**MEMO**

From: Richard Rennehan, 0413101, IT Data Analytics

To: Bill Cunningham

Date: November 29, 2018

SUBJECT: DBAS3018 ASSIGNMENT 4

# **Statement of Requirement**

The purpose of this memo is to share a function, procedure and trigger that manages the shipment\_line table of Clearwater Traders with one easy step. The goal of this memo is to display my knowledge of creating functions, procedures, and triggers.

The function will be able to calculate the total value of any inventory ID passed to it. The function multiplies the quantity on hand by the price and returns the result.

The procedure will be able to automatically update a desired shipment with today’s date once the shipment has been received. It is as simple as executing the function with the inventory ID and line number. The procedure then connects to the function to display the price and quantity on hand of the inventory.

The trigger will update the inventory table’s quantity on hand for any shipment that has arrived. After a successful arrival, it will call the procedure using the ID of the inventory that arrived.

To manage the inventory and shipment of the database, all that needs to be done is to execute the procedure and pass in the shipment number and inventory number as parameters.

# **Existing Materials**

The database used is the official Clearwater Traders database. The database was sourced from a textbook called *Database Systems: Design, Implementation, and Management*. It is written by Carlos Coronel and Steven Morris. It is the 12e edition.

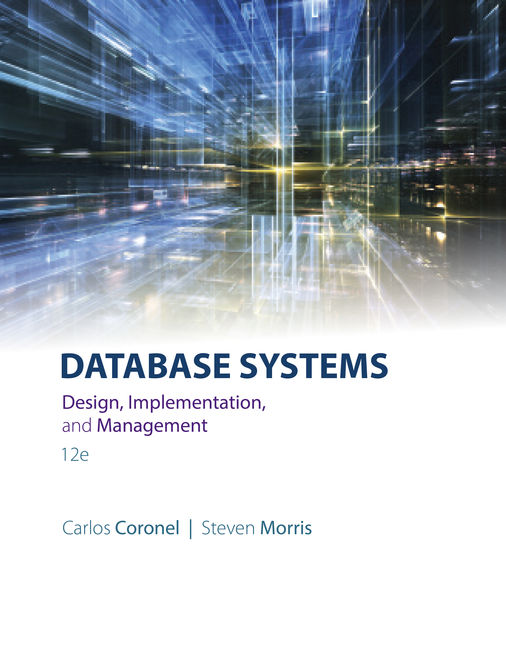


Figure 1: Textbook cover for the source of the data

# **SQL Code**

Ensure to run each in the order given below.

## **Function for Calculating Value of Item**

CREATE OR REPLACE FUNCTION FUNC\_VALUE (F\_INV\_ID IN NUMBER)

RETURN VARCHAR

IS

F\_TOTAL\_VALUE NUMBER;

BEGIN

SELECT (INV\_PRICE \* INV\_QOH) INTO F\_TOTAL\_VALUE

FROM INVENTORY

WHERE INVENTORY.INV\_ID = F\_INV\_ID;

RETURN F\_TOTAL\_VALUE;

END;

## **Procedure for Updating Shipment Arrival**

CREATE OR REPLACE PROCEDURE PRC\_SHIPMENT\_ARRIVAL

(P\_SHIP\_ID in NUMBER, P\_INV\_ID in NUMBER)

AS

P\_NEW\_QOH INVENTORY.INV\_QOH%TYPE;

P\_DATE SHIPMENT\_LINE.SL\_DATE\_RECEIVED%TYPE;

BEGIN

SELECT SL\_DATE\_RECEIVED INTO P\_DATE

FROM SHIPMENT\_LINE

WHERE SHIP\_ID = P\_SHIP\_ID AND INV\_ID = P\_INV\_ID;

IF P\_DATE IS NULL THEN

UPDATE SHIPMENT\_LINE

SET SL\_DATE\_RECEIVED = SYSDATE

WHERE SHIP\_ID = P\_SHIP\_ID AND INV\_ID = P\_INV\_ID;

SELECT INV\_QOH INTO P\_NEW\_QOH

FROM INVENTORY

WHERE INV\_ID = P\_INV\_ID;

DBMS\_OUTPUT.PUT\_LINE('Inventory has been updated. Inventory ' || P\_INV\_ID ||

' now has ' || P\_NEW\_QOH || ' units for a total value of $' ||

FUNC\_VALUE(P\_INV\_ID));

ELSE

DBMS\_OUTPUT.PUT\_LINE('The shipment has already arrived on ' || P\_DATE ||

'. Inventory QOH has not been updated');

END IF;

EXCEPTION

WHEN NO\_DATA\_FOUND THEN

DBMS\_OUTPUT.PUT\_LINE('The requested ship ID and inventory ID do not exist');

END;

## **Trigger Statement for updating Quantity on Hand (QOH)**

CREATE OR REPLACE TRIGGER TRIG\_QOH\_UPDATE

AFTER UPDATE ON SHIPMENT\_LINE

FOR EACH ROW

BEGIN

CASE

WHEN UPDATING('SL\_DATE\_RECEIVED') THEN

UPDATE INVENTORY

SET INV\_QOH = INV\_QOH + :OLD.SL\_QUANTITY

WHERE INV\_ID = :OLD.INV\_ID;

END CASE;

END;

# **Testing Plan**

| Iteration Number | Test | Expected Result | Actual Result | Solutions |
| --- | --- | --- | --- | --- |
| 1 | Test function with:  SELECT FUNC\_VALUE(2)  FROM DUAL; | The function returns the total value of items from the Inventory table with Inventory ID 2 | It displays the result of 3119.88.  259.99 multiplied by 12 is exactly that. | No problems encountered |
| 2 | Test Procedure and trigger with:  EXECUTE PRC\_SHIPMENT\_ARRIVAL (2,2); | Procedure alters the correct record with today’s date.  This automatically runs the trigger which updates the inventory QOH  Procedure prints a message stating the new number of units. Connects successfully to function | The shipment line table’s SHIP\_ID 2 with INV\_ID 2 was correctly updated to match today’s date of December 7, 2018.  The shipment had a quantity of 25. The QOH was successfully updated from 12 to 37. Thirty-seven multiplied by 259.99 gives the correct value of 9619.63  Statement processed. Inventory has been updated. Inventory 2 now has 37 units for a total value of $9619.63 | No problems encountered but will implement a quick fix preventing a shipment that has already arrived from updating QOH again. Also, will add an exception block if there is no matching inventory and ship ID |
| 3 | Test Procedure error handling with:  EXECUTE PRC\_SHIPMENT\_ARRIVAL(5,20); | The database prints a simple message stating the combination does not exist. | Statement processed. The requested ship ID and inventory ID do not exist | No problems encountered |
| 4 | Test shipment that has already arrived with:  EXECUTE PRC\_SHIPMENT\_ARRIVAL(1,1); | Procedure prints date it has arrived. Skips over the trigger and function call. | Statement processed. The shipment has already arrived on 10-SEP-06. Inventory QOH has not been updated | No problems encountered. |

# Snapshots

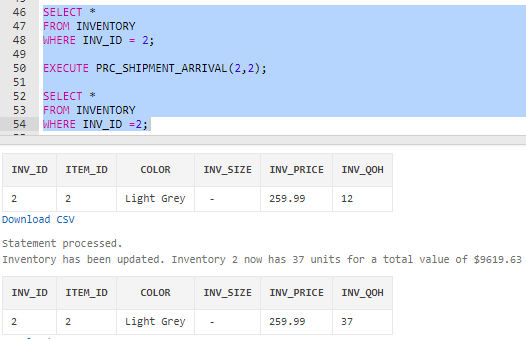


Figure : Proof that the trigger is ran successfully after the date is updated in the procedure

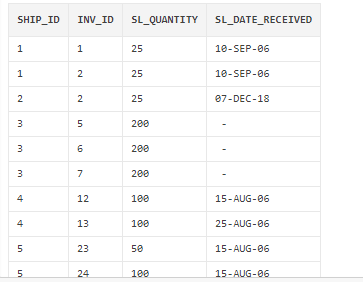


Figure : After running the procedure, ship 2 with inventory 2 gets updated to today’s date