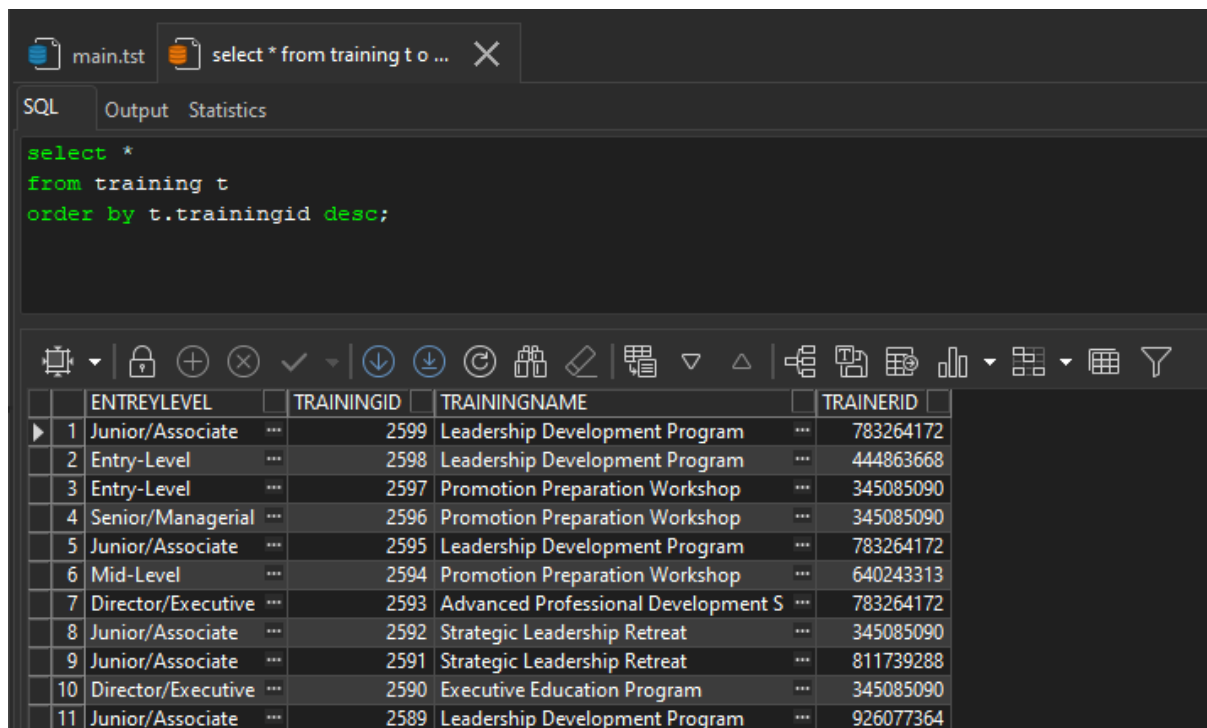


שלום שוב כעת אנחנו בעיצומו של השלב השלישי בפרוייקט שלנו וכעת נתעסק בפונקציות ופרוצדורות

### תוכנית להכשרת עובדים למשרות רצויות

כידוע שבחברה שלנו יש אפשרות לכל עובד לצאת להשתלמות ולהתקדם בחברה לכן הגדרנו תוכנית ראשית שתפקידה הוא לפתוח קבוצת התמחות חדשה שמעבירה אותה מנחה מסויים ואליו נצרך את האנשים הזמינים באותה תקופת זמן על מנת שזה לא יתנגש להם עם הבקשות חופשה שהם הגדירו

הנה מצב הפרוייקט לפני הרצת התוכנית –

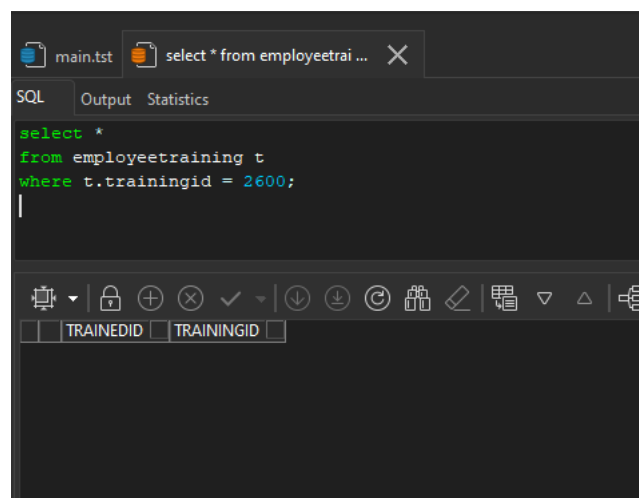


The screenshot shows a SQL IDE window with a file named 'main.tst' and a query editor containing the following SQL code:

```
select *
from training t
order by t.trainingid desc;
```

Below the query editor is a toolbar with various icons for database operations. The results pane displays a table with the following data:

	ENTREYLEVEL	TRAININGID	TRAININGNAME	TRAINERID
1	Junior/Associate	2599	Leadership Development Program	783264172
2	Entry-Level	2598	Leadership Development Program	444863668
3	Entry-Level	2597	Promotion Preparation Workshop	345085090
4	Senior/Managerial	2596	Promotion Preparation Workshop	345085090
5	Junior/Associate	2595	Leadership Development Program	783264172
6	Mid-Level	2594	Promotion Preparation Workshop	640243313
7	Director/Executive	2593	Advanced Professional Development S	783264172
8	Junior/Associate	2592	Strategic Leadership Retreat	345085090
9	Junior/Associate	2591	Strategic Leadership Retreat	811739288
10	Director/Executive	2590	Executive Education Program	345085090
11	Junior/Associate	2589	Leadership Development Program	926077364



The screenshot shows a SQL IDE window with a file named 'main.tst' and a query editor containing the following SQL code:

```
select *
from employeetraining t
where t.trainingid = 2600;
```

Below the query editor is a toolbar with various icons for database operations. The results pane displays a table with the following data:

TRAINEDID	TRAININGID
-----------	------------

הבא נתבונן בחלקים –

החלק הראשי, שבוחרת 50 אנשים מקבוצת האנשים ולוקחת את האנשים הזמינים עבור הקורס הנ"ל –

```
Declare
    Cursor EmpList is -- A cursor that holds the employees that we want to add to the new training
        Select Id
          From Employee
         Where ROWNUM <= 50;
    SingleEmp Employee%ROWTYPE;
    ProgramId NUMBER;
    ProgramSDate DATE := TO_DATE('2025-06-01', 'YYYY-MM-DD');
    ProgramEDate DATE := TO_DATE('2025-07-01', 'YYYY-MM-DD');
    Trainer Training.Trainerid%Type;
Begin

    Select max(TrainingId) + 1 into ProgramId from Training; -- Get the available id for the training
    Trainer := 783264172
    -- Craete new training
    Insert into Training (TrainingId, TrainingName, EntreyLevel,Trainerid)
    Values (ProgramId, 'New Training', 'Intermediate',Trainer);
    Commit;

    -- Add employees
    For SingleEmp in EmpList loop
        RegisterEmployeeForTraining(SingleEmp.id, ProgramId, ProgramSDate, ProgramEDate); -- Try to add the user to the training
    End loop;
End;
```

עבור כל אחד נקרא לפרוצדורה RegisterEmployeeForTraining שהיא אחראית להכניס את העובד לרשימת המשתתפים בקורס אם הוא זמין –

```
CREATE OR REPLACE PROCEDURE RegisterEmployeeForTraining(Emp_Id in Number, training_id in Number, StartProgram in Date, EndProgram in Date) is
    Eligible BOOLEAN;

    Cursor EmpList is
        Select e.Id, e.FirstName, e.LastName, l.Status, l.startDate, l.endDate
          From Employee e
         Left join LeaveRequest l on e.Id = l.EmpId
        Where e.Id = Emp_Id;

    SingleEmp EmpList % ROWTYPE;
Begin
    Open EmpList;
    Fetch EmpList into SingleEmp;

    If EmpList % Found then
        Eligible := CheckEligibility(SingleEmp.id, StartProgram, EndProgram);

        If Eligible then
            Insert into EmployeeTraining (TrainedId, TrainingId) values (SingleEmp.id, training_id);
            COMMIT;
            DBMS_OUTPUT.PUT_LINE('Employee ' || SingleEmp.id || ' has been registered for training ' || training_id || ' successfully.');
        Else
            DBMS_OUTPUT.PUT_LINE('Employee ' || SingleEmp.id || ' is not eligible for training.');
        End if;

    Else
        DBMS_OUTPUT.PUT_LINE('Employee not found.');
    End if;

    Close EmpList;
Exception
    -- If tries to add a user that already exists
    When DUP_VAL_ON_INDEX then
        DBMS_OUTPUT.PUT_LINE('User already Registered ' || SQLERRM);
        Rollback;
```

נגלה האם הוא זמין כאשר נקרא לפונקציה שמחזירה תשובה בוליאנית האם יש לו חופש באותו הזמן אם לא אזי ניקח אותו לתוכנית –

```

CREATE OR REPLACE FUNCTION CheckEligibility(emp_id IN NUMBER, T_SDate IN DATE, T_EDate IN DATE) RETURN BOOLEAN IS
    LR_List SYS_REFCURSOR;
    LR_Status VARCHAR2(35);
    LR_SDate DATE;
    LR_EDate DATE;

Begin
    Open LR_List For
        Select Status, StartDate, EndDate
        From LeaveRequest
        Where EmpId = emp_id;

    Loop
        Fetch LR_List into LR_Status, LR_SDate, LR_EDate;
        Exit when LR_List % NotFound;

        If LR_Status = 'Approved' then
            If LR_SDate <= T_EDate And LR_EDate >= T_SDate then
                Close LR_List;
                return true;
            End If;
        End If;
    End Loop;

    Close LR_List;
    return true;

Exception
    When Others then
        IF LR_List % IsOpen then
            DBMS_OUTPUT.PUT_LINE('In Check Status ERROR: ' || SQLERRM);
            Close LR_List;
        End If;

End;

```

בנוסף קיימים גם חריגות שנועדו לתת אינדיקציה של מה הבעיה בהרצה ובנוסף עוד חריגה בפרוצדורה שנועדה לתפוס שגיאה של כאשר אנחנו נכניס את אותה השורה שוב לטבלה ובכך נגרום לתקלה שאין יחודיות של המפתח בטבלה

נריץ את התוכנית

```

Employee 9202644 has been registered for training 2400 successfully.
Employee 9441503 has been registered for training 2400 successfully.
Employee 10929018 has been registered for training 2400 successfully.
Employee 11314953 has been registered for training 2400 successfully.
Employee 11571926 has been registered for training 2400 successfully.
Employee 12847336 has been registered for training 2400 successfully.
Employee 13399728 has been registered for training 2400 successfully.
Employee 13703248 has been registered for training 2400 successfully.
Employee 14236924 has been registered for training 2400 successfully.
Employee 14023397 has been registered for training 2400 successfully.
Employee 14033390 has been registered for training 2400 successfully.
Employee 19323890 has been registered for training 2400 successfully.
Employee 19855853 has been registered for training 2400 successfully.
Employee 21248971 has been registered for training 2400 successfully.
Employee 22455608 has been registered for training 2400 successfully.
Employee 23290305 has been registered for training 2400 successfully.
Employee 23340299 has been registered for training 2400 successfully.
Employee 23390250 has been registered for training 2400 successfully.
Employee 24072465 has been registered for training 2400 successfully.
Employee 25937474 has been registered for training 2400 successfully.
Employee 26010639 has been registered for training 2400 successfully.
Employee 30744889 has been registered for training 2400 successfully.
Employee 31482327 has been registered for training 2400 successfully.
Employee 31729881 has been registered for training 2400 successfully.
Employee 32829927 has been registered for training 2400 successfully.
Employee 33504846 has been registered for training 2400 successfully.
Employee 34444148 has been registered for training 2400 successfully.
Employee 36022678 has been registered for training 2400 successfully.
Employee 36129793 has been registered for training 2400 successfully.
Employee 37010884 has been registered for training 2400 successfully.
Employee 37024490 has been registered for training 2400 successfully.
Employee 38214674 has been registered for training 2400 successfully.
Employee 40579104 has been registered for training 2400 successfully.
Employee 40585737 has been registered for training 2400 successfully.
Employee 42132261 has been registered for training 2400 successfully.
Employee 42617884 has been registered for training 2400 successfully.

```

ניתן לראות שהעובדי הוכנסו בהצלחה

וכעת נריץ שאילתות כדי לראות שההתמחות נוספה וכן"ל 50 עובדים –

main.tst RegisterEmployeeForTraining.prc CheckEligibility.fnc select \* from training t

SQL Output Statistics

```
select *
from training t
order by t.trainingid desc;

select *
from EmployeeTraining et
where et.trainingid = 2600;
```

Select training Select employeeetraining

	ENTREYLEVEL	TRAININGID	TRAININGNAME	TRAINERID
1	Intermediate	2600	New Training	783264172
2	Junior/Associate	2599	Leadership Development Program	783264172
3	Entry-Level	2598	Leadership Development Program	444863668
4	Entry-Level	2597	Promotion Preparation Workshop	345085090
5	Senior/Managerial	2596	Promotion Preparation Workshop	345085090
6	Junior/Associate	2595	Leadership Development Program	783264172
7	Mid-Level	2594	Promotion Preparation Workshop	640243313
8	Director/Executive	2593	Advanced Professional Development S	783264172

```
select *
from training t
order by t.trainingid desc;

select *
from EmployeeTraining et
where et.trainingid = 2600;
```

Select training Select employeeetraining

	TRAINEDID	TRAININGID
1	822356	2600
2	1051795	2600
3	3123355	2600
4	3230649	2600
5	3322195	2600
6	5037019	2600
7	5230132	2600
8	5242121	2600
9	5347782	2600
10	7235642	2600
11	8660736	2600
12	8758852	2600
13	9202644	2600
14	9841903	2600
15	10929018	2600
16	11314953	2600
17	11571926	2600
18	12867536	2600
19	13399728	2600
20	13703248	2600
21	14236824	2600
22	16025397	2600
23	16033390	2600

## תכנית מס' 2 –

### הפונקציה - get\_employee\_for\_shift

הפונקציה מקבלת כפרמטר זמן התחלה של משמרת, ומחזירה את מספר העובד שמתאים לביצוע משמרת הזו.

אם אין עובד מתאים הפונקציה מחזירה שגיאה עם הודעה מתאימה.

### עובד שעבר הכשרה ספציפית:

הכשרה בשם "Succession Planning Seminar"

עובד שלא עובד במשמרת באותו הזמן.

התנאי שהעובד לא עובד במשמרת באותו זמן המשמרת המבוקשת - (shift\_start\_time)

עובד עם פחות משמרות -

מבין העובדים המתאימים, העובד עם מספר המשמרות הנמוך ביותר ייבחר, בתנאי שמספר

המשמרות שלו נמוך מ-3

התועלת שזה מביא לארגון -

הפונקציה מבטיחה שהמשמרות יאוישו רק על ידי עובדים שמתאימים וזמינים למשמרת מסוימת. זה תורם לחלוקה יעילה של כוח האדם ולשמירה על רמת שירות גבוהה

צילום מסך של הפונקציה -

```

1 CREATE OR REPLACE FUNCTION get_employee_for_shift(shift_start_time TIMESTAMP)
2 RETURN NUMBER IS
3     employee_id NUMBER;
4     least_shifts NUMBER := 9999;
5     CURSOR c_employee IS
6         SELECT e.Id
7         FROM Employee e
8         JOIN employeetraining et ON e.Id = et.trainedid
9         JOIN Training t ON et.TrainingId = t.TrainingId
10        WHERE t.TrainingName = 'Succession Planning Seminar' -- Example training name
11        AND NOT EXISTS (
12            SELECT 1
13            FROM employeeshift ws
14            WHERE ws.empId = e.Id
15            AND ws.Starttime = shift_start_time
16        );
17    -- ref_cur SYSREFCURSOR;
18    BEGIN
19        OPEN c_employee;
20        LOOP
21            FETCH c_employee INTO employee_id;
22            EXIT WHEN c_employee%NOTFOUND;
23
24            -- Count shifts assigned to the employee
25            OPEN ref_cur FOR
26                SELECT COUNT(*)
27                FROM employeeshift |
28                WHERE empId = employee_id;
29            FETCH ref_cur INTO least_shifts;
30            CLOSE ref_cur;
31
32            IF least_shifts < 5 THEN -- Threshold for least number of shifts
33                RETURN employee_id;
34            END IF;
35        END LOOP;
36        CLOSE c_employee;
37
38        RAISE_APPLICATION_ERROR(-20001, 'No suitable employee found for the shift.');
```

בס"ד

פרוצדורה - create\_and\_assign\_shifts

מה היא עושה?

הפרוצדורה מאפשרת לבצע שיבוץ אוטומטי של עובדים למשמרות על בסיס תאריכים נתונים. היא מפשטת את תהליך השיבוץ ומבטיחה שכל המשמרות יאושרו בצורה אופטימלית. בנוסף, טיפול בשגיאות מאפשר לזהות בעיות בשיבוץ ולפתור אותן במהירות.

התועלת שזה מביא לארגון -

הפרוצדורה - create\_and\_assign\_shifts

מהווה כלי יעיל לארגון לניהול המשמרות והקצאת העובדים בצורה אוטומטית ומבוססת על זמינות וכשירות, ובכך תורמת לשיפור התפעול היום-יומי ולניהול יעיל של המשאבים האנושיים בארגון.

צילום מסך של הפרוצדורה -

```
Code section: While loop Code section: If Code section: Block
1 CREATE OR REPLACE NONEDITIONABLE PROCEDURE create_and_assign_shifts(p_num_shifts NUMBER, p_start_date DATE) IS
2   v_employee_id NUMBER;
3   v_shift_date DATE := p_start_date;
4   v_shift_count NUMBER := 0;
5 BEGIN
6   WHILE v_shift_count < p_num_shifts LOOP
7     -- Morning shift: 09:00 - 17:00
8     IF v_shift_count < p_num_shifts THEN
9       BEGIN
10        v_employee_id := get_employee_for_shift(v_shift_date + INTERVAL '9' HOUR);
11        INSERT INTO Shift (Starttime, Endtime)
12        VALUES (v_shift_date + INTERVAL '9' HOUR, v_shift_date + INTERVAL '17' HOUR);
13        INSERT INTO employeeshift (empId, Starttime)
14        VALUES (v_employee_id, v_shift_date + INTERVAL '9' HOUR);
15
16        v_shift_count := v_shift_count + 1;
17      EXCEPTION
18        WHEN OTHERS THEN
19          DBMS_OUTPUT.PUT_LINE('Error creating/assigning morning shift on ' || v_shift_date || ': ' || SQLERRM);
20      END;
21    END IF;
22
23    -- Evening shift: 17:00 - 01:00
24    IF v_shift_count < p_num_shifts THEN
25      BEGIN
26        v_employee_id := get_employee_for_shift(v_shift_date + INTERVAL '17' HOUR);
27        INSERT INTO Shift (Starttime, Endtime)
28        VALUES (v_shift_date + INTERVAL '17' HOUR, v_shift_date + INTERVAL '25' HOUR);
29        INSERT INTO employeeshift (empId, Starttime)
30        VALUES (v_employee_id, v_shift_date + INTERVAL '17' HOUR);
31
32        v_shift_count := v_shift_count + 1;
33      EXCEPTION
34        WHEN OTHERS THEN
35          DBMS_OUTPUT.PUT_LINE('Error creating/assigning evening shift on ' || v_shift_date || ': ' || SQLERRM);
36      END;
37    END IF;
38
39    -- Move to the next day
40    v_shift_date := v_shift_date + 1;
41  END LOOP;
42 END;
```

## תכנית ראשית -

### מה התכנית עושה –

בתוכנית הראשית אנו מבצעים קריאה לפרוצדורה - create\_and\_assign\_shifts שמטרתה ליצור ולהקצות משמרות חדשות בארגון. לאחר מכן, אנו מציגים את המשמרות שנוצרו ואת העובדים שהוקצו להם. כך ניתן לוודא שהמשמרות נוצרו בצורה תקינה והעובדים שויכו להם בהתאם לזמינותם וכשירותם.

התוכנית הראשית פועלת כמעין כלי בקרה ובדיקה לוודא שפעולת הפרוצדורה התבצעה כראוי. היא מדפיסה את תוצאות השיבוץ של העובדים למשמרות, מה שתורם לשקיפות, ניהול תקין ושיפור תהליכי ניהול המשמרות בארגון.

### צילום מסך של התכנית הראשית –

```

1 DECLARE
2   v_start_date DATE := TO DATE('20-JUL-2024', 'DD-MON-YYYY');
3   v_num_shifts NUMBER := 10;
4 BEGIN
5   -- Call the procedure to create and assign shifts
6   create_and_assign_shifts(v_num_shifts, v_start_date);
7
8   -- Display the created shifts and their assigned employees
9   FOR rec IN (SELECT s.Starttime, s.Endtime, e.lastName
10                FROM Shift s
11                JOIN employeeshift es ON s.Starttime = es.Starttime
12                JOIN Employee e ON es.empId = e.Id
13                WHERE s.Starttime >= v_start_date) LOOP
14     DBMS_OUTPUT.PUT_LINE('Shift from ' || rec.Starttime || ' to ' || rec.Endtime || ' assigned to ' || rec.lastName);
15   END LOOP;
16
17 EXCEPTION
18   WHEN OTHERS THEN
19     DBMS_OUTPUT.PUT_LINE('Error: ' || SQLERRM);
20 END;
21 /

```

### צילום מסך של הקלט של התכנית הראשית –

```

Edit SYS.GET_EMPLOYEE_FOR_SHIFT@XE create_and_assign_shifts.prc mainProgram.tst select * from shift
Test script DBMS Output Statistics Profiler Trace
Clear Buffer size 10 Enabled
Shift from 09-JAN-26 03.30.00.000000 PM to 09-JAN-26 11.30.00.000000 PM assigned to Trevor
Shift from 15-SEP-25 03.30.00.000000 PM to 15-SEP-25 11.30.00.000000 PM assigned to Ladel
Shift from 19-SEP-25 03.30.00.000000 PM to 19-SEP-25 11.30.00.000000 PM assigned to O'Dowling
Shift from 08-NOV-25 07.30.00.000000 AM to 08-NOV-25 03.30.00.000000 PM assigned to Ogilvy
Shift from 25-DEC-25 07.30.00.000000 AM to 25-DEC-25 03.30.00.000000 PM assigned to Arington
Shift from 21-MAR-26 03.30.00.000000 PM to 21-MAR-26 11.30.00.000000 PM assigned to Lorentz
Shift from 23-JAN-26 11.30.00.000000 PM to 24-JAN-26 07.30.00.000000 AM assigned to Rooney
Shift from 17-MAR-26 07.30.00.000000 AM to 17-MAR-26 03.30.00.000000 PM assigned to Teaser
Shift from 10-SEP-25 07.30.00.000000 PM to 10-SEP-25 03.30.00.000000 PM assigned to Gillson
Shift from 28-JAN-26 11.30.00.000000 PM to 29-JAN-26 07.30.00.000000 AM assigned to Bang
Shift from 07-AUG-25 11.30.00.000000 PM to 08-AUG-25 07.30.00.000000 AM assigned to Fawthrop
Shift from 24-SEP-25 03.30.00.000000 PM to 24-SEP-25 11.30.00.000000 PM assigned to Mendus
Shift from 01-APR-26 07.30.00.000000 AM to 01-APR-26 03.30.00.000000 PM assigned to Dales
Shift from 02-DEC-25 03.30.00.000000 PM to 02-DEC-25 11.30.00.000000 PM assigned to Stammirs
Shift from 17-FEB-26 03.30.00.000000 PM to 17-FEB-26 11.30.00.000000 PM assigned to Lyffe
Shift from 04-MAR-26 07.30.00.000000 AM to 04-MAR-26 03.30.00.000000 PM assigned to Myrick
Shift from 08-NOV-25 07.30.00.000000 AM to 08-NOV-25 03.30.00.000000 PM assigned to Ledford
Shift from 19-SEP-25 11.30.00.000000 PM to 20-SEP-25 07.30.00.000000 AM assigned to Syllett
Shift from 22-OCT-25 11.30.00.000000 PM to 23-OCT-25 07.30.00.000000 AM assigned to Willingale
Shift from 04-NOV-25 03.30.00.000000 PM to 04-NOV-25 11.30.00.000000 PM assigned to Doram
Shift from 01-FEB-26 03.30.00.000000 PM to 01-FEB-26 11.30.00.000000 PM assigned to Blewett
Shift from 16-DEC-25 03.30.00.000000 PM to 16-DEC-25 11.30.00.000000 PM assigned to Blackler

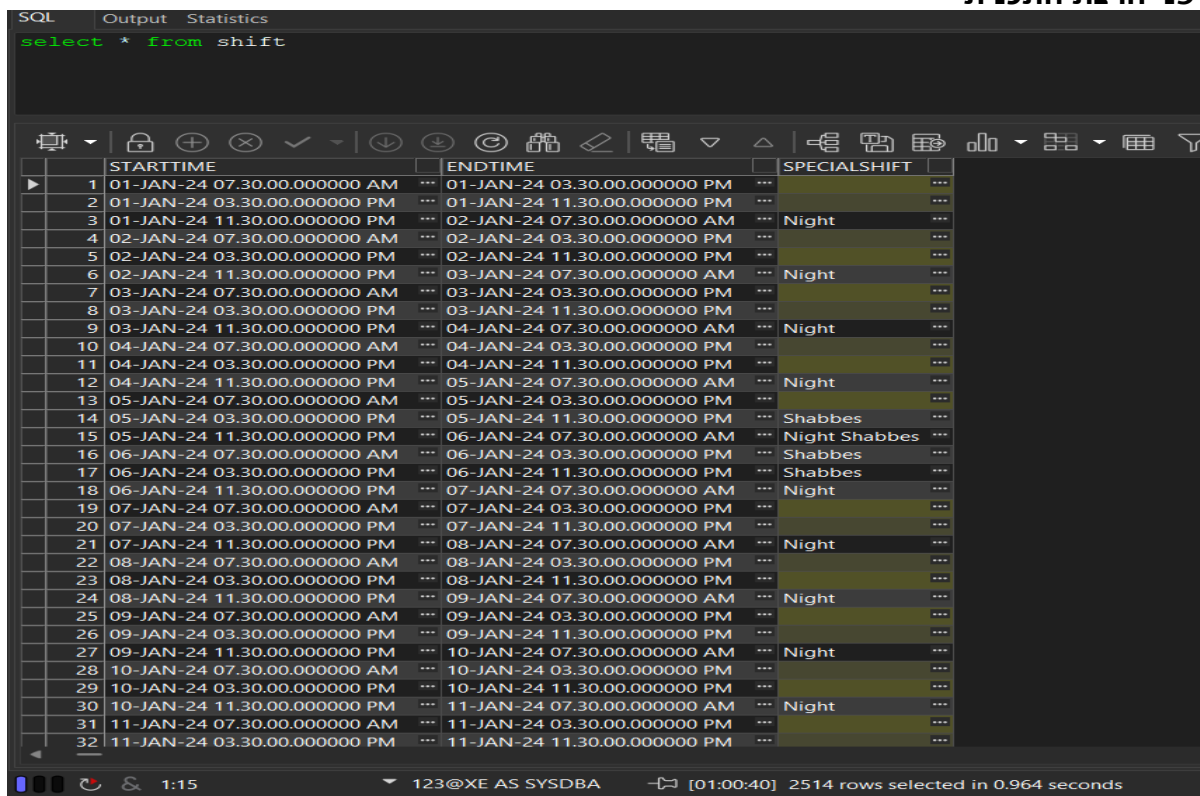
```

בס"ד

מאחר ובמהלך התכנית הראשית אנחנו קוראים לפרוצדורה שיוצרת משמרות חדשות ומכניסה אליהם אנשים שפנויים באותו הזמן.

אז צריך להראות אינדיקציה שהפרוצדורה באמת מוסיפה משמרות.

**לפני הרצת התכנית –**



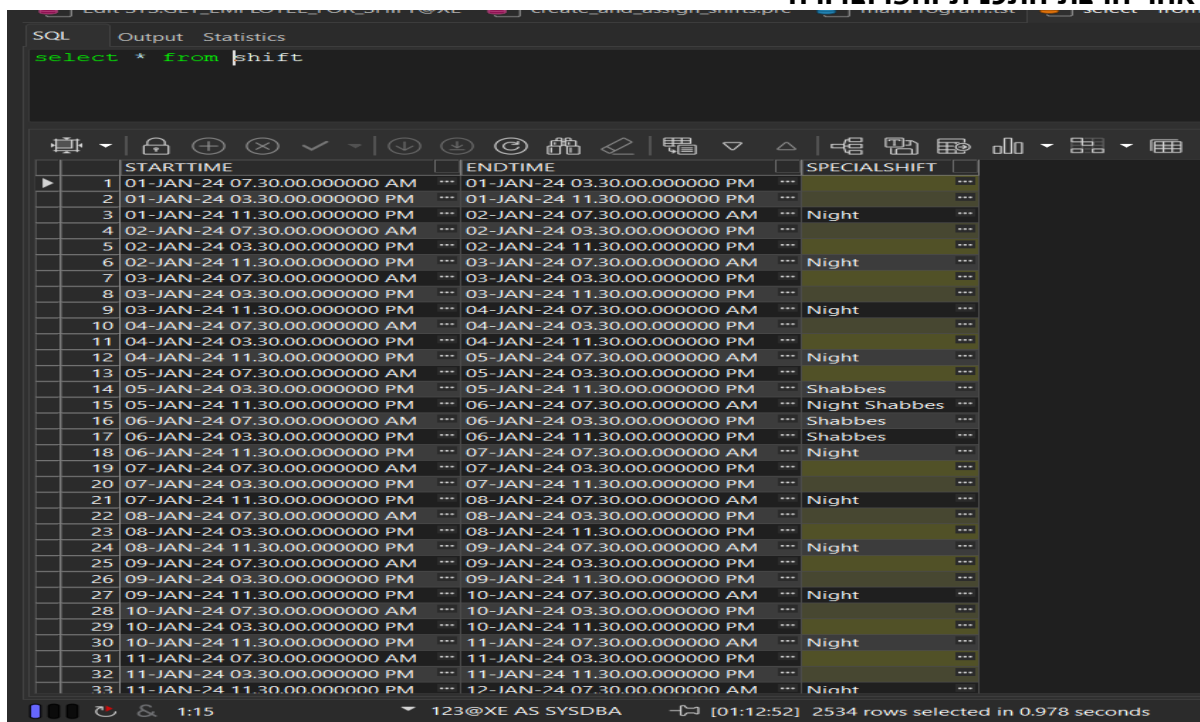
SQL Output Statistics

```
select * from shift
```

	STARTTIME	ENDTIME	SPECIALSHIFT
1	01-JAN-24 07.30.00.000000 AM	01-JAN-24 03.30.00.000000 PM	
2	01-JAN-24 03.30.00.000000 PM	01-JAN-24 11.30.00.000000 PM	
3	01-JAN-24 11.30.00.000000 PM	02-JAN-24 07.30.00.000000 AM	Night
4	02-JAN-24 07.30.00.000000 AM	02-JAN-24 03.30.00.000000 PM	
5	02-JAN-24 03.30.00.000000 PM	02-JAN-24 11.30.00.000000 PM	
6	02-JAN-24 11.30.00.000000 PM	03-JAN-24 07.30.00.000000 AM	Night
7	03-JAN-24 07.30.00.000000 AM	03-JAN-24 03.30.00.000000 PM	
8	03-JAN-24 03.30.00.000000 PM	03-JAN-24 11.30.00.000000 PM	
9	03-JAN-24 11.30.00.000000 PM	04-JAN-24 07.30.00.000000 AM	Night
10	04-JAN-24 07.30.00.000000 AM	04-JAN-24 03.30.00.000000 PM	
11	04-JAN-24 03.30.00.000000 PM	04-JAN-24 11.30.00.000000 PM	
12	04-JAN-24 11.30.00.000000 PM	05-JAN-24 07.30.00.000000 AM	Night
13	05-JAN-24 07.30.00.000000 AM	05-JAN-24 03.30.00.000000 PM	
14	05-JAN-24 03.30.00.000000 PM	05-JAN-24 11.30.00.000000 PM	Shabbes
15	05-JAN-24 11.30.00.000000 PM	06-JAN-24 07.30.00.000000 AM	Night Shabbes
16	06-JAN-24 07.30.00.000000 AM	06-JAN-24 03.30.00.000000 PM	Shabbes
17	06-JAN-24 03.30.00.000000 PM	06-JAN-24 11.30.00.000000 PM	Shabbes
18	06-JAN-24 11.30.00.000000 PM	07-JAN-24 07.30.00.000000 AM	Night
19	07-JAN-24 07.30.00.000000 AM	07-JAN-24 03.30.00.000000 PM	
20	07-JAN-24 03.30.00.000000 PM	07-JAN-24 11.30.00.000000 PM	
21	07-JAN-24 11.30.00.000000 PM	08-JAN-24 07.30.00.000000 AM	Night
22	08-JAN-24 07.30.00.000000 AM	08-JAN-24 03.30.00.000000 PM	
23	08-JAN-24 03.30.00.000000 PM	08-JAN-24 11.30.00.000000 PM	
24	08-JAN-24 11.30.00.000000 PM	09-JAN-24 07.30.00.000000 AM	Night
25	09-JAN-24 07.30.00.000000 AM	09-JAN-24 03.30.00.000000 PM	
26	09-JAN-24 03.30.00.000000 PM	09-JAN-24 11.30.00.000000 PM	
27	09-JAN-24 11.30.00.000000 PM	10-JAN-24 07.30.00.000000 AM	Night
28	10-JAN-24 07.30.00.000000 AM	10-JAN-24 03.30.00.000000 PM	
29	10-JAN-24 03.30.00.000000 PM	10-JAN-24 11.30.00.000000 PM	
30	10-JAN-24 11.30.00.000000 PM	11-JAN-24 07.30.00.000000 AM	Night
31	11-JAN-24 07.30.00.000000 AM	11-JAN-24 03.30.00.000000 PM	
32	11-JAN-24 03.30.00.000000 PM	11-JAN-24 11.30.00.000000 PM	

1:15 123@XE AS SYSDBA [01:00:40] 2514 rows selected in 0.964 seconds

**לאחר הרצת התכנית והפרוצדורה -**



SQL Output Statistics

```
select * from shift
```

	STARTTIME	ENDTIME	SPECIALSHIFT
1	01-JAN-24 07.30.00.000000 AM	01-JAN-24 03.30.00.000000 PM	
2	01-JAN-24 03.30.00.000000 PM	01-JAN-24 11.30.00.000000 PM	
3	01-JAN-24 11.30.00.000000 PM	02-JAN-24 07.30.00.000000 AM	Night
4	02-JAN-24 07.30.00.000000 AM	02-JAN-24 03.30.00.000000 PM	
5	02-JAN-24 03.30.00.000000 PM	02-JAN-24 11.30.00.000000 PM	
6	02-JAN-24 11.30.00.000000 PM	03-JAN-24 07.30.00.000000 AM	Night
7	03-JAN-24 07.30.00.000000 AM	03-JAN-24 03.30.00.000000 PM	
8	03-JAN-24 03.30.00.000000 PM	03-JAN-24 11.30.00.000000 PM	
9	03-JAN-24 11.30.00.000000 PM	04-JAN-24 07.30.00.000000 AM	Night
10	04-JAN-24 07.30.00.000000 AM	04-JAN-24 03.30.00.000000 PM	
11	04-JAN-24 03.30.00.000000 PM	04-JAN-24 11.30.00.000000 PM	
12	04-JAN-24 11.30.00.000000 PM	05-JAN-24 07.30.00.000000 AM	Night
13	05-JAN-24 07.30.00.000000 AM	05-JAN-24 03.30.00.000000 PM	
14	05-JAN-24 03.30.00.000000 PM	05-JAN-24 11.30.00.000000 PM	Shabbes
15	05-JAN-24 11.30.00.000000 PM	06-JAN-24 07.30.00.000000 AM	Night Shabbes
16	06-JAN-24 07.30.00.000000 AM	06-JAN-24 03.30.00.000000 PM	Shabbes
17	06-JAN-24 03.30.00.000000 PM	06-JAN-24 11.30.00.000000 PM	Shabbes
18	06-JAN-24 11.30.00.000000 PM	07-JAN-24 07.30.00.000000 AM	Night
19	07-JAN-24 07.30.00.000000 AM	07-JAN-24 03.30.00.000000 PM	
20	07-JAN-24 03.30.00.000000 PM	07-JAN-24 11.30.00.000000 PM	
21	07-JAN-24 11.30.00.000000 PM	08-JAN-24 07.30.00.000000 AM	Night
22	08-JAN-24 07.30.00.000000 AM	08-JAN-24 03.30.00.000000 PM	
23	08-JAN-24 03.30.00.000000 PM	08-JAN-24 11.30.00.000000 PM	
24	08-JAN-24 11.30.00.000000 PM	09-JAN-24 07.30.00.000000 AM	Night
25	09-JAN-24 07.30.00.000000 AM	09-JAN-24 03.30.00.000000 PM	
26	09-JAN-24 03.30.00.000000 PM	09-JAN-24 11.30.00.000000 PM	
27	09-JAN-24 11.30.00.000000 PM	10-JAN-24 07.30.00.000000 AM	Night
28	10-JAN-24 07.30.00.000000 AM	10-JAN-24 03.30.00.000000 PM	
29	10-JAN-24 03.30.00.000000 PM	10-JAN-24 11.30.00.000000 PM	
30	10-JAN-24 11.30.00.000000 PM	11-JAN-24 07.30.00.000000 AM	Night
31	11-JAN-24 07.30.00.000000 AM	11-JAN-24 03.30.00.000000 PM	
32	11-JAN-24 03.30.00.000000 PM	11-JAN-24 11.30.00.000000 PM	
33	11-JAN-24 11.30.00.000000 PM	12-JAN-24 07.30.00.000000 AM	Night

1:15 123@XE AS SYSDBA [01:12:52] 2534 rows selected in 0.978 seconds

כמו שאפשר לראות לאחר הרצת התכנית הראשית יש יותר משמרות בטבלת המשמרות.