Islamic University of Technology

Organisation of Islamic Cooperation (OIC)

Department of Computer Science and Engineering (CSE)

MID TERM EXAMINATION **SUMMER SEMESTER, 2019-2020**

Duration: 1 Houre 30 Minutes **Full Marks: 50**

CSE 4409: Database Management Systems II

Programmable calculators are not allowed. Do not write anything on the question paper. There are **03 (three)** questions. Answer **all** of them. Figures in the right margin indicate marks. This is an **Open Book Exam**.

---

1. (a) i. Mention one major benefit of using %TYPE operator in PL/SQL programming. [2]

   ii. State one example where the use of SELECT...FOR UPDATE is justified. [2]

   (b) Consider a typical Result Processing System (RPS) where there are the following entities *(Note: pk stands for primary key while fk[x] indicates foreign key referencing entity x)*:

   - `Students(SID (pk),Name)`
   - `Courses(CID (pk), Titile, Credit)`
   - `Grades (SID fk[Students], CID fk[Courses],LetterGrade)`

   The formula for calculating the CGPA is as follows:

   $$CGPA = \frac{\sum (C_i \times N_i)}{\sum C_i}$$

   where C_i and N_i are the credit of the course and numeric value of the letter grade respectively.

   In order to enhance the flexibility of the system, now it is required that binding between numeric value and its corresponding letter grade should not be fixed (i.e. hard coded) rather it must be dynamically configured by the admin user of the system.

   Your tasks are as follows:

   i. Make the required changes (i.e. addition or modification) in the given entity design. [4]

   ii. Write a PL/SQL function as follows: [6]

Input :	Student ID
Output :	CGPA
Algorithm:	As given by the formula

Note: Question 2 and 3 are both based on the following system description:

General Structure: ABC is a private bank in Bangladesh. It has many branches. Customers normally come to a specific branch to open an account. Primarily he/she fills up a Personal Information Form (PIF) before opening an account. PIF includes Name, Date of Birth, address and contact no. A customer does not need to fill up PIF for opening another account. Normally one account is attached to one customer, but in some cases, joint accounts are also possible. In this case, one account is attached to a number of customers.

There are a number of accounts in the bank such as *Savings*, *Current* and *Joint*. Each type of account is characterized by two specific parameters: I. InterestRateYearly (IRY) and II. InterestCalculatePeriod (ICP). ICP can take 2 values as follows:

ICP Value	Meaning	Description
1	Monthly	The interest will be calculated after each month (i.e 30 days)
2	bi-yearly	The interest will be calculated after each 6 months (i.e. 180 days)

It is to mention that the Account must store the `Last Date of Interest Payment (LDIP)` and `Current Balance (CB)` information.

Business Logic Part 1: Transaction. The transactions are either deposit or withdraw. It should have a Transaction ID (TID), Account No, Amount, Type (deposit or withdraw) and DateTime. Format of the TID is *TYYMMDD.NNNN*, where T is the type of the account (i.e. 1 for Savings, 2 for Current and 3 for Joint), YYMMDD is the time of transaction and NNNN is an incremental number.

Business Logic Part 2: End of the Day. Everyday after the normal working hour, each Bank Account is checked if the interest is due today according based on the following values:

- ICP (Interest Calculation Period)
- LDIP (Last Date of Interest Payment)
- CB (Current Balance)

For each account the difference between its LDIP and sysdate is calculated. If this difference is equal to its ICP value then the net payable interest is calculated on the CB amount for LDIP period at the rate of IRY and a transaction (type: deposit) is made accordingly. After the transaction, the LDIP is updated to the current date.

Otherwise that account is skipped (i.e. no transaction is made and LDIP is not updated).

Business Logic Part 3: Database User Login Pattern. The banking software is a multi-user system. To ensure security of the system the users are strictly monitored. To accomplish this the following normal user profile is considered:

On Friday no user is supposed to login. If any user logs in on Friday then it is called a *suspicious login*. users normally login between 10.00 and 18.00 hours. Beyond this time, any login will be also considered as *suspicious login*.

2. (a) Make the DDL statements to satisfy the given requirements. [8]

   (b) Write subprograms (i.e function and trigger) to automatically generate the TID as described. [8]

3. (a) Present a solution for the End of the Day Process as described (i.e. Procedure using explicit cursor. Other functionalities should also be coded in modular forms). [12]

   (b) How can you ensure the Current Balance (CB) is always updated automatically? Explain with required implementation. [3]

   (c) Your task is to find out the suspicious login in the system. Present a suitable solution in this regard.[Hint: Database-level (for all users) trigger can monitor login event]. The solution will not stop the user from such suspicious login, but it will flag a signal to the admin. [5]