

CSD 4203 – Database Programming

Term : 2023S

Student ID : C0930321

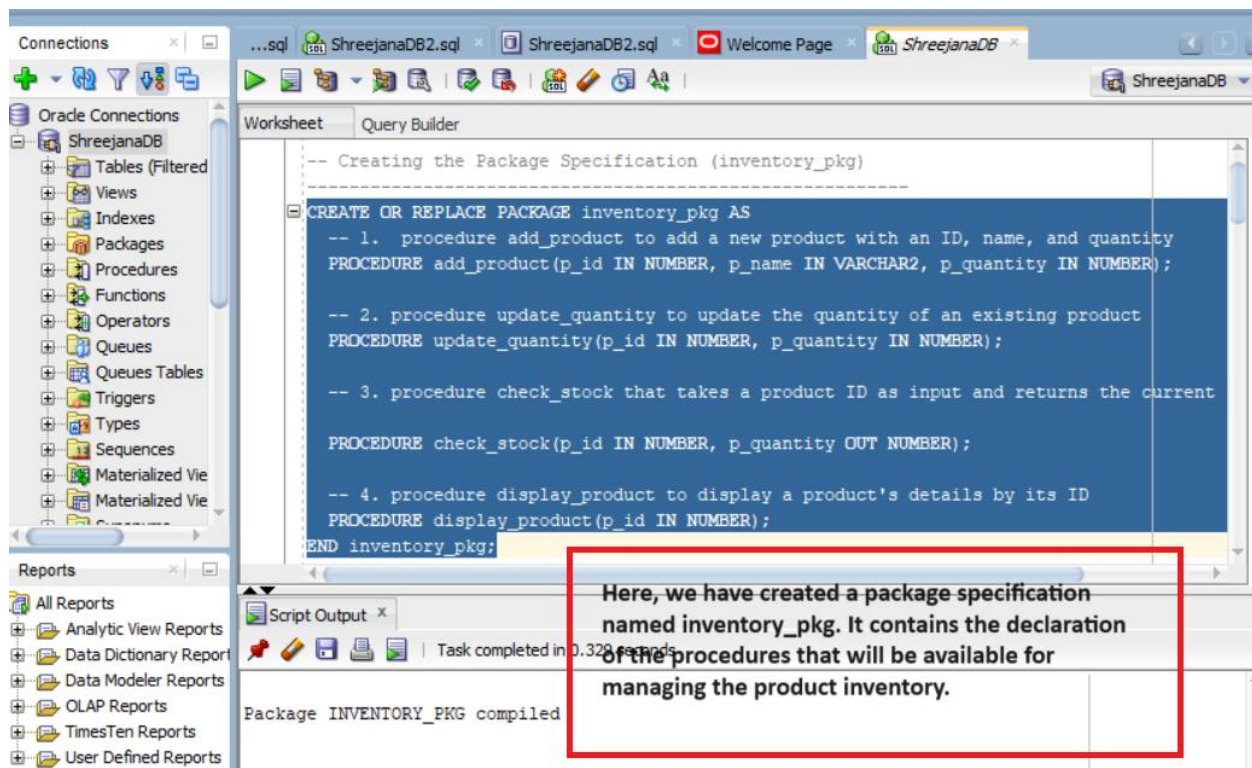
Student Name : Shreejana Shrestha

Assignment # 9

PL/ SQL – Package

Package for Inventory Management

Creation of the package



The screenshot displays the Oracle SQL Developer interface. The left pane shows the 'Connections' tree with 'ShreejanaDB' selected. The main workspace is titled 'Worksheet' and contains the following SQL code:

```
-- Creating the Package Specification (inventory_pkg)

CREATE OR REPLACE PACKAGE inventory_pkg AS
-- 1. procedure add_product to add a new product with an ID, name, and quantity
PROCEDURE add_product(p_id IN NUMBER, p_name IN VARCHAR2, p_quantity IN NUMBER);

-- 2. procedure update_quantity to update the quantity of an existing product
PROCEDURE update_quantity(p_id IN NUMBER, p_quantity IN NUMBER);

-- 3. procedure check_stock that takes a product ID as input and returns the current
PROCEDURE check_stock(p_id IN NUMBER, p_quantity OUT NUMBER);

-- 4. procedure display_product to display a product's details by its ID
PROCEDURE display_product(p_id IN NUMBER);
END inventory_pkg;
```

The 'Script Output' pane at the bottom shows the message: 'Package INVENTORY_PKG compiled'. A red rectangular box highlights the text: 'Here, we have created a package specification named inventory_pkg. It contains the declaration of the procedures that will be available for managing the product inventory.'

Creation of the products table

The screenshot displays the Oracle SQL Developer interface. The left sidebar shows the 'Connections' pane with 'ShreejanaDB' selected. The main workspace is divided into 'Worksheet' and 'Query Builder' tabs. The 'Worksheet' tab contains the following SQL code:

```
-- Creating the products table  
  
CREATE TABLE products (  
  product_id NUMBER PRIMARY KEY,  
  product_name VARCHAR2(100),  
  quantity NUMBER  
);
```

A red rectangular box highlights the SQL code, with a text overlay stating: "Here, we have created the products table that stores the product information including product ID, name and quatity." (Note the typo 'quatity' in the original image).

At the bottom, the 'Script Output' pane shows the message: "Table PRODUCTS created." and "Task completed in 0.165 seconds".

Creation of package body with the logic to add/update/display the product records

The screenshot displays the Oracle SQL Developer interface. On the left, the 'Connections' pane shows 'ShreejanaDB' selected. The main workspace is titled 'Worksheet' and contains the following SQL code:

```
CREATE OR REPLACE PACKAGE BODY inventory_pkg AS
-- 1. Procedure to add a new product
PROCEDURE add_product(p_id IN NUMBER, p_name IN VARCHAR2, p_quantity IN NUMBER) IS
BEGIN
    INSERT INTO products (product_id, product_name, quantity)
    VALUES (p_id, p_name, p_quantity);
EXCEPTION
    WHEN DUP_VAL_ON_INDEX THEN
        DBMS_OUTPUT.PUT_LINE('Product with this ID already exists.');
```

The code continues with a procedure to update the quantity of an existing product:

```
PROCEDURE update_quantity(p_id IN NUMBER, p_quantity IN NUMBER) IS
BEGIN
    UPDATE products
    SET quantity = p_quantity
    WHERE product_id = p_id;

    IF SQL%ROWCOUNT = 0 THEN
        DBMS_OUTPUT.PUT_LINE('Product not found.');
```

The code concludes with a procedure to check the stock quantity of a product:

```
END update_quantity;

-- 3. Procedure to check the stock quantity of a product
```

A red oval highlights the entire SQL code block. A red rectangle highlights the following text:

This is the package body provides the implementation of the procedures declared in the package specification

At the bottom, the 'Script Output' pane shows the following message:

Task completed in 0.214 seconds
Package Body INVENTORY_PKG compiled

Connections

Oracle Connections

- ShreejanaDB
 - Tables (Filtered)
 - Views
 - Indexes
 - Packages
 - Procedures
 - Functions
 - Operators
 - Queues
 - Queues Tables
 - Triggers
 - Types
 - Sequences
 - Materialized View
 - Materialized View

Reports

- All Reports
- Analytic View Reports
- Data Dictionary Report
- Data Modeler Reports
- OLAP Reports
- TimesTen Reports
- User Defined Reports

Worksheet

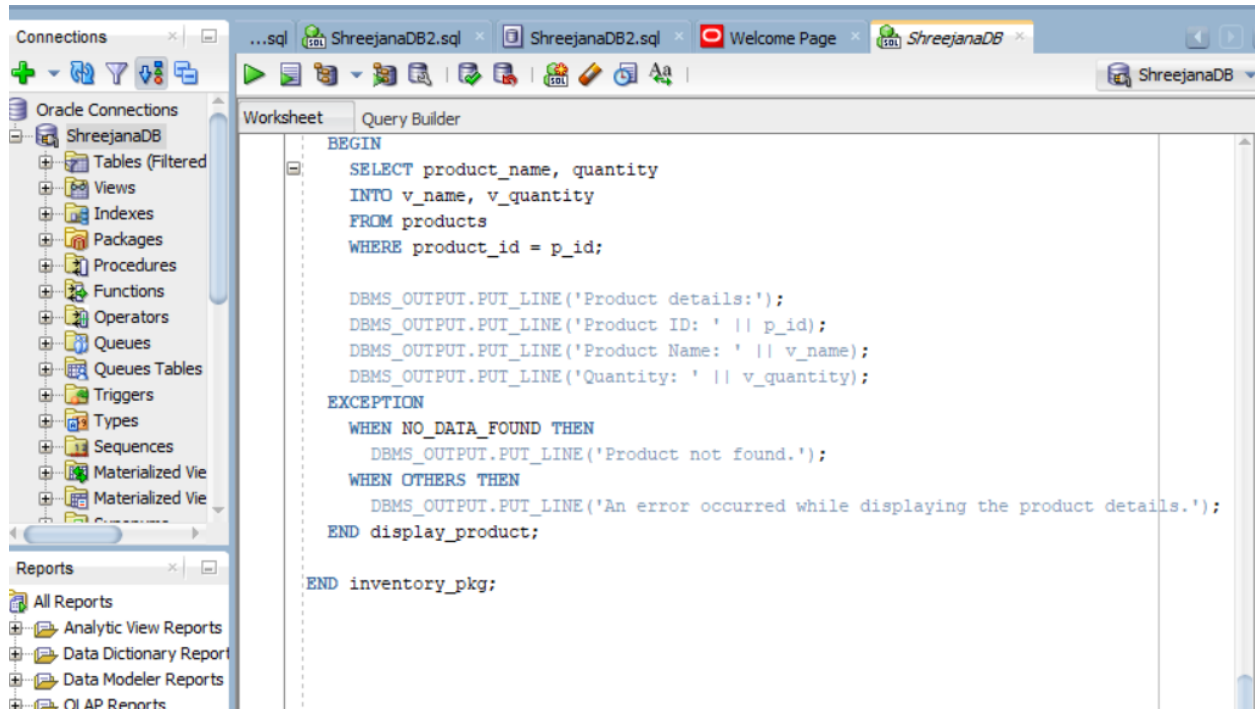
Query Builder

```
-- 3. Procedure to check the stock quantity of a product
PROCEDURE check_stock(p_id IN NUMBER, p_quantity OUT NUMBER) IS
BEGIN
    SELECT quantity
    INTO p_quantity
    FROM products
    WHERE product_id = p_id;

    DBMS_OUTPUT.PUT_LINE('Stock check successful. ');
    DBMS_OUTPUT.PUT_LINE('Product ID: ' || p_id);
    DBMS_OUTPUT.PUT_LINE('Current Quantity: ' || p_quantity);
EXCEPTION
    WHEN NO_DATA_FOUND THEN
        p_quantity := -1; -- Indicates product not found
        DBMS_OUTPUT.PUT_LINE('Product not found. ');
    WHEN OTHERS THEN
        p_quantity := -2; -- Indicates an error occurred
        DBMS_OUTPUT.PUT_LINE('An error occurred while checking the stock. ');
END check_stock;

-- 4. Procedure to display a product's details
PROCEDURE display_product(p_id IN NUMBER) IS
    v_name products.product_name%TYPE;
    v_quantity products.quantity%TYPE;
BEGIN
    SELECT product_name, quantity
    INTO v_name, v_quantity
    FROM products
    WHERE product_id = p_id;

    DBMS_OUTPUT.PUT_LINE('Product details: ');
    DBMS_OUTPUT.PUT_LINE('Product ID: ' || p_id);
    DBMS_OUTPUT.PUT_LINE('Product Name: ' || v_name);
    DBMS_OUTPUT.PUT_LINE('Quantity: ' || v_quantity);
EXCEPTION
```



Implementation of the package

Inserting the product

The screenshot displays the Oracle SQL Developer interface. The left sidebar shows the 'Connections' pane with 'ShreejanaDB' selected, and the 'Reports' pane. The main workspace is the 'Query Builder' window, which contains the following SQL code:

```
END display_product;  
  
END inventory_pkg;  
  
-- Implementation  
-- Implementing the package to add the product  
  
SET SERVEROUTPUT ON;  
  
BEGIN  
    inventory_pkg.add_product(3, 'Product 3', 150);  
END;
```

The code is executed, and the 'Script Output' window at the bottom shows the results:

```
Product added successfully.  
Product ID: 3  
Product Name: Product 3  
Quantity: 150  
  
PL/SQL procedure successfully completed.
```

Annotations in the image highlight the implementation code and the successful execution output.

Here, we are adding a new product to the products table and outputting the details with the success message

Updating the product

The screenshot displays the Oracle SQL Developer interface. On the left, the 'Connections' pane shows 'ShreejanaDB' selected. The main workspace is the 'Query Builder' tab, containing a PL/SQL script. The script has two sections: one for adding a product and another for updating its quantity. The second section is highlighted with a red box. Below the script, the 'Script Output' pane shows the execution results, also highlighted with a red box. A larger red box on the right contains a descriptive text.

```
-- Implementing the package to add the product

SET SERVEROUTPUT ON;

BEGIN
    inventory_pkg.add_product(3, 'Product 3', 150);
END;

-- Updating Quantity

BEGIN
    inventory_pkg.update_quantity(1, 150);
END;
```

Script Output x | Query Result x

Task completed in 0.086 seconds

PL/SQL procedure successfully completed.

Quantity updated successfully.
Product ID: 1
New Quantity: 150

PL/SQL procedure successfully completed.

Here, we are Updating the quantity of an existing product and outputting the new quantity along with the success message

Checking the stock

The screenshot displays the Oracle SQL Developer interface. The left sidebar shows the 'Connections' pane with 'ShreejanaDB' selected, and the 'Reports' pane. The main workspace is the 'Query Builder' tab, showing a PL/SQL script. The script includes a comment '-- Updating Quantity' followed by a procedure call to update the quantity. Below this, a red box highlights a section titled '-- Checking Stock' which contains a PL/SQL block to declare a variable, call a procedure to check the stock, and output the result. The bottom pane shows the 'Script Output' tab with the execution results, also highlighted by a red box. A second red box on the right contains a descriptive text.

```
-- Updating Quantity

BEGIN
    inventory_pkg.update_quantity(1, 150);
END;

-- Checking Stock

DECLARE
    v_quantity NUMBER;
BEGIN
    inventory_pkg.check_stock(1, v_quantity);
    DBMS_OUTPUT.PUT_LINE('Current Stock: ' || v_quantity);
END;
```

Stock check successful.
Product ID: 1
Current Quantity: 150
Current Stock: 150

Here, we have implemented the procedure to check and output the current stock quantity of a product

Displaying the product

The screenshot displays the Oracle SQL Developer interface. On the left, the 'Connections' pane shows 'ShreejanaDB' selected. The main workspace is titled 'Worksheet' and contains a PL/SQL procedure. The procedure is divided into two sections: 'Checking Stock' and 'Displaying product'. The 'Displaying product' section is highlighted with a red box. Below the code, the 'Script Output' pane shows the execution results, also highlighted with a red box. A status bar at the bottom indicates 'Task completed in 0.073 seconds'.

```
-- Checking Stock

DECLARE
    v_quantity NUMBER;
BEGIN
    inventory_pkg.check_stock(1, v_quantity);
    DBMS_OUTPUT.PUT_LINE('Current Stock: ' || v_quantity);
END;

-- Displaying product

BEGIN
    inventory_pkg.display_product(1);
END;
```

Product details:
Product ID: 1
Product Name: Product 1
Quantity: 150

Here, we are implementing the procedure to Display the details of a product by its ID