

פונקציה 1:

חישוב יתרה ממוצעת של כל החשבונות בסניף

הפונקציה תחזיר את היתרה הממוצעת של כל החשבונות בסניף מסוים.

הסבר על הפונקציה-

קלט: p_branch_id - מזהה הסניף.

משתנים:

v_total_balance: סכום כל היתרות.

v_account_count: מספר החשבונות.

v_balance: יתרת החשבון הנוכחי.

v_average_balance: יתרת הממוצעת המחושבת.

c_accounts: Cursor לקריאת היתרות של החשבונות בסניף.

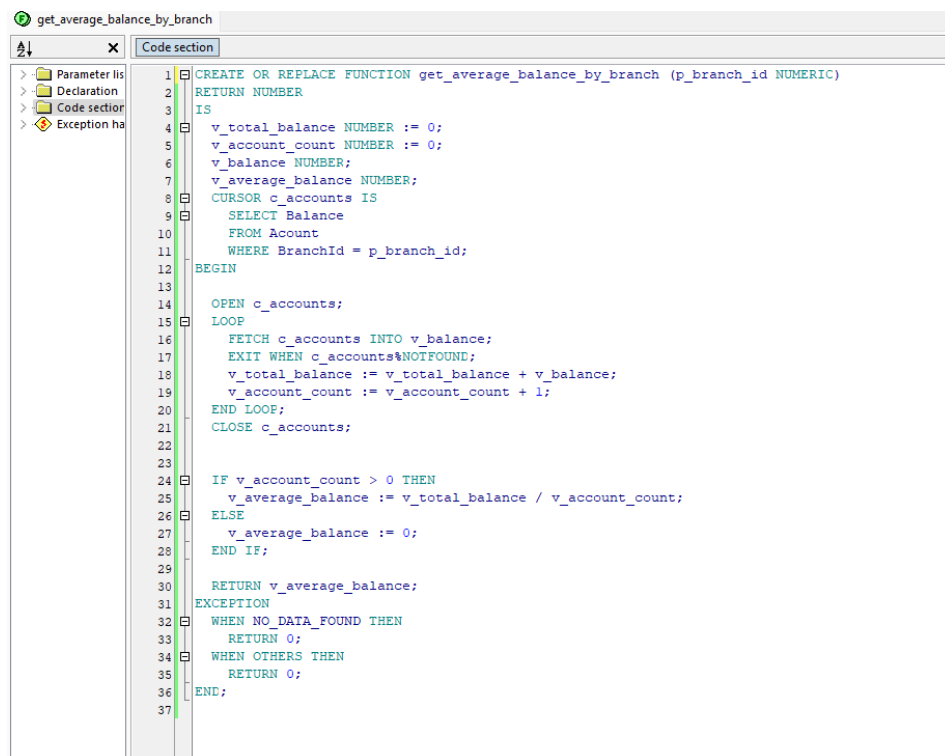
שלבים:

פתיחת ה-Cursor ולולאה דרכו כדי לאסוף את היתרות.

חישוב יתרת הממוצעת.

החזרת יתרת הממוצעת.

טיפול בשגיאות: במקרה של שגיאה מחזירה 0.



```
1 CREATE OR REPLACE FUNCTION get_average_balance_by_branch (p_branch_id NUMERIC)
2 RETURN NUMBER
3 IS
4     v_total_balance NUMBER := 0;
5     v_account_count NUMBER := 0;
6     v_balance NUMBER;
7     v_average_balance NUMBER;
8     CURSOR c_accounts IS
9         SELECT Balance
10        FROM Account
11       WHERE BranchId = p_branch_id;
12 BEGIN
13
14     OPEN c_accounts;
15     LOOP
16         FETCH c_accounts INTO v_balance;
17         EXIT WHEN c_accounts%NOTFOUND;
18         v_total_balance := v_total_balance + v_balance;
19         v_account_count := v_account_count + 1;
20     END LOOP;
21     CLOSE c_accounts;
22
23     IF v_account_count > 0 THEN
24         v_average_balance := v_total_balance / v_account_count;
25     ELSE
26         v_average_balance := 0;
27     END IF;
28
29     RETURN v_average_balance;
30 EXCEPTION
31     WHEN NO_DATA_FOUND THEN
32         RETURN 0;
33     WHEN OTHERS THEN
34         RETURN 0;
35 END;
```

405	405	200	123456	1	AAAE5VAABAAAK+iABQ ...
-----	-----	-----	--------	---	------------------------

179	99	756221	528163	1	AAAE5VAABAAK+hACy ...
-----	----	--------	--------	---	-----------------------

80	181	668519	443889	119	AAAE5VAABAAK+hABP ...
81	1	200	123456	1	AAAE5VAABAAK+hABQ ...
82	2	0	654321	2	AAAE5VAABAAK+hABR ...
83	3	1234	615243	1	AAAE5VAABAAK+hABS ...

Test script
DBMS Output
Statistics
Profiler
Trace
205

```

1 begin
2   -- Call the function
3   :result := get_average_balance_by_branch(p_branch_id => :p_branch_id);
4 end;

```

Variable	Type	Value
result	Float	189463.75
p_branch_id	Float	1
*		

פונקציה 2:

פונקציה להחזרת שמות לקוחות המחזיקים בחשבון מסוים

הסבר על הפונקציה-

קלט: p_account_id - מזהה החשבון.

משתנים:

client_record: סוג רשומה להכיל נתוני לקוח.

v_client_list: מחרוזת המכילה את רשימת השמות.

v_client: משתנה מסוג client_record.

v_cursor: SYS_REFCURSOR לקריאת נתוני הלקוחות.

שלבים:

פתיחת ה-Cursor וקריאת נתוני הלקוחות הקשורים לחשבון.

בניית מחרוזת של שמות הלקוחות.

סגירת ה-Cursor והחזרת המחרוזת.

טיפול בשגיאות: סגירת ה-Cursor אם הוא פתוח והחזרת NULL במקרה של שגיאה,

get_account_clients

Parameter list

Declaration

Code section

Exception handler

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

```
CREATE OR REPLACE FUNCTION get_account_clients(p_account_id NUMERIC)
RETURN VARCHAR2
IS
TYPE client_record IS RECORD (
  client_id NUMERIC(5),
  client_name VARCHAR2(30),
  client_last_name VARCHAR2(30)
);
v_client_list VARCHAR2(4000) := '';
v_client client_record;
v_cursor SYS_REFCURSOR;

BEGIN
  OPEN v_cursor FOR
  SELECT c.ClientId, c.ClientName, c.ClientLastName
  FROM Client c
  JOIN activity a ON c.ClientId = a.ClientId
  WHERE a.AccountId = p_account_id;

  LOOP
    FETCH v_cursor INTO v_client;

    EXIT WHEN v_cursor%NOTFOUND;
    IF v_client_list IS NOT NULL THEN
      v_client_list := v_client_list || ', ';
    END IF;

    v_client_list := v_client_list || v_client.client_name || ' ' || v_client.client_last_name;
  END LOOP;

  CLOSE v_cursor;

  RETURN v_client_list;
EXCEPTION
  WHEN OTHERS THEN
    IF v_cursor%ISOPEN THEN
      CLOSE v_cursor;
    END IF;
    RETURN NULL;
END;
```

Test script DBMS Output Statistics Profiler Trace

1

2

3

4

```
begin
  -- Call the function
  :result := get_account_clients(p_account_id => :p_account_id);
end;
```

Variable

Type

Value

result

String

Patty Emmerich, Kimberly Tate

p_account_id

Float

12

	ACTIVITYID	CLIENTID	ACTIONID	ACCOUNTID	WORKERID	ROWID
1	609	235	340	12	47	AAAE5bAABAAK/aACH ...
2	745	144	319	12	199	AAAE5bAABAAK/aAEP ...

	CLIENTID	CLIENTNAME	CLIENTADDRESS	CLIENTLASTNAME	ROWID
1	144	Patty	113 Curtis-Hall	Emmerich	AAAE5ZAABAAK/BAAo ...

	CLIENTID	CLIENTNAME	CLIENTADDRESS	CLIENTLASTNAME	ROWID
1	235	Kimberly	71 Flack Road	Tate	AAAE5ZAABAAK/DAAq ...

פרוצדורה 1:

עדכון יתרה בחשבון לאחר פעולה

הפרוצדורה תעדכן את יתרת החשבון לאחר ביצוע פעולה (הפקדה או משיכה).

קלט:

p_account_id - מזהה החשבון.

p_action_name - שם הפעולה (Deposit או Withdrawing).

p_action_sum - סכום הפעולה.

משתנים:

v_balance: יתרת החשבון הנוכחית.

שלבים:

שליפת יתרת החשבון הנוכחית.

עיבוד הפעולה (הפקדה או משיכה) ועדכון היתרה.

עדכון יתרת החשבון במסד הנתונים.

טיפול בשגיאות: הדפסת הודעת שגיאה.

```
update_account_balance
Code section | if
1 CREATE OR REPLACE PROCEDURE update_account_balance (
2   p_account_id NUMERIC,
3   p_action_name VARCHAR2,
4   p_action_sum INT
5 )
6 IS
7   v_balance INT;
8 BEGIN
9   -- שליפת יתרת החשבון הנוכחית
10  SELECT Balance INTO v_balance
11  FROM Account
12  WHERE AccountId = p_account_id;
13
14  -- עיבוד הפעולה ועדכון היתרה
15  IF p_action_name = 'Deposit' THEN
16    v_balance := v_balance + p_action_sum;
17  ELSIF p_action_name = 'Withdrawing Funds' THEN
18    IF v_balance >= p_action_sum THEN
19      v_balance := v_balance - p_action_sum;
20    ELSE
21      RAISE_APPLICATION_ERROR(-20001, 'Insufficient balance for withdrawal.');
```

select t.*, t.rowid from ACCOUNT t

	ACCOUNTID	BALANCE	ACCOUNTNUMBER	BRANCHID	ROWID
1	102	758435	927934	254	AAAE5VAABAAAK+hAAA
2	103	242779	491276	377	AAAE5VAABAAAK+hAAB
3	104	132936	973559	65	AAAE5VAABAAAK+hAAC
4	105	346754	990328	144	AAAE5VAABAAAK+hAAD
5	106	382756	844587	247	AAAE5VAABAAAK+hAAE
6	107	614219	664427	390	AAAE5VAABAAAK+hAAF
7	108	635030	139149	312	AAAE5VAABAAAK+hAAG

```

begin
-- Call the procedure
update_account_balance(p_account_id => :p_account_id,
p_action_name => :p_action_name,
p_action_sum => :p_action_sum);
end;

```

Variable	Type	Value
p_account_id	Float	103
p_action_name	String	Withdrawing Funds
p_action_sum	Float	10000

Output Statistics

select t.*, t.rowid from ACCOUNT t

	ACCOUNTID	BALANCE	ACCOUNTNUMBER	BRANCHID	ROWID
1	102	758435	927934	254	AAAE5VAABAAAK+hAAA
2	103	232779	491276	377	AAAE5VAABAAAK+hAAB
3	104	132936	973559	65	AAAE5VAABAAAK+hAAC
4	105	346754	990328	144	AAAE5VAABAAAK+hAAD

פרוצדורה 2:

יצירת לקוח חדש עם חשבון ראשוני

הפרוצדורה תיצור רשומת לקוח חדשה ורשומת חשבון חדשה עבורו, עם יתרה ראשונית מסוימת.

קלט:

p_client_name - שם הלקוח.

p_client_last_name - שם משפחה של הלקוח.

p_client_address - כתובת הלקוח.

p_initial_balance - יתרת התחלתית בחשבון.

p_branch_id - מזהה הסניף.

משתנים:

v_client_id :מזהה הלקוח החדש.

v_account_id :מזהה החשבון החדש.

שלבים:

יצירת מזהה לקוח חדש.

הוספת לקוח חדש לטבלת הלקוחות.

יצירת מזהה חשבון חדש.

הוספת חשבון חדש ללקוח החדש.

ביצוע COMMIT.

טיפול בשגיאות: הדפסת הודעת שגיאה.

```
1 CREATE OR REPLACE PROCEDURE create_client_with_account(p_client_name VARCHAR2, p_client_last_name VARCHAR2, p_client_address VARCHAR2,
2 p_initial_balance INT,
3 p_branch_id NUMERIC) IS
4
5     v_client_id NUMERIC;
6     v_account_id NUMERIC;
7     v_exists NUMERIC;
8
9 BEGIN
10     v_exists := 0;
11     begin
12         select 1
13         into v_exists
14         from Client
15         where ClientName = p_client_name
16               and ClientLastName = p_client_last_name
17               and rownum = 1;
18     EXCEPTION
19     WHEN OTHERS THEN
20         NULL;
21     end;
22     if v_exists = 0 then
23         -- Generate a new client ID
24         SELECT NVL(MAX(ClientId), 0) + 1 INTO v_client_id FROM Client;
25         -- Insert a new client record
26         INSERT INTO Client
27         (ClientId, ClientName, ClientLastName, ClientAddress)
28         VALUES
29         (v_client_id, p_client_name, p_client_last_name, p_client_address);
30         -- Generate a new account ID
31         SELECT NVL(MAX(AccountId), 0) + 1 INTO v_account_id FROM Account;
32         -- Insert a new account record for the client
33         INSERT INTO Account
34         (AccountId, Balance, AccountNumber, BranchId)
35         VALUES
36         (v_account_id, p_initial_balance, v_account_id + 1000, p_branch_id);
37         COMMIT;
38         DBMS_OUTPUT.PUT_LINE('Client ' || p_client_name || ' ' || p_client_last_name || ' has been created with a new account.');
```

Test Window - Script for procedure CREATE_CLIENT_WITH_ACCOUNT@XE

Test script DBMS Output Statistics Profiler Trace

```
1 begin
2     -- Call the procedure
3     create_client_with_account(p_client_name => :p_client_name,
4                               p_client_last_name => :p_client_last_name,
5                               p_client_address => :p_client_address,
6                               p_initial_balance => :p_initial_balance,
7                               p_branch_id => :p_branch_id);
8 end;
```

Variable	Type	Value
p_client_name	String	Reut
p_client_last_name	String	Agam
p_client_address	String	Noyfeld 8, Beny Brak
p_initial_balance	Float	100000
p_branch_id	Float	66

select t.*, t.rowid from CLIENT t					
	CLIENTID	CLIENTNAME	CLIENTADDRESS	CLIENTLASTNAME	ROWID
254	582	Steffane	25 Tomscot Plaza	Jeffes	AAAE5ZAABAAAK/CAA/
255	472	Shara	96 Thackeray Court	Benninger	AAAE5ZAABAAAK/CABA
256	466	Myrtle	948 David Terrace	Dentith	AAAE5ZAABAAAK/CABB
257	556	Cell	2 Shoshone Avenue	Mor	AAAE5ZAABAAAK/CABC
258	469	Skelly	556 Arrowood Center	Alben	AAAE5ZAABAAAK/CABD
259	611	Kelcey	59358 Shasta Street	Abraham	AAAE5ZAABAAAK/CABE
260	679	Gnni	7 Lindbergh Terrace	Smidmore	AAAE5ZAABAAAK/CABF
261	646	Pascal	33166 Golf Course Circle	Rawlings	AAAE5ZAABAAAK/CABG
262	607	Chucho	4608 Farwell Park	Gheerhaert	AAAE5ZAABAAAK/CABH
263	579	Phedra	99 Dapin Avenue	Ausher	AAAE5ZAABAAAK/CABI
264	680	Reut	Noyfeld 8, Beny Brak	Agam	AAAE5ZAABAAAK/CABJ
265	193	Maceo	41 Chemnitz Street	Mueller-Stahl	AAAE5ZAABAAAK/DAAA

402	402	50748	530794	239	AAAE5VAABAAAK+iABN
403	403	957002	743297	11	AAAE5VAABAAAK+iABO
404	404	791167	942120	317	AAAE5VAABAAAK+iABP
405	405	200	123456	1	AAAE5VAABAAAK+iABQ
406	406	0	654321	2	AAAE5VAABAAAK+iABR
407	407	0	777771	2	AAAE5VAABAAAK+iABS
408	408	100000	1408	66	AAAE5VAABAAAK+iABT

תוכנית ראשית 1:
מזמנת את הפונקציה - get_average_balance_by_branch

ואת הפרוצדורה - create_client_with_account

```

DECLARE
v_branch_id NUMERIC := 65; -- Example branch ID
v_average_balance NUMBER;
v_client_name VARCHAR2(30) := 'Reut'; -- Example client name
v_client_last_name VARCHAR2(30) := 'Agam'; -- Example client last name
v_client_address VARCHAR2(100) := 'simchoni 1, Bney Brak'; -- Example client address
v_initial_balance INT := 20000; -- Example initial balance
v_branch_id_create NUMERIC := 2; -- Branch ID for account creation

BEGIN
-- Calling the function to get the average balance by branch
v_average_balance := get_average_balance_by_branch(v_branch_id);
DBMS_OUTPUT.PUT_LINE('Average balance for branch ' || v_branch_id || ' is: ' || v_average_balance);

-- Calling the procedure to create a new client with an account
create_client_with_account(
    p_client_name => v_client_name,
    p_client_last_name => v_client_last_name,
    p_client_address => v_client_address,
    p_initial_balance => v_initial_balance,
    p_branch_id => v_branch_id_create
);
DBMS_OUTPUT.PUT_LINE('Client ' || v_client_name || ' ' || v_client_last_name || ' has been created with a new account.');
```

```

EXCEPTION
WHEN OTHERS THEN
    DBMS_OUTPUT.PUT_LINE('Error: ' || SQLERRM);
END;
```

```

Average balance for branch 65 is: 132936
Client Reut Agam has been created with a new account.
```

	CLIENTID	CLIENTNAME	CLIENTADDRESS	CLIENTLASTNAME	ROWID
1	2	Reut	RabiAkiva 11,Bnei Brak	Agam	AAAE5ZAABAAAK/BABB
2	680	Reut	Noyfeld 8, Beny Brak	Agam	AAAE5ZAABAAAK/CABJ
3	687	Reut	simchoni 1, Bney Brak	Agam	AAAE5ZAABAAAK/CABQ

	ACCOUNTID	BALANCE	ACCOUNTNUMBER	BRANCHID	ROWID
▶ 1	104	132936	973559	65	AAAE5VAABAAAK+hAAC ...

תוכנית ראשית 2:

מזמנת את הפונקציה - get_account_clients

ואת הפרוצדורה - update_account_balance

```

1 DECLARE
2   v_account_id NUMERIC := 15; -- Example account ID
3   v_client_list VARCHAR2(4000);
4   v_action_name VARCHAR2(15) := 'Withdrawing'; -- Example action
5   v_action_sum INT := 1500; -- Example action sum
6   v_update_account_id NUMERIC := 105; -- Example account ID for update
7
8 BEGIN
9   -- Calling the function to get account clients
10  v_client_list := get_account_clients(v_account_id);
11  DBMS_OUTPUT.PUT_LINE('Clients for account ' || v_account_id || ': ' || v_client_list);
12
13  -- Calling the procedure to update account balance
14  update_account_balance(p_account_id => v_update_account_id, p_action_name => v_action_name, p_action_sum => v_action_sum);
15  DBMS_OUTPUT.PUT_LINE('Account ' || v_update_account_id || ' has been updated successfully.');
```

Clients for account 15: Giovanni Cruz, Phil Rucker
Account 105 has been updated successfully.

	ACTIVITYID	CLIENTID	ACTIONID	ACCOUNTID	WORKERID	ROWID
▶ 1	538	179	27	15	352	AAAE5bAABAAAK/aABA ...
▶ 2	758	225	186	15	391	AAAE5bAABAAAK/aAEc ...

	CLIENTID	CLIENTNAME	CLIENTADDRESS	CLIENTLASTNAME	ROWID
▶ 1	225	Phil	40 Kingsley Street	Rucker	AAAE5ZAABAAAK/DAAg ...

	CLIENTID	CLIENTNAME	CLIENTADDRESS	CLIENTLASTNAME	ROWID
1	179	Giovanni	487 Tinsley Drive	Cruz	AAAE5ZAABAAAK/BACw

1	102	4418435	927934	254	AAAE5VAABAAAK+hAAA
2	103	232779	491276	377	AAAE5VAABAAAK+hAAB
3	104	132936	973559	65	AAAE5VAABAAAK+hAAC
4	105	346754	990328	144	AAAE5VAABAAAK+hAAD
5	106	382756	844587	247	AAAE5VAABAAAK+hAAE
6	107	614219	664427	390	AAAE5VAABAAAK+hAAF

SQL Output Statistics

```
select t.*, t.rowid from ACOUNT t
```

	ACCOUNTID	BALANCE	ACCOUNTNUMBER	BRANCHID	ROWID
1	102	4418435	927934	254	AAAE5VAABAAAK+hAAA
2	103	232779	491276	377	AAAE5VAABAAAK+hAAB
3	104	132936	973559	65	AAAE5VAABAAAK+hAAC
4	105	345254	990328	144	AAAE5VAABAAAK+hAAD
5	106	382756	844587	247	AAAE5VAABAAAK+hAAE
6	107	614219	664427	390	AAAE5VAABAAAK+hAAF
7	108	635030	139149	312	AAAE5VAABAAAK+hAAG