

מיני פרוייקט בסיסי נתונים

מרכז חוגים – שלב ד

אילה אפטרקוט 213136813 Tmunot1234567@gmail.com

נעמה שוייצר 325447654 Naamasch1@gmail.com

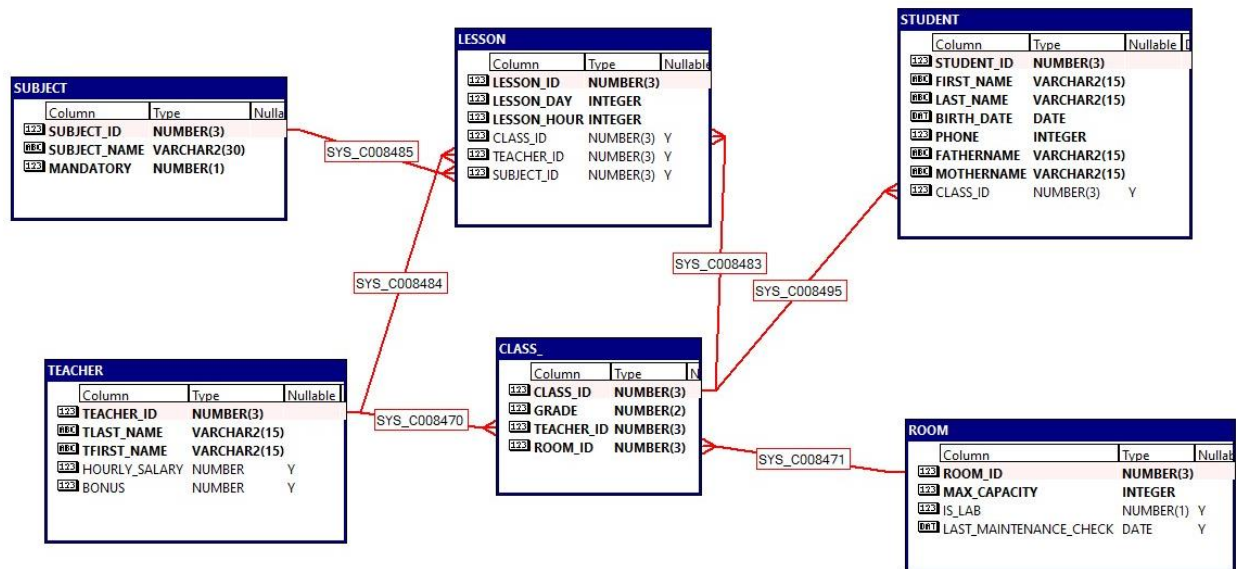
תשפ"ה

תוכן עניינים

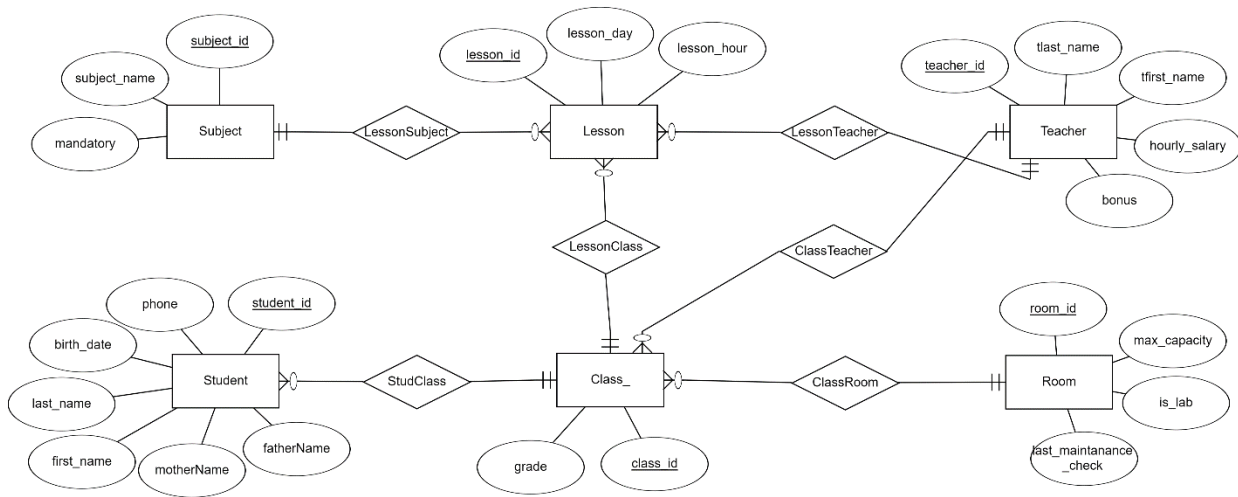
2.....	שלב ד: אינטגרציה ומבטים
2	DSD אגף חדש
3	ERD אגף חדש – בית ספר
4	ERD אגף ישן – מרכז חוגים
5	ERD משותף
6	DSD לאחר אינטגרציה
7	מבטים
11.....	מבט 1: אגף ישן (מרכז סדנאות)
12.....	שאלתא על מבט 1
13.....	מבט 2: אגף חדש (בית ספר)
15.....	שאלתא על מבט 2

שלב ד: אינטגרציה ומבטים

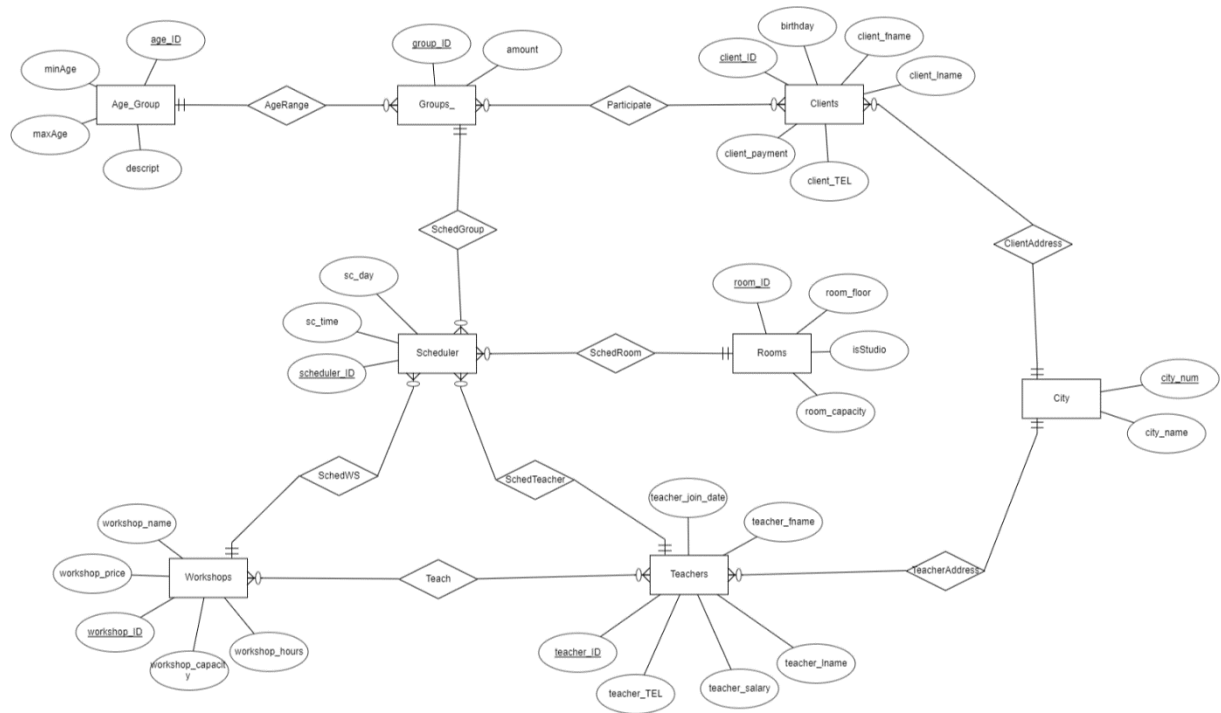
DSD אגף חדש



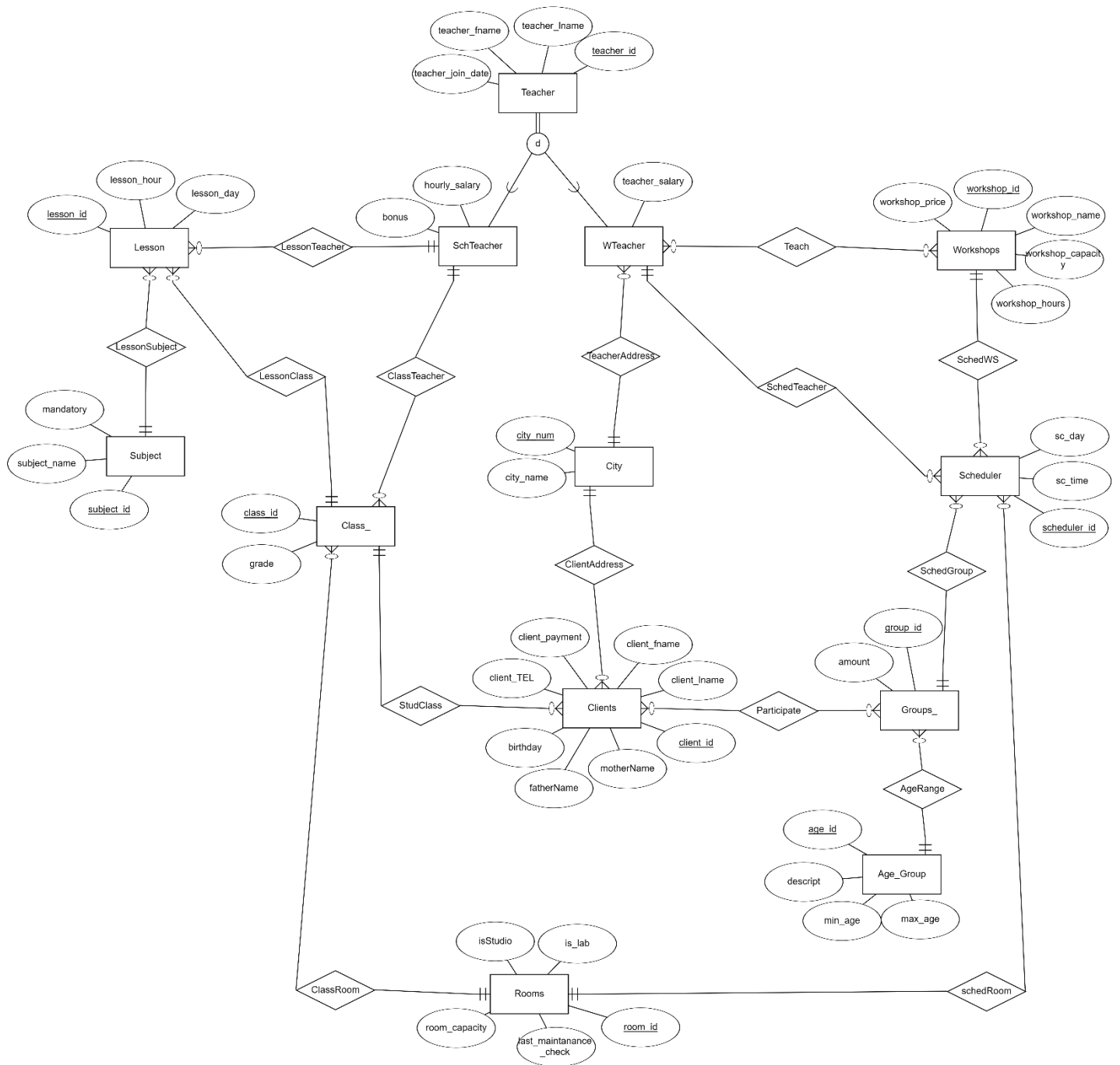
ERD אגף חדש – בית ספר



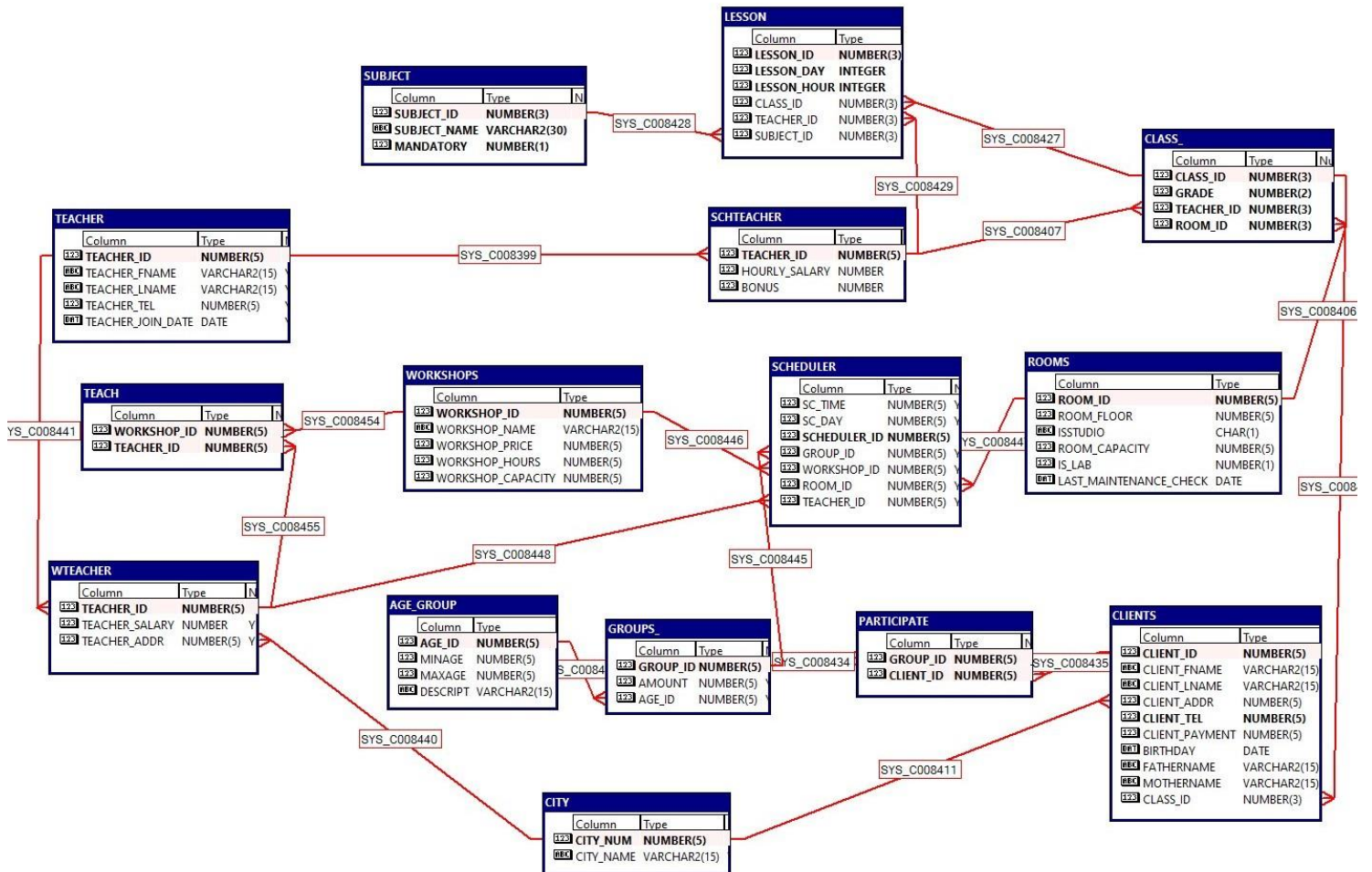
ERD אגף ישן – מרכז חוגים



ERD משותף



DSD לאחר אינטגרציה



אינטגרציה

הערה: נשמש לקיצור SCH במקום האגף בית ספר, וW עבור אגף הסדנאות.

החלטות שנעשו בשלב האינטגרציה

- עבור SCH, W.Teachers
בגלל החפיפה, החלטנו ליצור ישות חדשה Teacher עם התכונות המשותפות וממנה ירשו schteacher עם התכונות היחודיות למורה בי"ס wteacher עם התכונות היחודיות למורה של סדנא.
הסיבה שהחלטנו לא לאחד אותם היא כדי שכל מורה יוכל לשמור על היחודיות שלו ועל הקשרים האופייניים לו (לא כל מורה סדנא הוא בהכרח מורה בי"ס וכן להיפך).
- עבור SCH, W.Clients
בגלל החפיפה, החלטנו לאחד את הישויות לישות Clients. את התכונות הלא משותפות הוספנו כNullables כדי לא לחייב סטודנט בי"ס לקחת סדנא או לחייב לקוח של מרכז הסדנאות להיות תלמיד בי"ס.
- עבור SCH, W.Rooms
בגלל החפיפה, החלטנו לאחד את הישויות לישות אחת Rooms. את התכונות הלא משותפות, עבור התכונות הבוליאניות – האם החדר מעבדה/האם החדר סטודיו – הוספנו אותם עם ערכי ברירת מחדל של FALSE. את התכונות הלא בוליאניות הוספנו כNullables.
הביטוי העיקרי של שילוב המערכות מתבטא כאן; כאשר שיעורי בי"ס ושיעורי סדנאות משתמשים באותם חדרים.
שאר הטבלאות נותרו כמו באגפים המקוריים.

הסבר מילולי של התהליך והפקודות

Rooms

הוספנו לטבלה המקורית שלנו עמודות חדשות: is_lab, last_maintenance_check:

```
-- Adding new columns to the ROOMS table
alter table ROOMS add is_lab number(1);
alter table ROOMS add last_maintenance_check date;
```

העברת המידע:

בדקנו אם החדר קיים במערכת.

```
-- This block transfers data from ROOM1 to ROOMS, updating existing records or inserting new ones
DECLARE
    v_count NUMBER;
BEGIN
    -- Loop through ROOM1 records
    FOR r IN (SELECT * FROM ROOM1)
    LOOP
        -- Check if room exists in ROOMS
        SELECT COUNT(*) INTO v_count
        FROM ROOMS
        WHERE ROOM_ID = r.ROOM_ID;

        IF v_count > 0 THEN
            -- Update existing room
            UPDATE ROOMS
            SET IS_LAB = r.IS_LAB,
                LAST_MAINTENANCE_CHECK = r.LAST_MAINTENANCE_CHECK,
                ROOM_CAPACITY = LEAST(ROOM_CAPACITY, r.MAX_CAPACITY),
                ROOM_FLOOR = COALESCE(ROOM_FLOOR, NULL)
            WHERE ROOM_ID = r.ROOM_ID;
        ELSE
            -- Insert new room
            INSERT INTO ROOMS (ROOM_ID, IS_LAB, LAST_MAINTENANCE_CHECK, ROOM_CAPACITY, ROOM_FLOOR, ISSTUDIO)
            VALUES (r.ROOM_ID, r.IS_LAB, r.LAST_MAINTENANCE_CHECK, r.MAX_CAPACITY, NULL, 'F');
        END IF;
    END LOOP;

    COMMIT;

    DBMS_OUTPUT.PUT_LINE('Update completed successfully..');
EXCEPTION
    WHEN OTHERS THEN
        DBMS_OUTPUT.PUT_LINE('Error occurred: ' || SQLERRM);
        ROLLBACK;
END;
```

אם כן, הוספנו לחדר הקיים את המידע העדכני.

עבור תכונת קיבולת של חדר בחרנו את הערך הנמוך מבין השניים.

```
ELSE
    -- Insert new room
    INSERT INTO ROOMS (ROOM_ID, IS_LAB, LAST_MAINTENANCE_CHECK, ROOM_CAPACITY, ROOM_FLOOR, ISSTUDIO)
    VALUES (r.ROOM_ID, r.IS_LAB, r.LAST_MAINTENANCE_CHECK, r.MAX_CAPACITY, NULL, 'F');
END IF;
END LOOP;

COMMIT;

DBMS_OUTPUT.PUT_LINE('Update completed successfully..');
EXCEPTION
    WHEN OTHERS THEN
        DBMS_OUTPUT.PUT_LINE('Error occurred: ' || SQLERRM);
        ROLLBACK;
END;
```

אם החדר לא קיים במערכת, הוספנו אותו כאשר ערך ברירת המחדל עבור "האם סטודיו" הוא 'F') (FALSE).

Teacher

יצרנו 3 ישויות חדשות:

- Teacher עם התכונות המשותפות עבור המורים ב2 המערכות (כמפורט מקודם).
- schteacher שיורש מ Teacher, עם התכונות היחודיות עבור מורי בי"ס.
- wteacher שיורש מ Teacher, עם התכונות היחודיות והקשרים היחודיים עבור מורי סדנאות.

```
-- Teacher integration
-- Creating new tables for teacher data
CREATE TABLE TEACHER (
    TEACHER_ID NUMBER(5) PRIMARY KEY,
    TEACHER_FNAME VARCHAR2(15),
    TEACHER_LNAME VARCHAR2(15),
    TEACHER_TEL NUMBER(5),
    TEACHER_JOIN_DATE DATE
);
```

```
CREATE TABLE SCHTEACHER (
    TEACHER_ID NUMBER(5) PRIMARY KEY,
    HOURLY_SALARY NUMBER,
    BONUS NUMBER,
    FOREIGN KEY (TEACHER_ID) REFERENCES TEACHER(TEACHER_ID)
);

CREATE TABLE WTEACHER (
    TEACHER_ID NUMBER(5) PRIMARY KEY,
    TEACHER_SALARY NUMBER,
    TEACHER_ADDR NUMBER(5),
    FOREIGN KEY (TEACHER_ADDR) REFERENCES CITY(CITY_NUM),
    FOREIGN KEY (TEACHER_ID) REFERENCES TEACHER(TEACHER_ID)
);
```

העברת המידע:

הכנסנו את המורים שקיימים אצלינו (מורי הסדנאות) ואת המורי בי"ס שלא קיימים אצלינו לTeacher. עדכנו רק את התכונות המשותפות ל2 סוגי המורים (רק את התכונות של Teacher).

מילאנו את הטבלאות של מורי בי"ס ומורי סדנא (התכונות הלא משותפות ל2 סוגי המורים) בעזרת הטבלאות המקוריות.

```
DECLARE
    v_count NUMBER;
BEGIN
    -- Transfer data to TEACHER table
    INSERT INTO TEACHER (TEACHER_ID, TEACHER_FNAME, TEACHER_LNAME, TEACHER_TEL, TEACHER_JOIN_DATE)
    SELECT TEACHER_ID, TEACHER_FNAME, TEACHER_LNAME, TEACHER_TEL, TEACHER_JOIN_DATE
    FROM TEACHERS
    WHERE TEACHER_ID IS NOT NULL;

    -- Add teachers from TEACHER1 if they don't exist
    FOR rec IN (SELECT * FROM TEACHER1)
    LOOP
        SELECT COUNT(*) INTO v_count
        FROM TEACHER
        WHERE TEACHER_ID = rec.TEACHER_ID;

        IF v_count = 0 THEN
            INSERT INTO TEACHER (TEACHER_ID, TEACHER_FNAME, TEACHER_LNAME)
            VALUES (rec.TEACHER_ID, rec.TFIRST_NAME, rec.TLAST_NAME);
        END IF;
    END LOOP;

    -- Transfer data to SCHTEACHER
    INSERT INTO SCHTEACHER (TEACHER_ID, HOURLY_SALARY, BONUS)
    SELECT TEACHER_ID, HOURLY_SALARY, BONUS
    FROM TEACHER1
    WHERE TEACHER_ID IS NOT NULL;

    -- Transfer data to WTEACHER
    INSERT INTO WTEACHER (TEACHER_ID, TEACHER_SALARY, teacher_addr)
    SELECT TEACHER_ID, TEACHER_SALARY, teacher_addr
    FROM TEACHERS
    WHERE TEACHER_ID IS NOT NULL;
```

Client

הוספנו לטבלה המקורית שלנו עמודות חדשות: fathername, mothername ומפתח זר: class_id.

```
alter table CLIENTS add fathername VARCHAR2(15);
alter table CLIENTS add mothername VARCHAR2(15);
alter table CLIENTS add class_id NUMBER(3);
alter table CLIENTS
  add foreign key (CLASS_ID)
  references class1 (CLASS_ID);
```

העברת המידע:

```
-- Update existing clients with data from STUDENT1
UPDATE CLIENTS C
SET C.FATHERNAME = (SELECT S.FATHERNAME FROM STUDENT1 S WHERE S.STUDENT_ID = C.CLIENT_ID),
    C.MOTHERNAME = (SELECT S.MOTHERNAME FROM STUDENT1 S WHERE S.STUDENT_ID = C.CLIENT_ID),
    C.CLASS_ID = (SELECT S.CLASS_ID FROM STUDENT1 S WHERE S.STUDENT_ID = C.CLIENT_ID)
WHERE EXISTS (SELECT 1 FROM STUDENT1 S WHERE S.STUDENT_ID = C.CLIENT_ID);
commit;
```

עבור לקוח שהוא סטודנט (ערכים משותפים): עידכנו את הלקוחות הקיימים עם הנתונים הנוספים של סטודנט.

```
-- Insert new students as clients
INSERT INTO CLIENTS (
  CLIENT_ID,
  CLIENT_FNAME,
  CLIENT_LNAME,
  CLIENT_ADDR,
  CLIENT_TEL,
  CLIENT_PAYMENT,
  BIRTHDAY,
  FATHERNAME,
  MOTHERNAME,
  CLASS_ID
)
SELECT
  S.STUDENT_ID,
  S.FIRST_NAME,
  S.LAST_NAME,
  (SELECT MIN(CITY_NUM) FROM CITY), -- Default value for CLIENT_ADDR
  MOD(S.PHONE, 100000), -- convert to fit CLIENT.Client_TEL$type
  0, -- Default value for CLIENT_PAYMENT
  S.BIRTH_DATE,
  S.FATHERNAME,
  S.MOTHERNAME,
  S.CLASS_ID
FROM STUDENT1 S
WHERE NOT EXISTS (SELECT 1 FROM CLIENTS C WHERE C.CLIENT ID = S.STUDENT ID);
```

הוספנו למערכת את הסטודנטים של בי"ס שלא קיימים במערכת כלקוחות עם ערכי ברירת מחדל מתאימים עבור תכונות שיחודיות ללקוח.

התאמת הקשרים למבנה החדש:

- בLESSON: העברנו את מקור המפתח הזר Teacher_id להיות הטבלה החדשה של מורה ב"ס SCHTEACHER.
- בCLASS: העברנו את מקור המפתח הזר Teacher_id להיות הטבלה החדשה של מורה ב"ס SCHTEACHER.
- בCLASS: העברנו את מקור המפתח הזר Room_id להיות הטבלה המאוחדת Rooms.
- בTEACH: העברנו את מקור המפתח הזר Teacher_id להיות הטבלה החדשה של מורה סדנא WTEACHER.
- בSCHEDULER: העברנו את מקור המפתח הזר Teacher_id להיות הטבלה החדשה של מורה סדנא WTEACHER.

מחיקת טבלאות שאינן בשימוש במבנה החדש:

מחקנו את הטבלאות: STUDENT, ROOM, את TEACHER של ב"ס ואת TEACHERS של מרכז הסדנאות.

מבטים

מבט 1: אגף ישן (מרכז סדנאות)

תיאור: לוח זמנים שבועי מנקודת מבט של מרכז סדנאות.

יצירת המבט:

```

--- our view: workshops
CREATE OR REPLACE VIEW scheduled_workshops_view AS
SELECT
    s.scheduler_ID,
    w.workshop_name,
    t.teacher_fname || ' ' || t.teacher_lname AS teacher_name
    r.room_ID,
    r.room_floor,
    r.room_capacity,
    g.group_ID,
    ag.descript AS age_group,
    s.sc_day,
    s.sc_time,
    g.amount AS participants,
    w.workshop_capacity
FROM
    Scheduler s
    JOIN Workshops w ON s.workshop_ID = w.workshop_ID
    JOIN Teacher t ON s.teacher_ID = t.teacher_ID
    JOIN Rooms r ON s.room_ID = r.room_ID
    JOIN Groups g ON s.group_ID = g.group_ID
    JOIN Age_Group ag ON g.age_ID = ag.age_ID
WHERE T.TEACHER ID IN (SELECT TEACHER ID FROM WTEACHER);

```

Done in 0.016 seconds

שליפת נתונים:

SELECT * FROM scheduled_workshops_view;

SCHEDULER_ID	WORKSHOP_NAME	TEACHER_NAME	ROOM_ID	ROOM_FLOOR	ROOM_CAPACITY	GROUP_ID	AGE_GROUP	SC_DAY	SC_TIME	PARTICIPANTS	WORKSHOP_CAPACITY
1	5 Ballet	Gilberto Satriani	7	3	33	5	Adults	4	17	1	27
2	190 Ballet	Seth Ford	1	4	28	156	Adults	1	6	24	27
3	269 Cooking	Giovanni Vince	2	1	21	206	Pensioners	1	6	22	34
4	321 Cooking	Liev Burns	7	3	33	498	Pensioners	1	6	33	34
5	1695 Sewing	Patrica Morhead	472	0	39	284	Pensioners	2	9	15	22
6	1697 Drawing	Diane Chan	345	4	14	23	Adults	2	20	4	10
7	1698 Ballet	Steven Tomlin	365	2	35	157	Pensioners	6	22	9	27
8	1701 Judo	Lizzy Posener	87	2	20	338	Kids	4	17	6	19
9	1702 Zumba	Connie Coburn	268	3	18	494	Pensioners	2	10	3	12
10	1707 Zumba	Andrea Hawn	478	3	14	144	Pensioners	6	22	1	12
11	1709 Baking	Natacha Spears	5	1	23	416	Adults	1	6	15	29

18 rows selected in 0.063 seconds (more...)

שאילתא 1 על מבט 1

תיאור: הצגת 10 השעות השבועיות העמוסות ביותר ומספר הסדנאות באותן שעות.

הקוד:

```
-- This query finds the top 10 busiest days and times for workshops
-- It counts the number of sessions for each day and time combination
-- and orders them in descending order of session count
SELECT *
FROM (
    SELECT s.sc_day, s.sc_time, COUNT(*) AS session_count
    FROM scheduled_workshops_view s
    GROUP BY s.sc_day, s.sc_time
    ORDER BY session_count DESC
)
WHERE ROWNUM <= 10;
```

תוצאת הרצה:

	SC_DAY	SC_TIME	SESSION_COUNT
1	1	6	72
2	1	7	9
3	3	10	7
4	4	20	6
5	4	17	6
6	6	22	5
7	4	8	5
8	2	19	5
9	1	22	5
10	2	24	5

10 rows selected in 0.032 seconds

שאילתא 2 על מבט 1

תיאור: הצגת הסדנאות הפופולריות/ עם הכי הרבה נרשמים.

הקוד:

```
-- the most attended/popular workshops
SELECT WORKSHOP_NAME, COUNT(*) AS TOTAL_SESSIONS,
    SUM(PARTICIPANTS) AS TOTAL_PARTICIPANTS,
    ROUND(AVG(PARTICIPANTS), 2) AS AVG_PARTICIPANTS_PER_SESSION
FROM SCHEDULED_WORKSHOPS_VIEW
GROUP BY WORKSHOP_NAME
ORDER BY TOTAL_PARTICIPANTS DESC;
```

תוצאת הרצה:

Select select Select scheduled_workshops_view

	WORKSHOP_NAME	TOTAL_SESSIONS	TOTAL_PARTICIPANTS	AVG_PARTICIPANTS_PER_SESSION
1	Cooking	48	779	16.23
2	Baking	52	771	14.83
3	Ballet	38	555	14.61
4	Sewing	46	528	11.48
5	Judo	46	480	10.43
6	Photography	32	312	9.75
7	Zumba	39	270	6.92
8	Swimming	26	188	7.23
9	Gardening	24	134	5.58
10	Drawing	21	120	5.71
11	Guitar	8	19	2.38
12	Singing	2	3	1.5
13	Piano	2	2	1
14	Violin	2	2	1

80:1 ayala@xe 14 rows selected in 0.031 seconds

מבט 2: אגף חדש (בית ספר)

תיאור: לוח זמנים שבועי מנקודת מבט של בית ספר.

יצירת המבט:

```
--other point of view- school
CREATE OR REPLACE VIEW LESSON_DETAILS_VIEW AS
SELECT
    L.LESSON_ID,
    L.LESSON_DAY,
    L.LESSON_HOUR,
    L.CLASS_ID,
    C.GRADE,
    L.TEACHER_ID,
    T.TEACHER_FNAME,
    L.SUBJECT_ID,
    S.SUBJECT_NAME,
    C.ROOM_ID,
    R.ROOM_CAPACITY,
    R.IS_LAB
FROM
    LESSON L
    LEFT JOIN CLASS C ON L.CLASS_ID = C.CLASS_ID
    LEFT JOIN TEACHER T ON L.TEACHER_ID = T.TEACHER_ID
    LEFT JOIN SUBJECT S ON L.SUBJECT_ID = S.SUBJECT_ID
    LEFT JOIN ROOMS R ON C.ROOM_ID = R.ROOM_ID
WHERE T.TEACHER_ID IN (SELECT TEACHER_ID FROM SCHTEACHER);
```

Done in 0.031 seconds

שליפת נתונים:

```
SELECT * FROM LESSON_DETAILS_VIEW;
```

Create lesson_details_view

Select lesson_details_view

Select lesson_details_view

Create scheduled_workshops_view

Select scheduled_workshops_view

Select select

</

28:1 NAAMA@XE 18 rows selected in 0.094 seconds (more...)

שאלתא 1 על מבט 2

תיאור: הצגת המורים העסוקים ביותר בימים ראשון עד שלישי (כולל) וסך השיעורים שכל מורה מלמד באותם ימים.

הקוד:

```
-- all teachers that teach from sunday to tuesday- and counts the numbers of lessons in those days
SELECT TEACHER_FNAME, COUNT(*) AS LESSON_COUNT
FROM LESSON_DETAILS_VIEW
WHERE LESSON_DAY BETWEEN 1 AND 3
GROUP BY TEACHER_FNAME
ORDER BY LESSON_COUNT DESC;
```

תוצאת הרצה:

Create lesson_details_view		Select lesson_details_view	Select lesson_details_view	Create scheduled_workshops_v
TEACHER_FNAME	LESSON_COUNT			
1 David	4			
2 Philip	4			
3 Oro	3			
4 Boyd	3			
5 Eliza	3			
6 Geggy	3			
7 Tamala	3			
8 Marlon	3			
9 Adina	3			
10 Hal	3			
11 Beverlev	3			

15:13 NAAMA@XE 18 rows selected in 0.047 seconds (more...)

שאלתא 2 על מבט 2

תיאור: הצגת הכיתות עם הרבה הרבה שיעור בשבוע.

הקוד:

```
-- This query returns the 20 classes with the highest number of lessons per week
SELECT CLASS_ID, GRADE, COUNT(*) AS LESSONS_PER_WEEK
FROM LESSON_DETAILS_VIEW
WHERE ROWNUM <= 20
GROUP BY CLASS_ID, GRADE
ORDER BY LESSONS_PER_WEEK DESC;
```

תוצאת הרצה:

CLASS_ID	GRADE	LESSONS_PER_WEEK
1	2	2
2	1	2
3	2	2
4	2	2
5	11	2
6	10	2
7	7	1
8	12	1
9	11	1
10	11	1
11	2	1
12	5	1
13	11	1
14	2	1

ayala@xe 14 rows selected in 0.032 seconds