עבודה



```

SQL Window - updateSalaryAndBonus.sql
SQL Output Statistics
UPDATE Teacher t
SET hourly_salary = hourly_salary * 1.1, bonus = 50
WHERE t.Teacher_ID IN (
  SELECT Teacher_ID
  FROM Lesson
  GROUP BY Teacher_ID
  HAVING COUNT(Lesson_ID) >= 4
);

```

7:12 Hadassah@XE SQL script saved successfully

הטבלה לפני השאילתה:

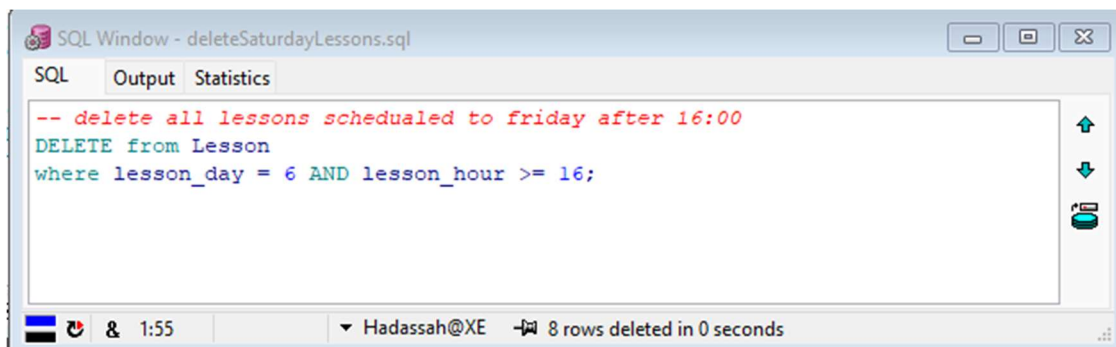
Select room	Select class_	Select teacher	Select subject	Select lesson	Select student
TEACHER_ID	TLAST_NAME	TFIRST_NAME	HOURLY_SALARY	BONUS	
32	37	Vanian	Ellen	35	0
33	38	O'Neal	Julianna	35	0
34	39	Griffin	Cameron	35	0
35	40	Lynne	Rita	35	0
36	41	Stevens	Hal	35	0
37	42	Hershey	Rhys	35	0
38	44	Navarro	Adrien	35	0
39	45	Cantrell	Sammy	35	0

הטבלה לאחר השאילתה:

Select room	Select class_	Select teacher	Select subject	Select lesson	Select student
TEACHER_ID	TLAST_NAME	TFIRST_NAME	HOURLY_SALARY	BONUS	
32	37	Vanian	Ellen	35	0
33	38	O'Neal	Julianna	35	0
34	39	Griffin	Cameron	35	0
35	40	Lynne	Rita	35	0
36	41	Stevens	Hal	38.5	50
37	42	Hershey	Rhys	35	0
38	44	Navarro	Adrien	35	0
39	45	Cantrell	Sammy	35	0

שאלות Delete

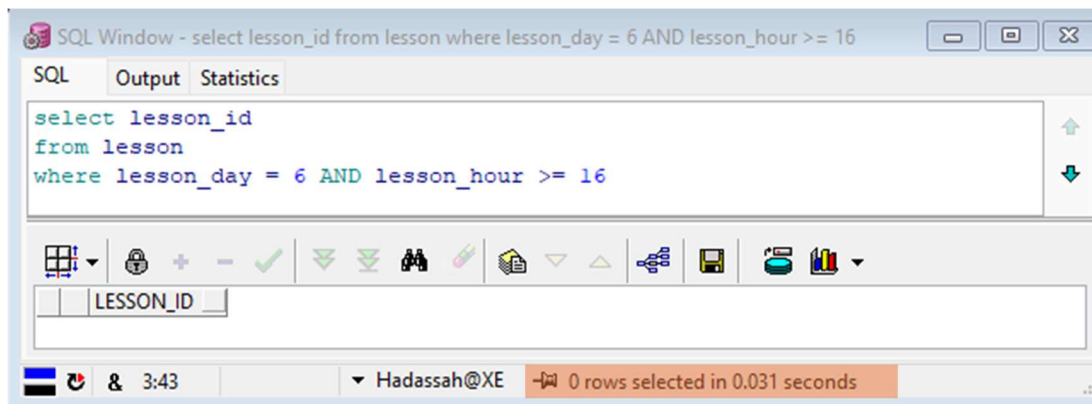
1. מחיקת כל השיעורים שנקבעו ליום שישי אחרי 16:00 בצהרים.



הטבלה לפני השאלתה:

Select room	Select class_	Select teacher	Select subject	Select lesson	Select student	
LESSON_ID	LESSON_DAY	LESSON_HOUR	CLASS_ID	TEACHER_ID	SUBJECT_ID	
395	443	3	15	38	266	42
396	444	4	16	18	167	41
397	446	6	10	36	15	5
398	448	2	10	23	51	15
399	449	3	11	10	73	38
400	450	1	15	8	350	30
401	480	6	19	3	150	5

אחרי השאלתה נוכיח שלא נותרו בטבלה שיעורים אחרי 16:00 ביום שישי:



2. מחיקת כל התלמידים שעל פי תאריך הלידה שלהם אינם מתאימים לגיל בית ספר

SQL Window - deletStudentBasedOnBirthday.sql

```

--delete all the student that have a invalid birthDate
DELETE FROM Student
WHERE birth_date > TO_DATE('2018-01-01', 'YYYY-MM-DD')
  OR birth_date < TO_DATE('2005-01-01', 'YYYY-MM-DD');

```

Hadassah@XE 264 rows deleted in 0.031 second:

הטבלה לפני השאלתה:

	STUDENT_ID	FIRST_NAME	LAST_NAME	BIRTH_DATE	PHONE	FATHERNAME	MOTHERNAME	CLASS_ID
1	1	Alexandra	Wright	10/11/2011	541868000	Derek	Deborah	
2	2	Mia	Walker	6/28/2024	511009005	Antonio	Kelly	
3	3	Cameron	Diaz	5/8/2008	540884427	Lucas	Samantha	
4	4	Karen	Clark	4/3/2017	543497247	Charles	Cheryl	
5	5	Sarah	Harris	12/30/2017	575968929	Douglas	Brenda	
6	6	Stephanie	Green	8/20/2022	520369940	Tyler	Charlene	
7	7	Theodore	Evans	10/9/2011	581154956	Henry	Jennifer	
8	8	Mia	Mitchell	12/5/2018	543150699	Adrian	Kristen	
9	9	Daniel	Thomas	9/22/2005	545467347	Insenh	Patricia	

אחרי השאלתה נוכיח שלא נשארו תלמידים שאינם מתאימים לגיל בית ספר:

SQL Window - SELECT student_id FROM Student WHERE birth_date > TO_DATE('2...

```

SELECT student_id
FROM Student
WHERE birth_date > TO_DATE('2018-01-01', 'YYYY-MM-DD')
  OR birth_date < TO_DATE('2005-01-01', 'YYYY-MM-DD');

```

Hadassah@XE 0 rows selected in 0.047 seconds

שאלות פרמטרים

1. מקבל כפרמטר שכבה בבית הספר וחודש, ומציג את רשימת התלמידים בשכבה שנולדו באותו חודש

```

SQL Window - parameter_selectStudentsInGradeAndBirthMonth.sql
SQL    Output    Statistics

--Show all the students how there is a birthday in a specific month and in a specific grade
SELECT Student_ID, First_Name, Last_Name, birth_date, c.Class_ID
FROM Student s left join class_ c on c.class_id = s.class_id
WHERE EXTRACT(MONTH FROM birth_date) = &<name="birth month">
      list= "1, January ,
            2, February,
            3, March,
            4, April,
            5, May,
            6, June,
            7, July,
            8, August,
            9, September,
            10, October,
            11, November,
            12, December"
      description="true" default="January" restricted="yes">
AND c.grade = &<name="grade Name" hint= "grade Name between (1-12)" type= "integer" >
order by EXTRACT(Day FROM s.birth_date);
  
```

החלונות הנפתחת:

בוחרים חודש בשנה ומצינים מספר שכבה

Name	Value
birth month	January
grade Name	5

OK Cancel Clear

grade Name between (1-12)

פלט השאילתה:

רשימת כל התלמידים מהשכבה שנולדו בחודש הנבחר ממויינים על פי תאריך

	STUDENT_ID	FIRST_NAME	LAST_NAME	BIRTH_DATE	CLASS_ID
1	224	Chad	Clark	1/3/2013	22
2	304	Claire	Diaz	1/11/2015	37
3	142	Kristen	Allen	1/14/2009	37
4	523	Dorothy	Clark	1/21/2006	37
5	425	Sarah	Scott	1/29/2014	25

Hadassah@XE 5 rows selected in 0.125 seconds

2. בהינתן יום ושכבה, נציג את המערכת של השכבה לאותו יום. המערכת כוללת עבור על שיעור את מספר הכיתה, שם ות.ז של המורה, מקצוע הנלמד בשיעור והשעה בה הוא מתקיים.

```
SQL Window - parameter_selectScheduleofGradeInDay.sql
SQL Output Statistics

-- Query to find the schedule of all classes in a certain grade on a chosen days
SELECT l.class_id, t.Teacher_ID, TFirst_Name || ' ' || TLast_Name as "Teacher name", subject_name, lesson_hour
FROM lesson l left join Teacher t on t.teacher_id = l.teacher_id
left join subject s on l.subject_id = s.subject_id
WHERE t.Teacher_ID IN (
  SELECT l.Teacher_ID
  FROM Lesson l
  inner join Class_ c ON l.Class_ID = c.Class_ID
  WHERE l.Lesson_Day = &<name = "Day"
        list = "1, Sunday,
                2, Monday,
                3, Tuesday,
                4, Wednesday,
                5, Thursday,
                6, Friday"
        description = "true" default = "Sunday" restricted="yes">
  AND c.grade = &<name="grade Name" hint= "grade Name between (1-12)" type= "integer">)
ORDER BY class_id;
```

החלונית הנפתחת:

בוחרים יום בשבוע ומזינים מספר שכבה

Name	Value
Day	Thursday
grade Name	8

OK Cancel Clear

grade Name between (1-12)

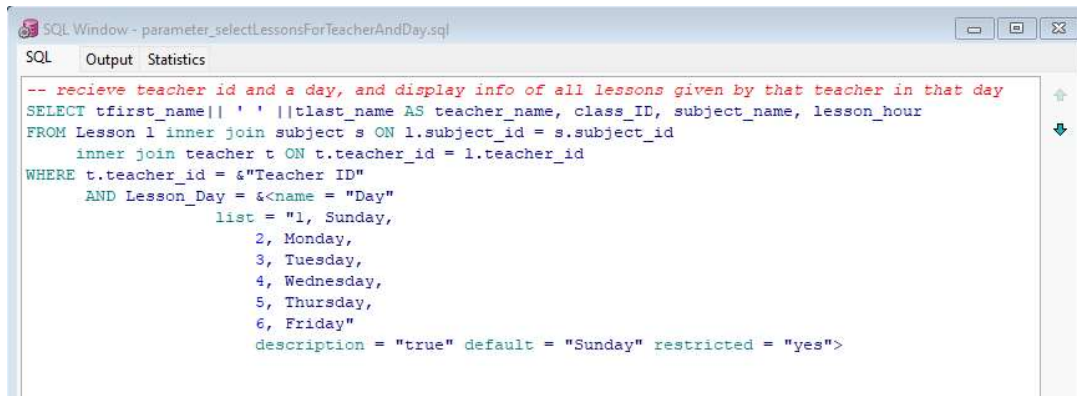
פלט השאילתה:

רשימת כל השיעורים שהשכבה הנבחרת לומדת ביום הנבחר

	CLASS_ID	TEACHER_ID	Teacher name	SUBJECT_NAME	LESSON_HOUR
1	4	150	Allan Shepard	Computer Science	15
2	8	398	Kathy Affleck	Health	14
3	9	88	Judi Yorn	Physical Education	13
4	12	366	Jean-Luc Dillon	Statistics	11
5	19	88	Judi Yorn	Drama	11
6	20	398	Kathy Affleck	Business and Entrepreneurship	10
7	20	150	Allan Shepard	Social Studies (General)	15
8	24	398	Kathy Affleck	World History	14

Hadassah@XE 13 rows selected in 0.22 seconds

3. השאילתה מציגה את המערכת של מורה שנבחר ביום מסוים.

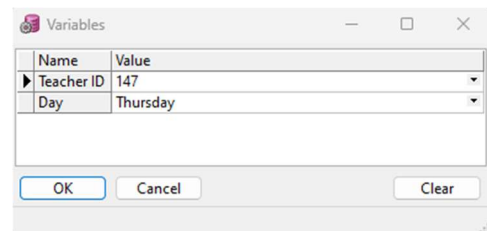


```
SQL Window - parameter_selectLessonsForTeacherAndDay.sql
SQL Output Statistics

-- recieve teacher id and a day, and display info of all lessons given by that teacher in that day
SELECT tfirst_name|| ' ' ||tlast_name AS teacher_name, class_ID, subject_name, lesson_hour
FROM Lesson l inner join subject s ON l.subject_id = s.subject_id
    inner join teacher t ON t.teacher_id = l.teacher_id
WHERE t.teacher_id = &"Teacher ID"
    AND Lesson_Day = &<name = "Day"
        list = "1, Sunday,
                2, Monday,
                3, Tuesday,
                4, Wednesday,
                5, Thursday,
                6, Friday"
        description = "true" default = "Sunday" restricted = "yes">
```

החלונית הנפתחת:

מזינים בחלונית את מספר התעודת זהות של המורה הרצוי, ובוחרים יום בשבוע

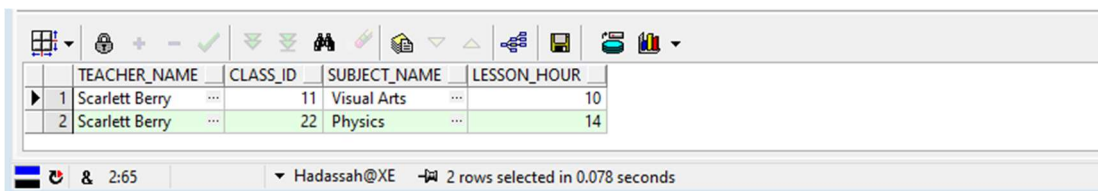


Name	Value
Teacher ID	147
Day	Thursday

OK Cancel Clear

פלט השאילתה:

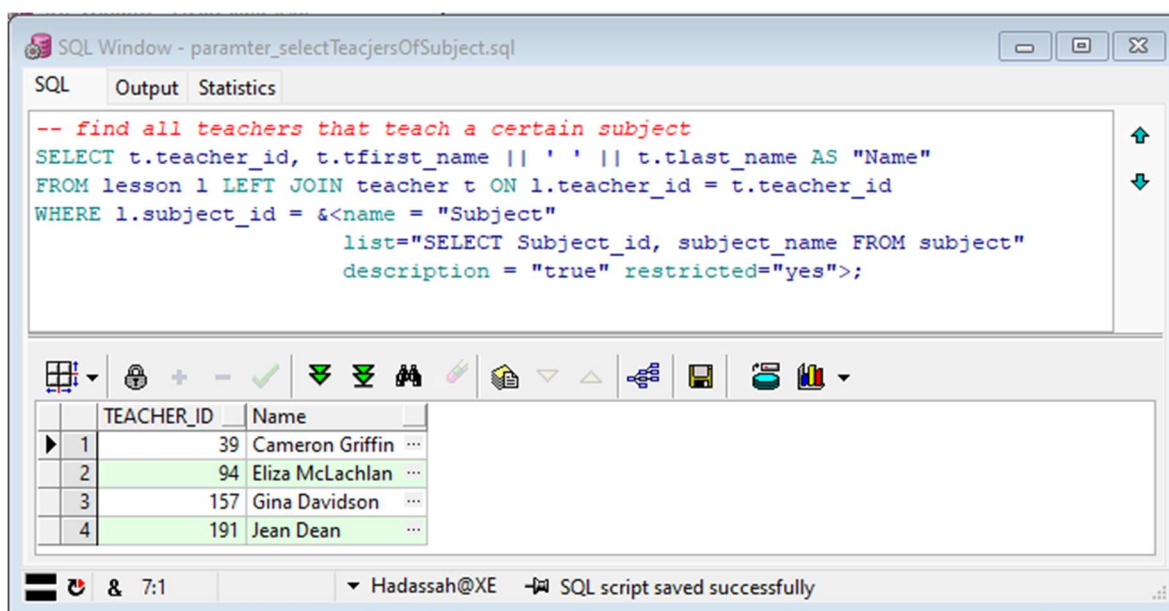
כל השיעורים שהמורה הנבחר מלמד באותו יום



	TEACHER_NAME	CLASS_ID	SUBJECT_NAME	LESSON_HOUR
1	Scarlett Berry	11	Visual Arts	10
2	Scarlett Berry	22	Physics	14

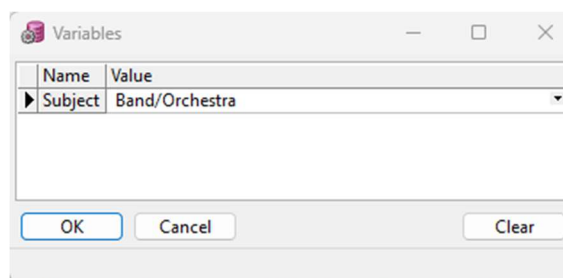
Hadassah@XE 2 rows selected in 0.078 seconds

4. השאילתה מציגה את כל המורים שמלמדים מקצוע מסוים



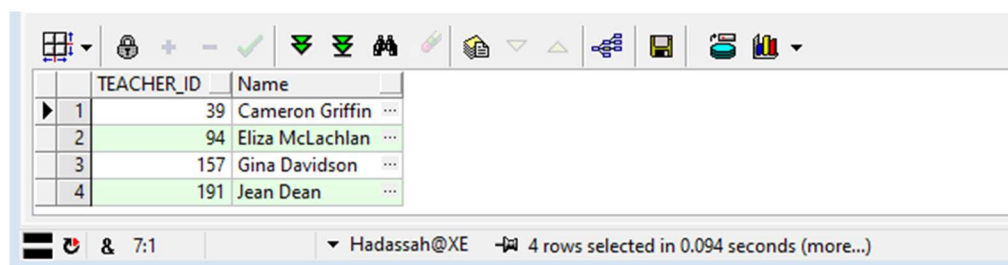
החלונית הנפתחת:

בוחרים מקצוע מרשימת המקצועות הנתונה



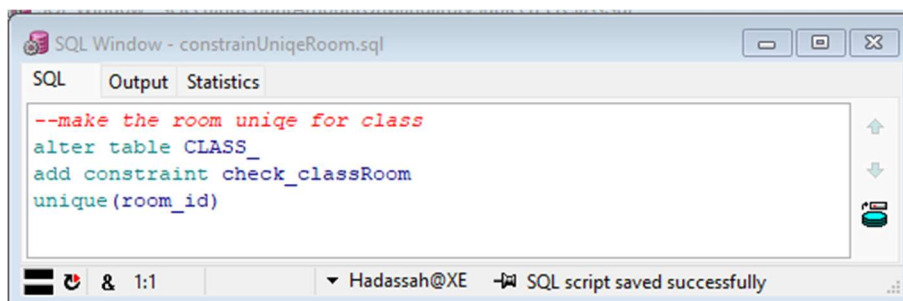
פלט השאילתה:

רשימת כל המורים המלמדים את אותו מקצוע

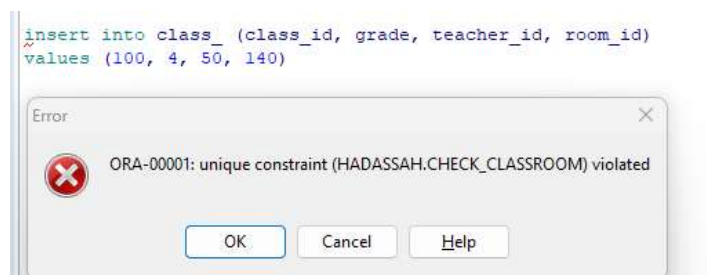


אילוצים

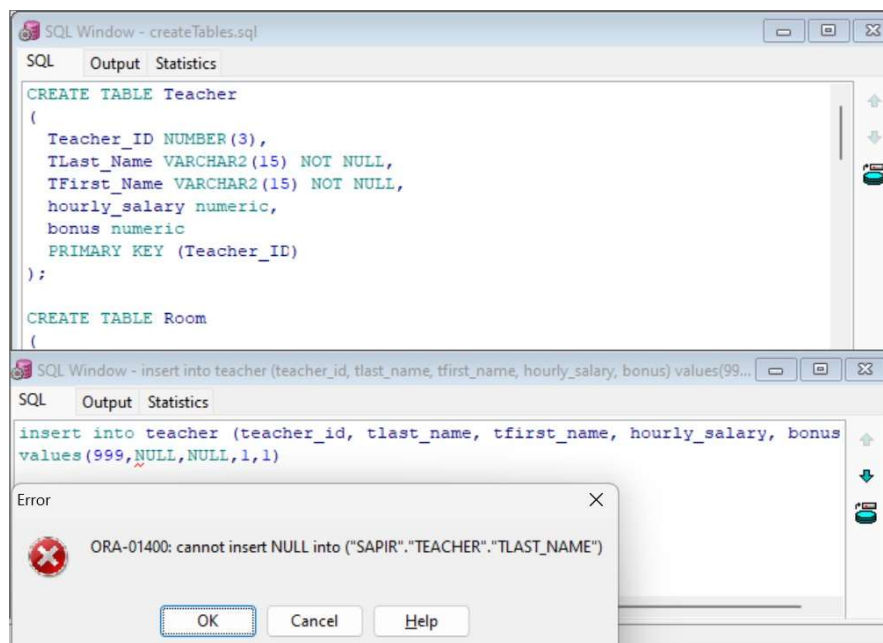
1. אילוץ Unique בטבלה Class_ עבור חדר הכיתה לומדת. לא יהיה מצב שבו שתי כיתות אם רשומת באותו חדר.



נסיון להזנת נתון שסותר את האילוץ (יש כבר כיתה שלומד בחדר 140)

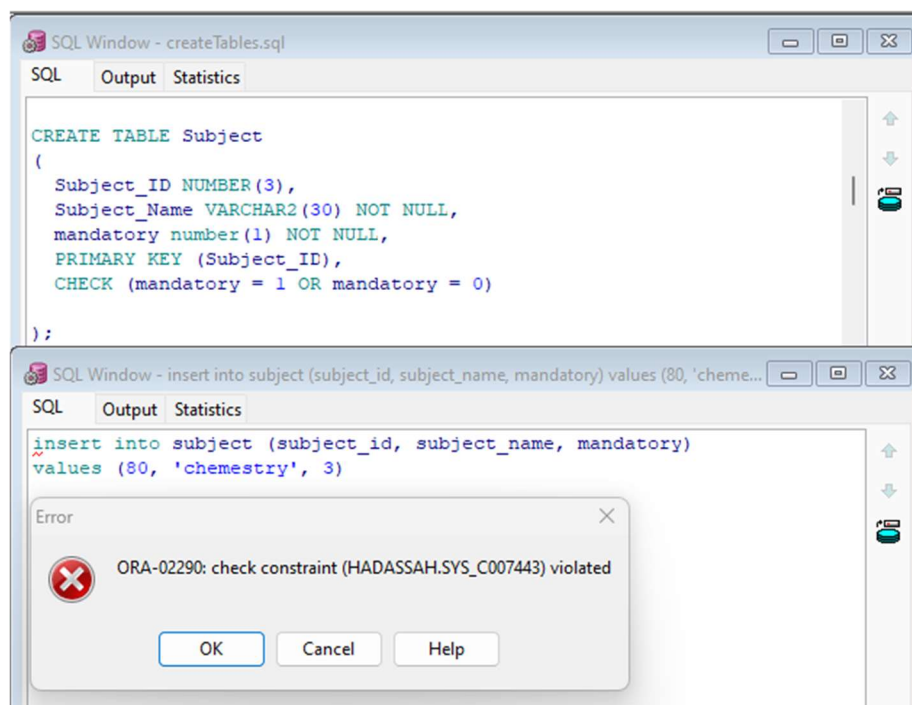


2. אילוץ NOT NULL בטבלה Teacher עבור שם המורה. לא ניתן להכניס לטבלה מורה בלי שם.



הערה: יצרנו את האילוץ בזמן יצירת הטבלה ולא על ידי שינוי הטבלה, מאושר על ידי המרצה שולמית.

3. אילוץ CHECK בטבלה Subject עבור התכונה Mandatory
התכונה יכולה לקבל מספר 1 או 0 ולא שום מספר אחר



הערה: יצרנו את האילוץ בזמן יצירת הטבלה ולא על ידי שינוי הטבלה, מאושר על ידי המרצה שולמית.