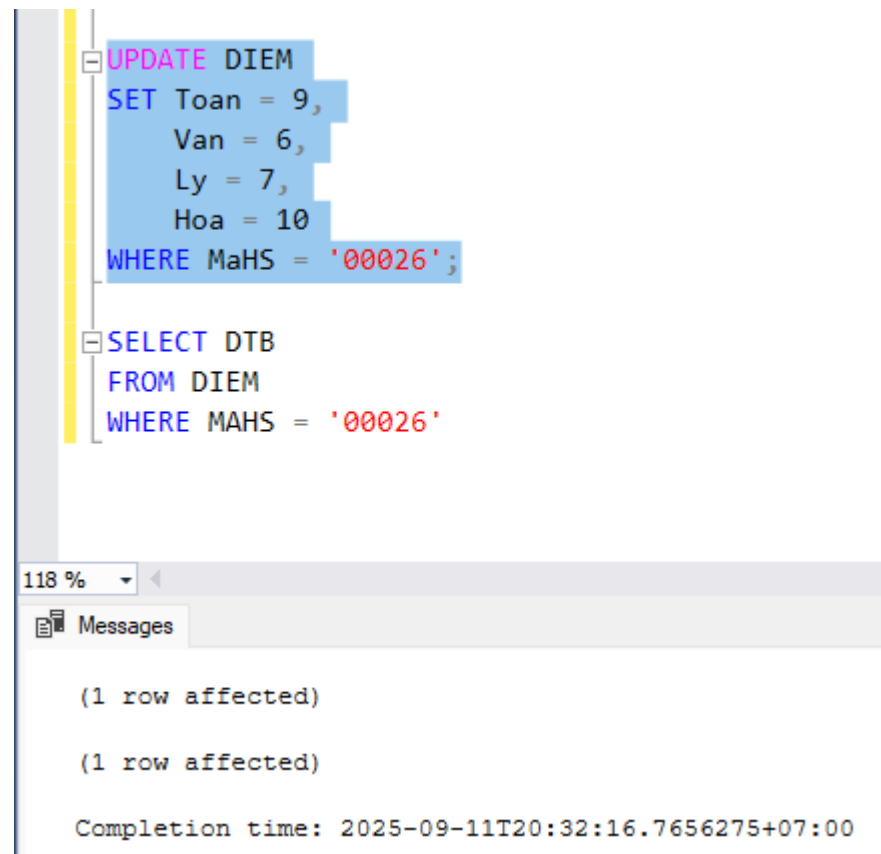


Bài tập 1: Cho cơ sở dữ liệu về quản lý học sinh (file QLHocSinh.sql):

1. Viết một Trigger gắn với bảng DIEM dựa trên sự kiện Insert, Update để tự động cập nhật điểm trung bình của học sinh khi thêm mới hay cập nhật bảng điểm
Điểm trung bình = $((\text{Toán} + \text{Văn}) * 2 + \text{Lý} + \text{Hóa}) / 6$

```
CREATE TRIGGER trg_1
ON DIEM
AFTER INSERT, UPDATE
AS
BEGIN
    UPDATE D
    SET DTB = ((D.TOAN + D.VAN) * 2 + D.LY + D.HOA) / 6
    FROM DIEM D
    INNER JOIN inserted I ON D.MAHS = I.MAHS
END
```

Kết quả:



```
UPDATE DIEM
SET Toan = 9,
    Van = 6,
    Ly = 7,
    Hoa = 10
WHERE MaHS = '00026';

SELECT DTB
FROM DIEM
WHERE MAHS = '00026'
```

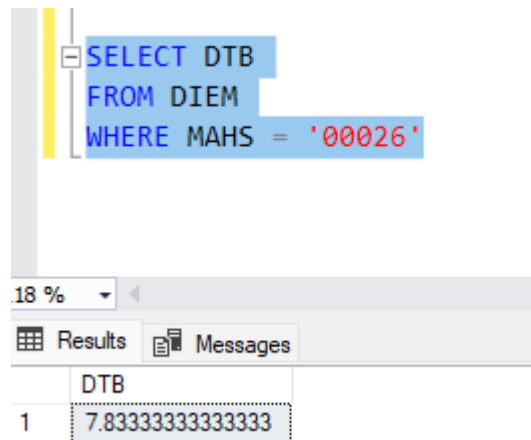
118 %

Messages

(1 row affected)

(1 row affected)

Completion time: 2025-09-11T20:32:16.7656275+07:00



The screenshot shows a SQL query window with the following text:

```
SELECT DTB
FROM DIEM
WHERE MAHS = '00026'
```

Below the query window, the 'Results' tab is active, displaying a single row of data:

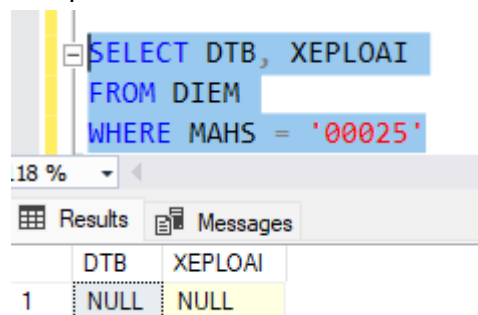
	DTB
1	7.83333333333333

2. Viết một Trigger gắn với bảng DIEM dựa trên sự kiện Insert, Update để tự động xếp loại học sinh, cách thức xếp loại như sau
 - Nếu Điểm trung bình ≥ 5 là lên lớp, ngược lại là lưu ban

```
CREATE TRIGGER trg_2
ON DIEM
AFTER INSERT, UPDATE
AS
BEGIN
    IF TRIGGER_NESTLEVEL() > 1 RETURN;

    UPDATE D
    SET XEPLOAI =
    CASE
        WHEN D.DTB >= 5 THEN N'Lên lớp'
        ELSE N'Lưu ban'
    END
    FROM DIEM D
    INNER JOIN inserted I ON D.MAHS = I.MAHS
END
```

Kết quả:



The screenshot shows a SQL query window with the following text:

```
SELECT DTB, XEPLOAI
FROM DIEM
WHERE MAHS = '00025'
```

Below the query window, the 'Results' tab is active, displaying a single row of data:

	DTB	XEPLOAI
1	NULL	NULL

```
UPDATE DIEM
SET Toan = 3,
    Van = 6,
    Ly = 7,
    Hoa = 4
WHERE MaHS = '00025';
```

8 %

Messages

(1 row affected)

(1 row affected)

(1 row affected)

(1 row affected)

Completion time: 2025-09-11T20:46:15.2046696+07:00

```
SELECT DTB, XEPLOAI
FROM DIEM
WHERE MAHS = '00025'
```

18 %

Results Messages

	DTB	XEPLOAI
1	4.83333333333333	Lưu ban

3. Viết một Trigger gắn với bảng DIEM dựa trên sự kiện Insert, Update để tự động xếp loại học sinh, cách thức xếp loại như sau

- Xét điểm thấp nhất (DTN) của các 4 môn
- Nếu DTB ≥ 5 và DTN ≥ 4 là “Lên Lớp”, ngược lại là lưu ban

```
CREATE TRIGGER trg_3
ON DIEM
AFTER INSERT, UPDATE
AS
BEGIN
    IF TRIGGER_NESTLEVEL() > 1 RETURN

    UPDATE D
    SET XEPLOAI =
    CASE
        WHEN D.DTB  $\geq 5$  AND LEAST(D.TOAN, D.VAN, D.LY, D.HOA)  $\geq 4$ 
        THEN N'Lên lớp'
        ELSE N'Lưu ban'
```

```

END
FROM DIEM D
INNER JOIN inserted I ON D.MAHS = I.MAHS
END

```

Kết quả:

SELECT DTB, XEPLOAI
FROM DIEM
WHERE MAHS = '00026'

18 %

Results Messages

	DTB	XEPLOAI
1	7.83333333333333	NULL

```

UPDATE DIEM
SET Toan = 10,
    Van = 10,
    Ly = 10,
    Hoa = 3
WHERE MaHS = '00026';

SELECT DTB, XEPLOAI
FROM DIEM
WHERE MAHS = '00026'

```

3 %

Messages

(1 row affected)

(1 row affected)

(1 row affected)

(1 row affected)

(1 row affected)

SELECT DTB, XEPLOAI FROM DIEM WHERE MAHS = '00026'		
18 %		
Results Messages		
	DTB	XEPLOAI
1	8.83333333333333	Lưu ban

4. Viết một trigger xóa tự động bản ghi về điểm học sinh khi xóa dữ liệu học sinh đó trong DSHS

```
CREATE TRIGGER trg_4
ON DSHS
AFTER DELETE
AS
BEGIN
    DELETE D
    FROM DIEM D
    INNER JOIN deleted de ON D.MAHS = de.MAHS
END
```

Kết quả

SELECT * FROM DIEM WHERE MaHS = '00026';							
118 %							
Results Messages							
	MAHS	TOAN	LY	HOA	VAN	DTB	XEPLOAI
1	00026	10	10	3	10	8.83333333333333	Lưu ban

```
DELETE FROM DSHS WHERE MaHS = '00026';

SELECT * FROM DIEM WHERE MaHS = '00026';
```

8 %

Messages

(1 row affected)

(1 row affected)

Completion time: 2025-09-11T21:04:01.2210656+07:00

```
SELECT * FROM DIEM WHERE MaHS = '00026';
```

118 %

Results Messages

MAHS	TOAN	LY	HOA	VAN	DTB	XEPLAI
------	------	----	-----	-----	-----	--------

Bài tập 2: Cho file cơ sở dữ liệu QLKhachSan.sql dùng cho quản lý khách sạn như sau:

2. Tạo Trigger tính tiền và điền tự động vào bảng tDoanhThu như sau:

Các trường lấy thông tin từ các bảng và các thông tin sau:

Trong đó:

(a) Số Ngày Ở= Ngày Ra – Ngày Vào

(b) ThucThu: Tính theo yêu cầu sau:

Nếu Số Ngày ở <10 Thành tiền = Đơn Giá * Số ngày ở

Nếu 10 <=Số Ngày ở <30 Thành Tiền = Đơn Giá* Số Ngày ở * 0.95 (Giảm5%)

Nếu Số ngày ở >= 30 Thành Tiền = Đơn Giá* Số Ngày ở * 0.9 (Giảm10%)

```
CREATE TRIGGER trg_1
ON tDoanhThu
AFTER INSERT, UPDATE
AS
BEGIN
    SET NOCOUNT ON;

    UPDATE dt
    SET
        SoNgayO = DATEDIFF(DAY, dk.NgayVao, dk.NgayRa),
        ThucThu = CASE
            WHEN DATEDIFF(DAY, dk.NgayVao, dk.NgayRa) < 10
            THEN lp.DonGia * DATEDIFF(DAY, dk.NgayVao, dk.NgayRa)
            WHEN DATEDIFF(DAY, dk.NgayVao, dk.NgayRa) < 30
            THEN lp.DonGia * DATEDIFF(DAY, dk.NgayVao, dk.NgayRa) * 0.95
            ELSE
```

```

        lp.DonGia * DATEDIFF(DAY, dk.NgayVao, dk.NgayRa) * 0.9
    END
    FROM tDoanhThu dt
    INNER JOIN inserted i ON dt.MaDK = i.MaDK
    INNER JOIN tDangKy dk ON dk.MaDK = i.MaDK
    INNER JOIN tLoaiPhong lp ON lp.LoaiPhong = dk.LoaiPhong;
END;

```

Kiểm tra
 INSERT INTO tDoanhThu (MaDK, LoaiPhong, SoNgayO, ThucThu)
 VALUES
 ('001', 'A', NULL, NULL)

```

SELECT
    dt.MaDK,
    dt.LoaiPhong,
    dk.NgayVao,
    dk.NgayRa,
    dt.SoNgayO,
    dt.ThucThu
FROM tDoanhThu dt
JOIN tDangKy dk ON dt.MaDK = dk.MaDK;

```

Kết quả:

	MaDK	LoaiPhong	NgayVao	NgayRa	SoNgayO	ThucThu
1	001	A	1998-04-26 00:00:00.000	1998-04-28 00:00:00.000	2	400000
2	012	A	2019-04-03 00:00:00.000	2019-04-05 00:00:00.000	2	400000
3	013	A	2019-04-03 00:00:00.000	2019-04-05 00:00:00.000	2	-400000
4	014	A	2019-04-03 00:00:00.000	2019-04-03 00:01:00.000	1	200000
5	015	A	2019-04-03 00:00:00.000	2019-04-04 00:00:00.000	1	200000

3. Thêm trường DonGia vào bảng tDangKy, tạo trigger cập nhật tự động cho trường này.

```

ALTER TABLE tDangKy
ADD DonGia INT NULL

CREATE TRIGGER trg_2
ON tDangKy
AFTER INSERT, UPDATE
AS
BEGIN
    SET NOCOUNT ON;

```

```

    UPDATE dk
    SET dk.DonGia = lp.DonGia
    FROM tDangKy dk
    INNER JOIN inserted i ON dk.MaDK = i.MaDK
    INNER JOIN tLoaiPhong lp ON lp.LoaiPhong = dk.LoaiPhong;

```

END

Kiểm tra

	MaDK	SoPhong	LoaiPhong	NgayVao	NgayRa	DonGia
▶	001	201	A	1998-04-26 00:00:00	1998-04-28 00:00:00	NULL

```
UPDATE tDangKy  
SET LoaiPhong = 'B'  
WHERE MaDK = '001'
```

```
SELECT MaDK, LoaiPhong, DonGia FROM tDangKy;
```

	MaDK	LoaiPhong	DonGia
1	001	B	150000
2	002	B	NULL
3	003	A	NULL
4	004	A	NULL
5	005	B	NULL
6	006	C	NULL
7	007	C	NULL
8	008	B	NULL

4. Thêm trường tổng tiêu dùng (TongTieuDung) và bảng khách hàng và tính tự động tổng tiền khách hàng đã trả cho khách sạn mỗi khi thêm, sửa, xóa bản tDangKy

```
ALTER TABLE tChiTietKH  
ADD TongTieuDung INT NULL
```

```
CREATE TRIGGER trg_3  
ON tDangKY  
AFTER INSERT, UPDATE, DELETE  
AS  
BEGIN  
    SET NOCOUNT ON;  
  
    ;WITH ChangedKH AS (  
        SELECT MaDK FROM inserted  
        UNION  
        SELECT MaDK FROM deleted  
    )  
    UPDATE kh  
    SET kh.TongTieuDung = ISNULL(  
        (  
            SELECT SUM(ISNULL(dt.ThucThu, 0))  
            FROM tDoanhThu dt  
            WHERE dt.MaDK = kh.MaDK  
        ), 0  
    )  
END
```



```
FROM tChiTietKH kh
INNER JOIN ChangedKH c ON kh.MaDK = c.MaDK;
END
```