**Lê Minh Quí\_2100003619**

**Chương 7**

**Câu 1: Giả sử ta có 3 Table Sinh Viên, Kết Quả và Môn Học**

SINHVIEN(MSSV, HOTEN, NGAYSINH, PHAI, LOP)

KETQUA(MSSV, MSMH, DIEM)

MONHOC(MSMH, TENMH, SOTC)

**Yêu cầu:**

a. Sử dụng Record và Khối vô danh, xuất thông tin SV như sau:

MSSV, HOTEN, TENMH, DIEM

b. Sử dụng Cursor và khối vô danh, tính điểm trung bình của mỗi SV. Thông tin

hiển thị gồm: MSSV, HOTEN, Điểm Trung Bình(ĐTB)

c. Sử dụng Cursor và khối vô danh để xếp loại sinh viên qua tiêu chuẩn sau:

9<=ĐTB<=10 thì xếp loại ‘Giỏi’

7<=ĐTB<9 thì xếp loại ‘Khá’

5<=ĐTB<7 thì xếp loại ‘Trung bình’

Còn lại thì xếp loại ‘Yếu’

Thêm vào bảng Sinh viên thuộc tính: Xếp loại (XepLoai)

Trong đó: ĐTB = Điểm trung bình tất các môn học của sinh viên

Bài làm:

1. **Sử dụng Record và Khối vô danh**

DECLARE

v\_MSSV SINHVIEN.MSSV%TYPE;

v\_HOTEN SINHVIEN.HOTEN%TYPE;

v\_TENMH MONHOC.TENMH%TYPE;

v\_DIEM KETQUA.DIEM%TYPE;

BEGIN

FOR rec IN (SELECT sv.MSSV, sv.HOTEN, mh.TENMH, kq.DIEM

FROM SINHVIEN sv

JOIN KETQUA kq ON sv.MSSV = kq.MSSV

JOIN MONHOC mh ON kq.MSMH = mh.MSMH)

LOOP

v\_MSSV := rec.MSSV;

v\_HOTEN := rec.HOTEN;

v\_TENMH := rec.TENMH;

v\_DIEM := rec.DIEM;

DBMS\_OUTPUT.PUT\_LINE(v\_MSSV || ', ' || v\_HOTEN || ', ' || v\_TENMH || ', ' || v\_DIEM);

END LOOP;

END;

1. **Sử dụng Cursor và Khối vô danh**

DECLARE

v\_MSSV SINHVIEN.MSSV%TYPE;

v\_HOTEN SINHVIEN.HOTEN%TYPE;

v\_TotalMark NUMBER := 0;

v\_TotalCredits NUMBER := 0;

v\_Average NUMBER;

BEGIN

FOR rec IN (SELECT DISTINCT sv.MSSV, sv.HOTEN

FROM SINHVIEN sv

JOIN KETQUA kq ON sv.MSSV = kq.MSSV)

LOOP

v\_MSSV := rec.MSSV;

v\_HOTEN := rec.HOTEN;

v\_TotalMark := 0;

v\_TotalCredits := 0;

FOR rec2 IN (SELECT kq.DIEM, mh.SOTC

FROM KETQUA kq

JOIN MONHOC mh ON kq.MSMH = mh.MSMH

WHERE kq.MSSV = v\_MSSV)

LOOP

v\_TotalMark := v\_TotalMark + (rec2.DIEM \* rec2.SOTC);

v\_TotalCredits := v\_TotalCredits + rec2.SOTC;

END LOOP;

v\_Average := v\_TotalMark / v\_TotalCredits;

DBMS\_OUTPUT.PUT\_LINE(v\_MSSV || ', ' || v\_HOTEN || ', ' || v\_Average);

END LOOP;

END;

1. **Sử dụng Cursor và Khối vô danh**

DECLARE

v\_MSSV SINHVIEN.MSSV%TYPE;

v\_Average NUMBER;

v\_XepLoai VARCHAR2(20);

BEGIN

FOR rec IN (SELECT DISTINCT sv.MSSV

FROM SINHVIEN sv

JOIN KETQUA kq ON sv.MSSV = kq.MSSV)

LOOP

v\_MSSV := rec.MSSV;

SELECT AVG(kq.DIEM \* mh.SOTC)

INTO v\_Average

FROM KETQUA kq

JOIN MONHOC mh ON kq.MSMH = mh.MSMH

WHERE kq.MSSV = v\_MSSV;

IF v\_Average BETWEEN 9 AND 10 THEN

v\_XepLoai := 'Giỏi';

ELSIF v\_Average BETWEEN 7 AND 8 THEN

v\_XepLoai := 'Khá';

ELSIF v\_Average BETWEEN 5 AND 6 THEN

v\_XepLoai := 'Trung bình';

ELSE

v\_XepLoai := 'Yếu';

END IF

UPDATE SINHVIEN

SET XepLoai = v\_XepLoai

WHERE MSSV = v\_MSSV;

END LOOP;

COMMIT;

END;

**Câu 2: Tìm số ngày trong tháng của năm (Với tháng và năm là 2 số nguyên cho**

**trước)**

**Bài làm:**

DECLARE @month INT = 2; -- Tháng

DECLARE @year INT = 2024; -- Năm

SELECT DAY(

EOMONTH(CAST(@year AS VARCHAR) + '-' + CAST(@month AS VARCHAR) + '-01')) AS DaysInMonth;

**Câu 3: (Lưu ý: SV có thể tạo mảng trong hoặc ngoài PL/SQL Block)**

a. Viết hàm tạo mảng 1 chiều A nguyên ngẫu nhiên với N các giá trị không

tăng và không giảm(Các giá trị ngẫu nhiên có giá trị <N, N nguyên

dương cho trước)

**Bài làm:**

CREATE FUNCTION GenerateArray(@N INT)

RETURNS TABLE

AS

RETURN (

WITH Numbers AS (

SELECT 1 AS Number

UNION ALL

SELECT Number + 1

FROM Numbers

WHERE Number < @N

)

SELECT Number AS Value

FROM Numbers

);

b. Xuất các giá trị của mảng A.

**Bài làm:**

SELECT Value FROM dbo.GenerateArray(N);

c. Hãy xắp xếp mảng A tăng dần.

**Bài làm:**

SELECT Value FROM dbo.GenerateArray(N) ORDER BY Value;

d. Đếm các số dương lẻ có trong mảng.

**Bài làm:**

SELECT COUNT(Value) AS OddPositiveCount

FROM dbo.GenerateArray(N)

WHERE Value > 0 AND Value % 2 = 1;

e. Viết hàm đảo ngược mảng A.

**Bài làm:**

CREATE FUNCTION ReverseArray(@N INT)

RETURNS TABLE

AS

RETURN (

SELECT Value FROM dbo.GenerateArray(@N) ORDER BY Value DESC;

);

f. Viết hàm tách mảng A thành 2 mảng B và C, mảng B chứa các số

chẵn còn mảng C chứa các số còn lại.

**Bài làm:**

CREATE FUNCTION SplitArray(@N INT)

RETURNS @Result TABLE (

EvenValue INT,

OddValue INT

)

AS

BEGIN

INSERT INTO @Result (EvenValue, OddValue)

SELECT Value, NULL

FROM dbo.GenerateArray(@N)

WHERE Value % 2 = 0;

INSERT INTO @Result (EvenValue, OddValue)

SELECT NULL, Value

FROM dbo.GenerateArray(@N)

WHERE Value % 2 = 1;

RETURN;

END;