Jalu Chaudhary

Class IT-314-DL1 **Practice Problems: Procedures**

# Instructions:

Write appropriate PL/SQL programs to complete Hands-On Assignments, Part I, problems 5-5 to 5-9 (five problems) for Chapter 5 given in the textbook.

**Assignment 5-5: Updating Order Status**

Create a procedure named STATUS\_SHIP\_SP that allows an employee in the Shipping

Department to update an order status to add shipping information. The BB\_BASKETSTATUS

table lists events for each order so that a shopper can see the current status, date, and

comments as each stage of the order process is finished. The IDSTAGE column of the

BB\_BASKETSTATUS table identifies each stage; the value 3 in this column indicates that an

order has been shipped.

The procedure should allow adding a row with an IDSTAGE of 3, date shipped, tracking

number, and shipper. The BB\_STATUS\_SEQ sequence is used to provide a value for the primary

key column. Test the procedure with the following information:

Basket # = 3

Date shipped = 20-FEB-12

Shipper = UPS

Tracking # = ZW2384YXK4957

A screenshot of a computer

Description automatically generated

**Assignment 5-6: Returning Order Status Information**

Create a procedure that returns the most recent order status information for a specified basket.

This procedure should determine the most recent ordering-stage entry in the BB\_BASKETSTATUS

table and return the data. Use an IF or CASE clause to return a stage description instead

of an IDSTAGE number, which means little to shoppers. The IDSTAGE column of the

BB\_BASKETSTATUS table identifies each stage as follows:

• 1—Submitted and received

• 2—Confirmed, processed, sent to shipping

• 3—Shipped

• 4—Cancelled

• 5—Back-ordered

The procedure should accept a basket ID number and return the most recent status

description and date the status was recorded. If no status is available for the specified basket

ID, return a message stating that no status is available. Name the procedure STATUS\_SP. Test

the procedure twice with the basket ID 4 and then 6.

A screenshot of a computer

Description automatically generated

**Assignment 5-7: Identifying Customers**

Brewbean’s wants to offer an incentive of free shipping to customers who haven’t returned to

the site since a specified date. Create a procedure named PROMO\_SHIP\_SP that determines

who these customers are and then updates the BB\_PROMOLIST table accordingly. The

procedure uses the following information:

• Date cutoff—Any customers who haven’t shopped on the site since this date

should be included as incentive participants. Use the basket creation date to

reflect shopper activity dates.

• Month—A three-character month (such as APR) should be added to the promotion

table to indicate which month free shipping is effective.

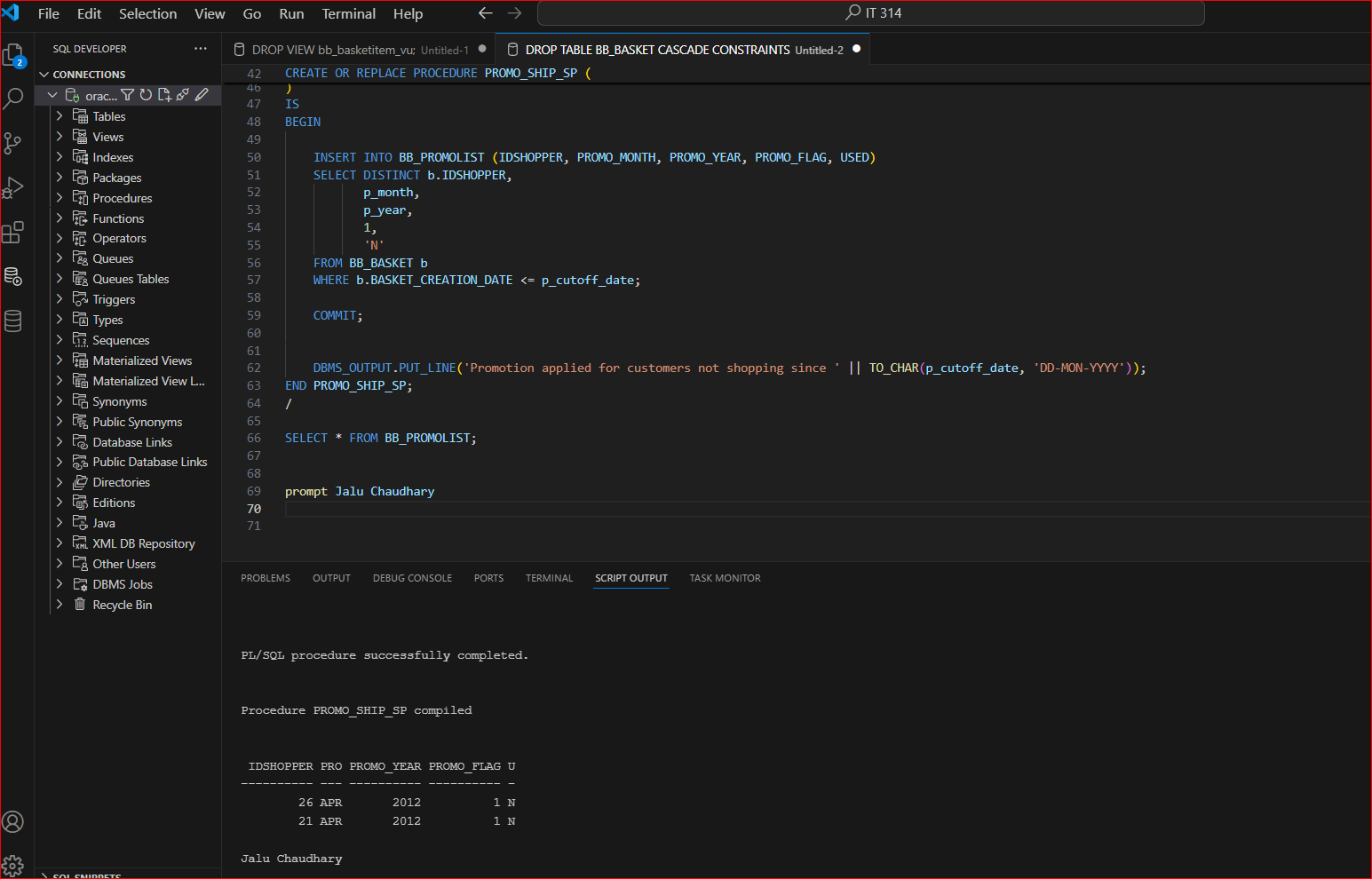
• Year—A four-digit year indicates the year the promotion is effective.

• promo\_flag—1 represents free shipping.

The BB\_PROMOLIST table also has a USED column, which contains the default value N

and is updated to Y when the shopper uses the promotion. Test the procedure with the cutoff

date 15-FEB-12. Assign free shipping for the month APR and the year 2012.



**Assignment 5-8: Adding Items to a Basket**

As a shopper selects products on the Brewbean’s site, a procedure is needed to add a newly

selected item to the current shopper’s basket. Create a procedure named BASKET\_ADD\_SP that

accepts a product ID, basket ID, price, quantity, size code option (1 or 2), and form code option

(3 or 4) and uses this information to add a new item to the BB\_BASKETITEM table. The table’s

PRIMARY KEY column is generated by BB\_IDBASKETITEM\_SEQ. Run the procedure with the

following values:

• Basket ID—14

• Product ID—8

• Price—10.80

• Quantity—1

• Size code—2

• Form code—4

A screenshot of a computer

Description automatically generated

**Assignment 5-9: Creating a Logon Procedure**

The home page of the Brewbean’s Web site has an option for members to log on with their IDs

and passwords. Develop a procedure named MEMBER\_CK\_SP that accepts the ID and password

as inputs, checks whether they make up a valid logon, and returns the member name and cookie

value. The name should be returned as a single text string containing the first and last name.

The head developer wants the number of parameters minimized so that the same

parameter is used to accept the password and return the name value. Also, if the user doesn’t

enter a valid username and password, return the value INVALID in a parameter named

p\_check. Test the procedure using a valid logon first, with the username rat55 and password

kile. Then try it with an invalid logon by changing the username to rat.

A screenshot of a computer

Description automatically generated