# Database Management Systems - II

## IC 2201

### Department of ICT

**Faculty of Technology, University of Colombo**

## PL/SQL Assignment

**MultiTrade is a whole sale company which imports products and distributes in Sri Lanka.**
Following tables represent the information about the products and warehouses available with them.

**Product**

| Product_id | Product_name | Warranty_period | Supplier_code | List_price |
|---|---|---|---|---|
| PRD01 | Air cooler | 5 | SW_00101 | 25,990.00 |
| PRD02 | Ceiling fan | 2 | IN_20034 | 6,690.00 |
| PRD03 | Dry iron | 0.5 | IN_20034 | 2,750.00 |
| PRD04 | Floor polisher | 1 | NULL | 15,690.00 |
| PRD05 | Stand fan | 0.5 | SG_34023 | 18,590.00 |
| PRD06 | Steam iron | 0.5 | NULL | 2,190.00 |
| PRD07 | Vacuum cleaner | 1.5 | SG_34023 | 9,990.00 |
| PRD08 | Water heater | 2 | TW_90846 | 18,890.00 |
| PRD09 | Water purifier | 2 | US_56798 | 11,850.00 |

**Warehouse**

| Warehouse_id | Warehouse_name | Location |
|---|---|---|
| ST001 | Shop Warehouse | Colombo |
| ST002 | Large Zone | Rathmalana |
| ST003 | Retail Zone | Kiribathgoda |
| ST004 | Whole Supply | Colombo |

**Inventory**

| Product_id | Warehouse_id | Qty_on_hand |
|---|---|---|
| PRD01 | ST001 | 30 |
| PRD02 | ST001 | 45 |
| PRD02 | ST002 | 20 |
| PRD02 | ST003 | 10 |
| PRD03 | ST002 | 50 |
| PRD03 | ST004 | 50 |
| PRD06 | ST002 | 75 |
| PRD07 | ST001 | 15 |
| PRD07 | ST003 | 10 |

1) Create three tables for Product, Warehouse and Inventory as listed above. Identify the suitable data types for the fields by considering the data values provided. Set PRIMARY KEYs and FOREIGN KEYs appropriately.

2) Insert the provided values into these three tables.

   ***Note:*** *Insert the product names in the same case pattern as given.*

3) Create a PL/SQL package which includes following subprograms.
   a. **Procedure to get the warehouse name:**
      Create a private procedure which accepts Warehouse_id as a parameter and return the name of the Warehouse. The procedure should return the name in capital letters. Handle the exceptions that can occur (there should be at least two exception handlers).
   b. **Function to get discounted price of a given product**
      (When standard discount percentages are applied)
      Create a function which calculates and returns the discounted price of a given product after applying the below standard discount percentages conditionally. (Hint: Use searched CASE statement)

| List price range | Discount (%) |
|---|---|
| < 6,000 | 12 |
| >= 6,000 and < 12,000 | 16 |
| >= 12,000 | 24 |

Handle the exceptions that can occur.

- Display a message when data is not found.
- If any other exception arose, re-raise the exception.

c. **Procedure to compute total profit without discounts:**
Declare a public variable ***G_total_profit_no_discount*** at the package header. Each product has 10% of its **list price** as its profit for a single unit of sale. Create a public procedure to reflect the total profit of the current inventory in the ***G_total_profit_no_discount*** variable, if they sell **all the units** in the current inventory for their list price. (*Hint: Use a cursor*).

d. **Function overloading:**
Overload the subprogram that you created in (c) above to capture the discount % (e.g 0.1 is passed when the discount is 10%) also as a formal parameter and return the discounted price of a given product based on the captured discount percentage.

e. **Procedure to compute total profit with discounts:**
Declare two more public variables ***G_total_profit_with_discount*** and the ***G_discount_percentage*** at the package header. Each product has 10% of its **discounted price** as its profit for a single unit of sale. Create a public procedure which takes the discount % as an input (e.g. 10% should be input as 0.1) and to reflect the total profit of the current inventory in the ***G_total_profit_with_discount*** variable, if they sell **all the units** in the current inventory for the discounted price. (*Hint: Use a cursor*).

f. **Calling package methods:**
Write an anonymous block to call subprograms from the above package to do the following.
i Print the total profit of the current inventory with no discounts. ii Print the total profit of the current inventory with 10% discounts for all the products. iii Print the total profit of the current inventory with 15% discounts for all the products.
iv Print the discounted price of the product PRD01. when standard discount percentages are applied. v Print the discounted price of the product PRD01, when 20% discount percentage is applied.

**Instructions:**

- Read the given questions carefully before you start working on this assignment.
- Use proper naming conventions when creating subprograms, variables etc.
- Apply best practices as much as possible. (e.g., when creating variables that match with table fields, use %TYPE).
- Subprogram logic can be implemented in many different ways. Student should be able to identify the best method applying your in-depth knowledge gained from PL/SQL practical sessions you have learnt so far.
- Student can create customized error messages and exceptions where needed.
- When using global variables, initializing them appropriately is a must.

**Assignment Submission Guidelines**

- You must submit 2 files to complete the submission.
    I. SQL file with all the codes
    II. Word document with all the codes
- In each file, the answers for each main question and the sub parts must be clearly stated.
- Both files should be named with your student number
    a. SQL file: <Student ID>.sql
    b. Word Document: <Student ID>.docx
- Inside each file, include your **student number and name**.
    - Inside the SQL file, include them **as a comment at the top**.
    - Inside the Word document, include them **as the header**.
- When marking, the programs in the SQL file will be executed and **no editing will be done by the marker.** i.e., you must make sure that each and every code is executable without any issue. For example, there can occur some whitespace identification problems when try to copy from the SQL file and paste it in SQL Plus. If that is the case, marks will be reduced per each case.
- Your SQL file and the word document should contain the same answers. Otherwise, your answer will not be marked, and a **zero mark** will be given.
- Word Document should contain a **cover page** including,
    a. Course code
    b. Assignment
    c. Student number
    d. Student name
    e. Submission date

- The demonstration video must be recorded on your PC, showcasing the execution flow of the PL/SQL codes using SQL Plus while clearly explaining the code functionality. Ensure the video is recorded using Zoom with your camera turned on and does not exceed 10 minutes in length.
- The Answer scripts and Demo video should be uploaded to the LMS as a zip file on or before the deadline mentioned in the LMS. (Use *"PL/SQL Assignment Submission Link"*)

**IMPORTANT:**

This is an individual assignment. You are expected to complete the assignment individually.

Late assignments will be penalized 10% per day (this includes week ends as well).

Any form of notification for inability to submit an assignment including medical certificates will not be entertained. This policy is strictly enforced. Assignments should be submitted on or before the date mentioned on the assignment. If plagiarism or cheating is identified a zero mark will be given for the assignment to all students involved. This policy is strictly enforced. If you do have any problem or queries regarding the assignment, you can contact the lecturer and discuss.

**********END OF THE ASSIGNMENT**********