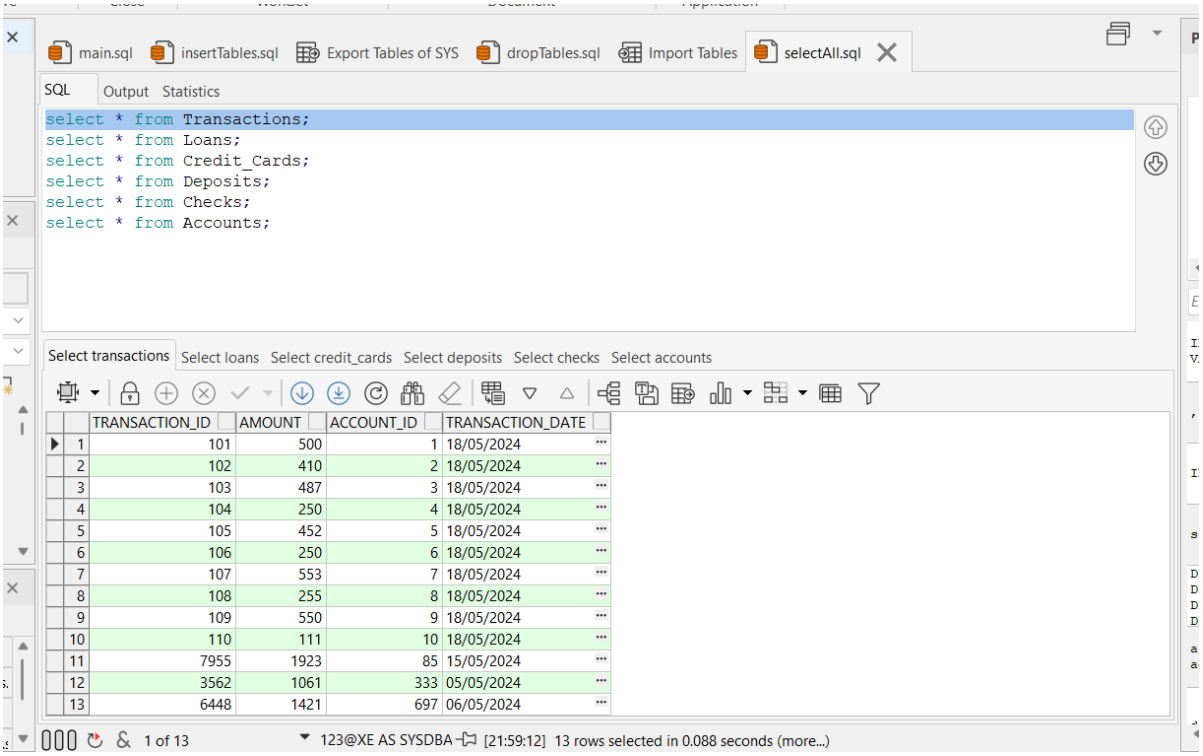
נריץ את הקובץ selectAll בכדי לראות שהנתונים חזרו:




| | TRANSACTION_ID | AMOUNT | ACCOUNT_ID | TRANSACTION_DATE |
|---|---|---|---|---|
| 1 | 101 | 500 | 1 | 18/05/2024 |
| 2 | 102 | 410 | 2 | 18/05/2024 |
| 3 | 103 | 487 | 3 | 18/05/2024 |
| 4 | 104 | 250 | 4 | 18/05/2024 |
| 5 | 105 | 452 | 5 | 18/05/2024 |
| 6 | 106 | 250 | 6 | 18/05/2024 |
| 7 | 107 | 553 | 7 | 18/05/2024 |
| 8 | 108 | 255 | 8 | 18/05/2024 |
| 9 | 109 | 550 | 9 | 18/05/2024 |
| 10 | 110 | 111 | 10 | 18/05/2024 |
| 11 | 7955 | 1923 | 85 | 15/05/2024 |
| 12 | 3562 | 1061 | 333 | 05/05/2024 |
| 13 | 6448 | 1421 | 697 | 06/05/2024 |

טא דא! הכל שב אלינו.

עד כאן שלב 1.

**שלב 2:**

ניצור שאילתות מורכבות:

SQL Output Statistics

```
SELECT Customer_ID, COUNT(Account_ID) AS Num_Accounts, AVG(Balance) AS Avg_Balance
FROM Accounts
GROUP BY Customer_ID
ORDER BY Num_Accounts DESC;
```

בשאילתה הנ"ל ניתן לראות שאנו מחפשים את כל הלקוחות שיש להם כמה חשבונות, ומחזירים לכל לקוח את כמות החשבונות שלו, ביחד עם הממוצע בכל החשבונות, וכמובן את מספר הלקוח.
הנה דוגמא לפלט:

| | ACCOUNT_ID | ACCOUNT_TYPE | BALANCE |
|---|---|---|---|
| 1 | 165 | Checking | 975649 |
| 2 | 163 | Checking | 915437 |
| 3 | 153 | Savings | 783778 |
| 4 | 149 | Checking | 892744 |
| 5 | 147 | Savings | 48356 |
| 6 | 143 | Checking | 380458 |
| 7 | 142 | Savings | 174840 |
| 8 | 135 | Checking | 696326 |
| 9 | 123 | Savings | 677440 |

הנה עוד שאילתה:

```
SELECT Account_ID, Account_Type, Balance
FROM Accounts
WHERE Account_Opening_Date < TO_DATE('2024-05-20', 'YYYY-MM-DD')
AND Account_ID NOT IN (
    SELECT DISTINCT Account_ID FROM Transactions);
```

בדוגמא הנ"ל ניתן לראות שאנו מחפשים את כל החשבונות שלא היו פעילים בשבוע האחרון, כלומר לא ביצעו של טרנזקציה בשבוע שחלף. אבל גם נוודא שהם לא חדשים בבנק ואכן החשבון שלהם נפתח לפני שבוע ויותר. נייצג את הנתונים בעזרת מספר חשבון, יתרת חשבון, וסוג חשבון.
דוגמא לפלט:

| | ACCOUNT_ID | ACCOUNT_TYPE | BALANCE |
|---|---|---|---|
| 1 | 165 | Checking | 975649 |
| 2 | 163 | Checking | 915437 |
| 3 | 153 | Savings | 783778 |
| 4 | 149 | Checking | 892744 |
| 5 | 147 | Savings | 48356 |
| 6 | 143 | Checking | 380458 |
| 7 | 142 | Savings | 174840 |
| 8 | 135 | Checking | 696326 |
| 9 | 123 | Savings | 677440 |

שאילתה 3:

```
select A.account_Id,
       max(D.Deposit_Amount) as maxDeposit,
       max(L.Loan_Amount) as maxLoan
from accounts A
Join Loans L on A.ACCOUNT_ID = L.Account_id
Join Deposits D on A.ACCOUNT_ID = D.Account_id
Group BY A.ACCOUNT_ID;
```

כאן אנו מחפשים עבור כל חשבון מה הסכום הגבוהה ביותר שהוא לקח להלוואה ומה הסכום הגבוה ביותר שהוא הפקיד לחשבון. את נתונים אלו נציג ביחד עם מספר חשבון:

| | ACCOUNT_ID | MAXDEPOSIT | MAXLOAN |
|---|---|---|---|
| 1 | 6 | 5000 | 5000 |
| 2 | 34293 | 225 | 71996 |
| 3 | 554 | 451 | 81157 |
| 4 | 124 | 172 | 9369 |
| 5 | 69493 | 85 | 47970 |
| 6 | 161 | 138 | 32841 |
| 7 | 66260 | 331 | 297 |
| 8 | 63305 | 474 | 78390 |
| 9 | 770 | 335 | 44754 |

שאילתה 4:

```
SELECT A.Account_ID,
       CC.Card_Number,
       CC.Expiration_Date AS Card_Expiration_Date,
       L.Loan_ID,
       L.Interest_Rate AS Loan_Interest_Rate
FROM Accounts A
LEFT JOIN Credit_Cards CC ON A.Account_ID = CC.Account_ID AND
      CC.Expiration_Date <= ADD_MONTHS(SYSDATE, 1)
LEFT JOIN Loans L ON A.Account_ID = L.Account_ID AND
      L.End_Date <= ADD_MONTHS(SYSDATE, 1);
```

כאן אנו בעצם מבקשים לחפש עבור כל מספר חשבון אם יש לו הלוואה שעומדת להגמר בקרוב וכן אם יש לו כרטיס אשראי שעומד לפוג תוקפו לא עלינו. לגבי ההלוואה נחזיר גם את מידת הריבית שהוא משלם על ההלוואה, בכדי שהלקוח יהיה מרוצה ויראה לפניו כמה ריבית הוא משלם ועכשיו ההלוואה עומדת להגמר ומה טוב ומה נעים שלא יצטרך להמשיך לשלם אותה ריבית.

| | ACCOUNT_ID | CARD_NUMBER | CARD_EXPIRATION_DATE | LOAN_ID | LOAN_INTEREST_RATE |
|---|---|---|---|---|---|
| 1 | 1 | | | 201 | 7 |
| 2 | 1 | | | 8893 | 1 |
| 3 | 2 | 5776057 | 01/05/2024 | 202 | 8 |
| 4 | 3 | 5029408 | 18/05/2024 | 203 | 7 |
| 5 | 3 | 5029408 | 18/05/2024 | 4176 | 3 |
| 6 | 3 | 4041855 | 15/05/2024 | 203 | 7 |
| 7 | 3 | 4041855 | 15/05/2024 | 4176 | 3 |
| 8 | 4 | | | 204 | 8 |
| 9 | 4 | | | 6916 | 4 |

ניתן לראות שבהתאם לאופי השאילתה לא כל הטבלאות מלאות, וזאת משום שלא כל חשבון יש לו גם את הנתונים האלו וגם את אלו, אל ישנם חלק שיש להם הלוואה שעומדת להגמר וחלק שכרטיס האשראי עומד לפוג.

שאילתת מחיקה:

```
DELETE FROM Credit_Cards
WHERE Expiration_Date < SYSDATE;

select * from Credit_Cards
```

נראה את התוצאה:

Select accounts | Select accounts | Select accounts | Select accounts | Delete credit_cards | Select credit_cards

| | CARD_ID | CARD_NUMBER | CARD_TYPE | EXPIRATION_DATE | CREDIT_LIMIT | ACCOUNT_ID |
|---|---|---|---|---|---|---|
| 1 | 301 | 123456789012 | Visa | 20/05/2027 | 15000 | 1 |
| 2 | 302 | 987654321012 | Mastercard | 20/05/2026 | 10000 | 2 |
| 3 | 303 | 111122223333 | Visa | 30/06/2028 | 12000 | 3 |
| 4 | 304 | 444455556666 | Mastercard | 15/04/2025 | 8000 | 4 |
| 5 | 305 | 777788889999 | Visa | 22/08/2027 | 9000 | 5 |
| 6 | 306 | 123443215678 | Mastercard | 31/12/2026 | 11000 | 6 |
| 7 | 307 | 876543219876 | Visa | 01/01/2029 | 14000 | 7 |
| 8 | 308 | 112233445566 | Mastercard | 20/07/2026 | 13000 | 8 |
| 9 | 309 | 998877665544 | Visa | 25/11/2027 | 16000 | 9 |

40:27 123@XE AS SYSDBA [22:26:52] 9 rows selected in 0.074 seconds (more...)

נריץ ROLLBACK בכדי לא לאבד את הנתונים.

הוכחה לאי איבוד מידע 👍

```
DELETE FROM Credit_Cards
WHERE Expiration_Date < SYSDATE;

ROLLBACK;

select * from Credit_Cards
where Expiration_Date < SYSDATE;
```

Select accounts | Select accounts | Select accounts | Select accounts | Delete credit_cards | Rollback | Select credit_cards

| | CARD_ID | CARD_NUMBER | CARD_TYPE | EXPIRATION_DATE | CREDIT_LIMIT | ACCOUNT_ID |
|---|---|---|---|---|---|---|
| 1 | 76249 | 9427907 | mastercard | 01/05/2024 | 6290 | 65893 |
| 2 | 19012 | 9252370 | debit | 01/05/2024 | 6262 | 14577 |
| 3 | 33795 | 1634604 | mastercard | 12/05/2024 | 6157 | 163 |
| 4 | 75548 | 1953775 | mastercard | 07/05/2024 | 5938 | 500 |
| 5 | 64505 | 7537240 | visa | 06/05/2024 | 5593 | 22252 |
| 6 | 92580 | 5919936 | mastercard | 07/05/2024 | 5157 | 421 |
| 7 | 56128 | 3997988 | debit | 15/05/2024 | 4721 | 69603 |
| 8 | 40997 | 4120698 | mastercard | 20/05/2024 | 4375 | 91608 |
| 9 | 98262 | 3428629 | debit | 07/05/2024 | 4158 | 46450 |

33:1 [54] 123@XE AS SYSDBA [22:29:09] 9 rows selected in 0.089 seconds (more...)

עוד שאילתת מחיקה

```
DELETE FROM Loans
WHERE End_Date < SYSDATE;

ROLLBACK;
```

נמחק את כל ההלוואות שזמן הסיום שלהם קטן מהיום, כלומר נגמרו.

שאילתת עידכון:

```
UPDATE Credit_Cards
SET Expiration_Date = ADD_MONTHS(Expiration_Date, 12);
```

נעדכן את תאריך התפוגה של כרטיס הארשאי בעוד שנה.

```
UPDATE Credit_Cards
SET Expiration_Date = ADD_MONTHS(Expiration_Date, 12);

select * from Credit_Cards
WHERE End_Date < SYSDATE;
```

Select accounts | Select accounts | Select accounts | Select accounts | Delete credit_cards | Rollback | Delete loans | Rollback | Update credit_cards | Select credit_car

(no result set)

כאן ניתן לראות שאחרי שהוספנו לכולם שנה, אין תוצאות, לפני זה ביקשנו רק את מי שקטן מהיום וכולם היו בתאריכים קטנים להיום אך קטנים ממנו, ברגע שהוספנו לכולם שנה, לא נותר את מי לבחור.

שאילתת עידכון 2:

```
UPDATE Loans
SET Interest_Rate = Interest_Rate + 0.5;
```

נעדכן את כל הריביות על הלוואות בעוד חצי אחוז. בכל זאת, הריבית עולה, אינפלציה בשמיים. בואו, זה לא מה שהיה פעם, 2 אחוז ריבית.

שאילתא עם פרמטרים 1:

```
DECLARE
    p_Balance NUMBER := 10000;
    CURSOR c_Accounts IS
        SELECT Account_ID, Customer_ID, Account_Type, Balance, Account_Opening_Date
        FROM Accounts
        WHERE Balance > p_Balance;
BEGIN
    FOR rec IN c_Accounts LOOP
        DBMS_OUTPUT.PUT_LINE('Account ID: ' || rec.Account_ID || ', Customer ID: ' || rec.Customer_ID ||
                             ', Account Type: ' || rec.Account_Type || ', Balance: ' || rec.Balance ||
                             ', Opening Date: ' || rec.Account_Opening_Date);
    END LOOP;
END;
/
```

נחפש את המידע על החשבון שהיתרה שלו גדולה מהפרמטר :
פלט:

```
Account ID: 27179, Customer ID: 371, Account Type: Savings, Balance: 775809, Opening Date: 20-MAY-24
Account ID: 45206, Customer ID: 368, Account Type: Savings, Balance: 184913, Opening Date: 07-MAY-24
Account ID: 31951, Customer ID: 363, Account Type: Savings, Balance: 81168, Opening Date: 15-MAY-24
Account ID: 21059, Customer ID: 357, Account Type: Checking, Balance: 387989, Opening Date: 15-MAY-24
Account ID: 46901, Customer ID: 344, Account Type: Checking, Balance: 907853, Opening Date: 20-MAY-24
Account ID: 73766, Customer ID: 337, Account Type: Savings, Balance: 579008, Opening Date: 06-MAY-24
Account ID: 12055, Customer ID: 329, Account Type: Savings, Balance: 879697, Opening Date: 18-MAY-24
Account ID: 43050, Customer ID: 318, Account Type: Savings, Balance: 75870, Opening Date: 06-MAY-24
Account ID: 94436, Customer ID: 310, Account Type: Checking, Balance: 258853, Opening Date: 12-MAY-24
Account ID: 51745, Customer ID: 300, Account Type: Savings, Balance: 397349, Opening Date: 20-MAY-24
Account ID: 57740, Customer ID: 289, Account Type: Checking, Balance: 905764, Opening Date: 15-MAY-24
Account ID: 44032, Customer ID: 279, Account Type: Savings, Balance: 818644, Opening Date: 16-MAY-24
Account ID: 24487, Customer ID: 268, Account Type: Checking, Balance: 712644, Opening Date: 18-MAY-24
Account ID: 21541, Customer ID: 257, Account Type: Savings, Balance: 696039, Opening Date: 07-MAY-24
Account ID: 80906, Customer ID: 247, Account Type: Savings, Balance: 64451, Opening Date: 05-MAY-24
Account ID: 45906, Customer ID: 237, Account Type: Checking, Balance: 879812, Opening Date: 15-MAY-24
Account ID: 13940, Customer ID: 228, Account Type: Checking, Balance: 752056, Opening Date: 18-MAY-24
Account ID: 81105, Customer ID: 218, Account Type: Checking, Balance: 573483, Opening Date: 19-MAY-24
Account ID: 43236, Customer ID: 211, Account Type: Checking, Balance: 679369, Opening Date: 18-MAY-24
Account ID: 13271, Customer ID: 202, Account Type: Checking, Balance: 918042, Opening Date: 07-MAY-24
Account ID: 48175, Customer ID: 196, Account Type: Checking, Balance: 713675, Opening Date: 15-MAY-24
Account ID: 36975, Customer ID: 190, Account Type: Savings, Balance: 692597, Opening Date: 06-MAY-24
Account ID: 6748, Customer ID: 184, Account Type: Checking, Balance: 674222, Opening Date: 19-MAY-24
Account ID: 13222, Customer ID: 179, Account Type: Savings, Balance: 633371, Opening Date: 12-MAY-24
Account ID: 17512, Customer ID: 176, Account Type: Savings, Balance: 919536, Opening Date: 20-MAY-24
Account ID: 66260, Customer ID: 171, Account Type: Savings, Balance: 663978, Opening Date: 06-MAY-24
```

638:1 123@XE AS SYSDBA [23:04:29] Done in 0.011 seconds

שאילתא עם פרמטרים 2:

```oracle_plsql
DECLARE
    p_Start_Date DATE := TO_DATE('2022-01-01', 'YYYY-MM-DD');
    p_End_Date DATE := TO_DATE('2026-12-31', 'YYYY-MM-DD');
    CURSOR c_Loans IS
        SELECT Loan_ID, Loan_Amount, Interest_Rate, Start_Date, End_Date, Account_ID
        FROM Loans
        WHERE Start_Date BETWEEN p_Start_Date AND p_End_Date;
BEGIN
    FOR rec IN c_Loans LOOP
        DBMS_OUTPUT.PUT_LINE('Loan ID: ' || rec.Loan_ID ||', Loan Amount: ' ||
         rec.Loan_Amount || ', Interest Rate: ' || rec.Interest_Rate ||
                             ', Start Date: ' || rec.Start_Date || ', End Date: ' || rec.End_Date ||
                             ', Account ID: ' || rec.Account_ID);
    END LOOP;
END;
/
```

בודקת איזה הלוואות נלקחו בין תאריכים מסויימים:

פלט:

```
Clear   Buffer size 3200000   Enabled
Loan ID: 9002, Loan Amount: 86087, Interest Rate: 1, Start Date: 20-MAY-24, End Date: 06-MAY-24, Account ID: 265
Loan ID: 3903, Loan Amount: 63723, Interest Rate: 1, Start Date: 20-MAY-24, End Date: 01-MAY-24, Account ID: 47
Loan ID: 5357, Loan Amount: 19896, Interest Rate: 1, Start Date: 07-MAY-24, End Date: 06-MAY-24, Account ID: 86
Loan ID: 9705, Loan Amount: 42304, Interest Rate: 1, Start Date: 12-MAY-24, End Date: 19-MAY-24, Account ID: 585
Loan ID: 5020, Loan Amount: 42617, Interest Rate: 1, Start Date: 07-MAY-24, End Date: 19-MAY-24, Account ID: 331
Loan ID: 1902, Loan Amount: 73929, Interest Rate: 1, Start Date: 15-MAY-24, End Date: 20-MAY-24, Account ID: 190
Loan ID: 7081, Loan Amount: 5156, Interest Rate: 1, Start Date: 15-MAY-24, End Date: 07-MAY-24, Account ID: 67
Loan ID: 8405, Loan Amount: 1194, Interest Rate: 1, Start Date: 18-MAY-24, End Date: 15-MAY-24, Account ID: 1904
Loan ID: 867, Loan Amount: 46430, Interest Rate: 1, Start Date: 18-MAY-24, End Date: 01-MAY-24, Account ID: 7617
Loan ID: 9824, Loan Amount: 72275, Interest Rate: 1, Start Date: 01-MAY-24, End Date: 01-MAY-24, Account ID: 969
Loan ID: 5056, Loan Amount: 47268, Interest Rate: 1, Start Date: 20-MAY-24, End Date: 07-MAY-24, Account ID: 945
Loan ID: 8899, Loan Amount: 61030, Interest Rate: 1, Start Date: 07-MAY-24, End Date: 18-MAY-24, Account ID: 973
Loan ID: 9586, Loan Amount: 73215, Interest Rate: 1, Start Date: 18-MAY-24, End Date: 01-MAY-24, Account ID: 218
Loan ID: 2300, Loan Amount: 85881, Interest Rate: 1, Start Date: 20-MAY-24, End Date: 20-MAY-24, Account ID: 409
Loan ID: 5013, Loan Amount: 58982, Interest Rate: 1, Start Date: 01-MAY-24, End Date: 07-MAY-24, Account ID: 964
Loan ID: 9894, Loan Amount: 95572, Interest Rate: 1, Start Date: 06-MAY-24, End Date: 18-MAY-24, Account ID: 994
Loan ID: 9101, Loan Amount: 78390, Interest Rate: 1, Start Date: 06-MAY-24, End Date: 20-MAY-24, Account ID: 633
Loan ID: 2937, Loan Amount: 34829, Interest Rate: 1, Start Date: 19-MAY-24, End Date: 07-MAY-24, Account ID: 633
Loan ID: 7070, Loan Amount: 53907, Interest Rate: 1, Start Date: 20-MAY-24, End Date: 19-MAY-24, Account ID: 9
Loan ID: 6391, Loan Amount: 75089, Interest Rate: 1, Start Date: 16-MAY-24, End Date: 01-MAY-24, Account ID: 144
Loan ID: 7550, Loan Amount: 47933, Interest Rate: 1, Start Date: 15-MAY-24, End Date: 01-MAY-24, Account ID: 812
Loan ID: 2707, Loan Amount: 43971, Interest Rate: 1, Start Date: 07-MAY-24, End Date: 15-MAY-24, Account ID: 846
Loan ID: 3498, Loan Amount: 40427, Interest Rate: 1, Start Date: 19-MAY-24, End Date: 19-MAY-24, Account ID: 487
Loan ID: 7877, Loan Amount: 64869, Interest Rate: 1, Start Date: 18-MAY-24, End Date: 15-MAY-24, Account ID: 284
```

שאילתת טרנזקציה לפי סוג חשבון:

SQL  Output  Statistics

```oracle
DECLARE
    p_Account_Type VARCHAR2(10) := 'Savings'; -- ערך לדוגמה לפרמטר
    CURSOR c_Transactions IS
        SELECT t.Transaction_ID, t.Amount, t.Account_ID, t.Transaction_Date
        FROM Transactions t
        JOIN Accounts a ON t.Account_ID = a.Account_ID
        WHERE a.Account_Type = p_Account_Type;
BEGIN
    FOR rec IN c_Transactions LOOP
        DBMS_OUTPUT.PUT_LINE('Transaction ID: ' || rec.Transaction_ID);
        DBMS_OUTPUT.PUT_LINE('Amount: ' || rec.Amount);
        DBMS_OUTPUT.PUT_LINE('Account ID: ' || rec.Account_ID);
        DBMS_OUTPUT.PUT_LINE('Transaction Date: ' || rec.Transaction_Date);
    END LOOP;
END;
/
```

פלט:

```
Transaction ID: 235
Amount: 2519
Account ID: 31429
Transaction Date: 01-MAY-24
Transaction ID: 64
Amount: 3717
Account ID: 55149
Transaction Date: 05-MAY-24
Transaction ID: 233
Amount: 3369
Account ID: 550
Transaction Date: 19-MAY-24
Transaction ID: 246
Amount: 4044
Account ID: 39364
Transaction Date: 16-MAY-24
Transaction ID: 122
Amount: 2549
Account ID: 243
Transaction Date: 06-MAY-24
Transaction ID: 802
Amount: 2834
Account ID: 94
Transaction Date: 12-MAY-24
```

4. שאילתת כרטיסי אשראי עם תאריך תפוגה בטווח מסוים:

```
DECLARE
    p_Start_Expiration DATE := TO_DATE('2022-01-01', 'YYYY-MM-DD');
    p_End_Expiration DATE := TO_DATE('2024-12-31', 'YYYY-MM-DD');
    CURSOR c_Credit_Cards IS
        SELECT Card_ID, Card_Number, Card_Type, Expiration_Date, Credit_Limit, Account_ID
        FROM Credit_Cards
        WHERE Expiration_Date BETWEEN p_Start_Expiration AND p_End_Expiration;
BEGIN
    FOR rec IN c_Credit_Cards LOOP
        DBMS_OUTPUT.PUT_LINE('Card ID: ' || rec.Card_ID);
        DBMS_OUTPUT.PUT_LINE('Card Number: ' || rec.Card_Number);
        DBMS_OUTPUT.PUT_LINE('Card Type: ' || rec.Card_Type);
        DBMS_OUTPUT.PUT_LINE('Expiration Date: ' || rec.Expiration_Date);
        DBMS_OUTPUT.PUT_LINE('Credit Limit: ' || rec.Credit_Limit);
        DBMS_OUTPUT.PUT_LINE('Account ID: ' || rec.Account_ID);
    END LOOP;
END;
/
```

```
Account ID: 97
Card ID: 17525
Card Number: 8139362
Card Type: visa
Expiration Date: 07-MAY-24
Credit Limit: 6875
Account ID: 833
Card ID: 31661
Card Number: 1756831
Card Type: mastercard
Expiration Date: 07-MAY-24
Credit Limit: 6719
Account ID: 916
Card ID: 71043
Card Number: 1773869
Card Type: mastercard
Expiration Date: 18-MAY-24
Credit Limit: 6467
Account ID: 916
Card ID: 26892
Card Number: 2920466
Card Type: visa
Expiration Date: 06-MAY-24
Credit Limit: 6121
Account ID: 38851
```

נדגים את קובץ האילוצים:

SQL Output Statistics

```sql
ALTER TABLE Loans
MODIFY (Interest_Rate NOT NULL);


ALTER TABLE Credit_CaDECLARE
ADD CONSTRAINT chk_Credit_Limit CHECK (Credit_Limit > 0);


ALTER TABLE Deposits
MODIFY (Interest_Rate DEFAULT 5);
```

Alter loans | Alter credit_cards | Alter deposits

שמנו אילוץ שבכום הריבית בהלוואה יהיה לא null
הגבלת אשראי גדולה מ0
ברירת המחדל של ריבית על הפקדות היא 5

נראה דוגמא של הכנסה לטבלה בלי ערך ריבית 👍

```
INSERT INTO Deposits (DepositID, Deposit_Amount, Deposit_Date, Maturity_Date, Account_ID)
VALUES (51510, 7000, DATE '2024-05-23', DATE '2025-05-28', 10);

select * from deposits
where depositid = 51510
```

| | DEPOSITID | DEPOSIT_AMOUNT | DEPOSIT_DATE | MATURITY_DATE | INTEREST_RATE | ACCOUNT_ID |
|---|---|---|---|---|---|---|
| 1 | 51510 | 7000 | 23/05/2024 | 28/05/2025 | 5 | 10 |

באופן אוטומטי הערך הוא 5.

נראה דוגמא של הכנסת מגבלת אשראי קטנה מ0:

```
INSERT INTO Credit_Cards (Card_ID, Card_Number, Card_Type, Expiration_Date, Credit_Limit, Account_ID)
VALUES (3154201, 334455667788, 'Mastercard', DATE '2025-05-30', -7000, 10);

select * from Credit_Cards
where Card_ID = 3154201
```

Error

ORA-02290: check constraint (SYS.CHK_CREDIT_LIMIT) violated

OK Cancel Help

ובכן קיבלנו הודעת שגיאה כמצופה.

חזק וברוך עד כאן שלב 2.