**Lab5. Creating Functions and Packages**

1. Create at least 3 functions for your database. You may use the anonymous blocks you created in Lab 2 as a basis.
2. At least 2 functions must be with parameters.
3. Combine your procedures (Lab 4) and functions (Lab 5) into at least 2 packages. Procedures and functions of one package must be logically related.
4. During the defense you must demonstrate the execution of procedures and functions.

CREATE OR REPLACE FUNCTION GET\_USER\_BY\_CARD\_NUMBER(

p\_card\_number IN VARCHAR2

) RETURN Users%ROWTYPE AS

v\_user\_details Users%ROWTYPE;

BEGIN

SELECT u.\*

INTO v\_user\_details

FROM Cards c

JOIN Users u ON c.user\_id = u.user\_id

WHERE c.card\_number = p\_card\_number;

RETURN v\_user\_details;

EXCEPTION

WHEN NO\_DATA\_FOUND THEN

RETURN NULL;

END GET\_USER\_BY\_CARD\_NUMBER;

/

DECLARE

v\_user\_info Users%ROWTYPE;

BEGIN

v\_user\_info := GET\_USER\_BY\_CARD\_NUMBER('1234566660123456');

DBMS\_OUTPUT.PUT\_LINE('User ID: ' || v\_user\_info.user\_id);

DBMS\_OUTPUT.PUT\_LINE('Username: ' || v\_user\_info.uname);

DBMS\_OUTPUT.PUT\_LINE('Last Name: ' || v\_user\_info.ulastname);

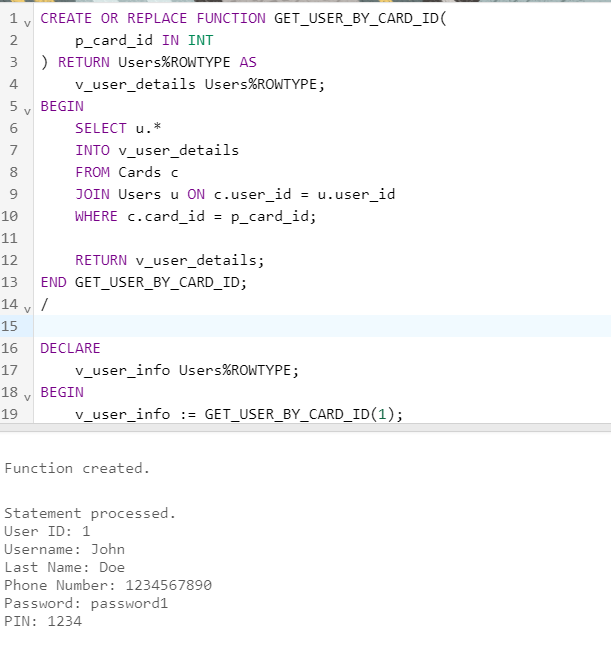
DBMS\_OUTPUT.PUT\_LINE('Phone Number: ' || NVL(v\_user\_info.uphone\_number, 'N/A'));

DBMS\_OUTPUT.PUT\_LINE('Password: ' || v\_user\_info.user\_password);

DBMS\_OUTPUT.PUT\_LINE('PIN: ' || v\_user\_info.user\_pin);

END;

/



CREATE OR REPLACE FUNCTION GET\_USER\_BY\_CARD\_ID(

p\_card\_id IN INT

) RETURN Users%ROWTYPE AS

v\_user\_details Users%ROWTYPE;

BEGIN

SELECT u.\*

INTO v\_user\_details

FROM Cards c

JOIN Users u ON c.user\_id = u.user\_id

WHERE c.card\_id = p\_card\_id;

RETURN v\_user\_details;

END GET\_USER\_BY\_CARD\_ID;

/

DECLARE

v\_user\_info Users%ROWTYPE;

BEGIN

v\_user\_info := GET\_USER\_BY\_CARD\_ID(1);

DBMS\_OUTPUT.PUT\_LINE('User ID: ' || v\_user\_info.user\_id);

DBMS\_OUTPUT.PUT\_LINE('Username: ' || v\_user\_info.uname);

DBMS\_OUTPUT.PUT\_LINE('Last Name: ' || v\_user\_info.ulastname);

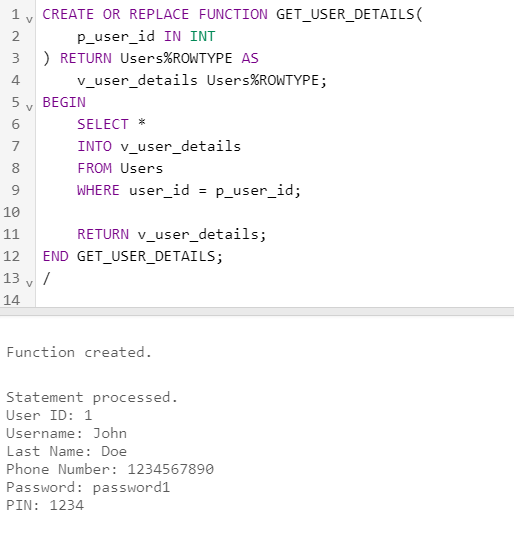
DBMS\_OUTPUT.PUT\_LINE('Phone Number: ' || NVL(v\_user\_info.uphone\_number, 'N/A'));

DBMS\_OUTPUT.PUT\_LINE('Password: ' || v\_user\_info.user\_password);

DBMS\_OUTPUT.PUT\_LINE('PIN: ' || v\_user\_info.user\_pin);

END;

/



CREATE OR REPLACE FUNCTION GET\_USER\_DETAILS(

p\_user\_id IN INT

) RETURN Users%ROWTYPE AS

v\_user\_details Users%ROWTYPE;

BEGIN

SELECT \*

INTO v\_user\_details

FROM Users

WHERE user\_id = p\_user\_id;

RETURN v\_user\_details;

END GET\_USER\_DETAILS;

/

DECLARE

v\_user\_details Users%ROWTYPE;

BEGIN

v\_user\_details := GET\_USER\_DETAILS(1);

DBMS\_OUTPUT.PUT\_LINE('User ID: ' || v\_user\_details.user\_id);

DBMS\_OUTPUT.PUT\_LINE('Username: ' || v\_user\_details.uname);

DBMS\_OUTPUT.PUT\_LINE('Last Name: ' || v\_user\_details.ulastname);

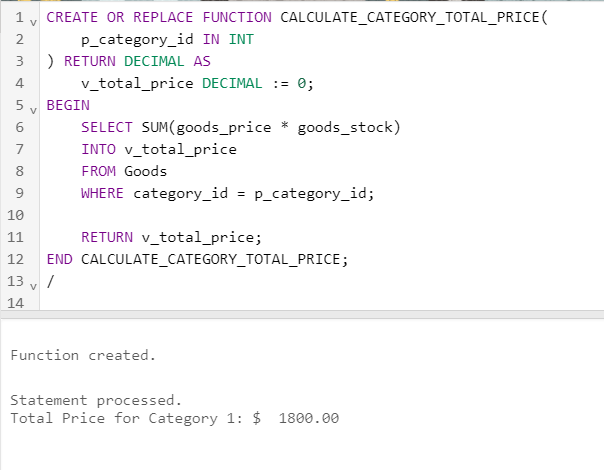
DBMS\_OUTPUT.PUT\_LINE('Phone Number: ' || NVL(v\_user\_details.uphone\_number, 'N/A'));

DBMS\_OUTPUT.PUT\_LINE('Password: ' || v\_user\_details.user\_password);

DBMS\_OUTPUT.PUT\_LINE('PIN: ' || v\_user\_details.user\_pin);

END;

/



CREATE OR REPLACE FUNCTION CALCULATE\_CATEGORY\_TOTAL\_PRICE(

p\_category\_id IN INT

) RETURN DECIMAL AS

v\_total\_price DECIMAL := 0;

BEGIN

SELECT SUM(goods\_price \* goods\_stock)

INTO v\_total\_price

FROM Goods

WHERE category\_id = p\_category\_id;

RETURN v\_total\_price;

END CALCULATE\_CATEGORY\_TOTAL\_PRICE;

/

DECLARE

v\_total\_price DECIMAL;

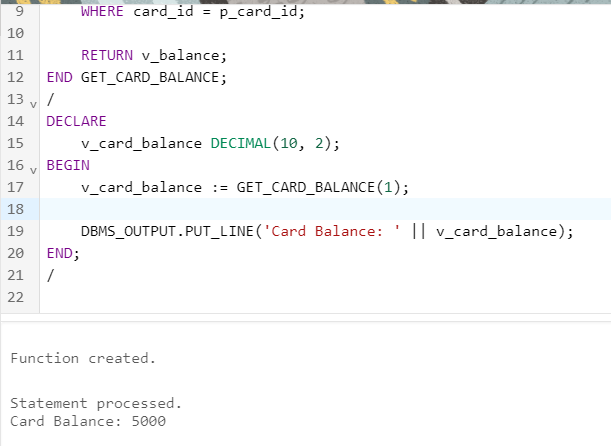
BEGIN

v\_total\_price := CALCULATE\_CATEGORY\_TOTAL\_PRICE(1);

DBMS\_OUTPUT.PUT\_LINE('Total Price for Category 1: $' || TO\_CHAR(v\_total\_price, '99999.99'));

END;

/



CREATE OR REPLACE FUNCTION GET\_CARD\_BALANCE(

p\_card\_id IN INT

) RETURN DECIMAL AS

v\_balance DECIMAL(10, 2);

BEGIN

SELECT balance

INTO v\_balance

FROM Cards

WHERE card\_id = p\_card\_id;

RETURN v\_balance;

END GET\_CARD\_BALANCE;

/

DECLARE

v\_card\_balance DECIMAL(10, 2);

BEGIN

v\_card\_balance := GET\_CARD\_BALANCE(1);

DBMS\_OUTPUT.PUT\_LINE('Card Balance: ' || v\_card\_balance);

END;

/

Questions:

1. Define a function in PL/SQL.
2. Different ways of invoking functions.
3. Packages and their structure.
4. The purpose of creating packages.
5. Private and public components of the package.
6. What is package specification?
7. What is a package body?
8. What is the difference between a procedure declared in the package specification and a procedure declared only in the package body?
9. Explain the notion of the forward declaration.
10. In which cases the forward declaration must be used?
11. What is overloading? And what are its limitations?
12. What is the package state?
13. How is a package created?
14. Is it allowed to initialize variables in the package specification?
15. How are methods invoked inside the package where it is declared?
16. How are methods invoked outside of the package where it is declared?
17. Define the notion of bodiless packages.
18. Describe the BEGIN section in a package.