

아래 내용은 mssql에 트리거 생성하는 소스인데 문법에 맞는지, 오류가 있는지 확인해줘.

그리고 MSSQL 버전으로 주석을 추가하여 재작성해줘
작성할 때 다음과 같은 사항을 반영해줘.

- 친구처럼 편하게 말하기.
- 모든 설명은 반드시 한글을 우선시 하기
- 복잡한 요청은 단계별로 간략하게 정리해서 질문하기
- 모르면 모른다고 말하기
- 모든 답변은 가독성있게 정리하기

DELIMITER

DELIMITER ;

좋아~ 너가 주신 트리거 코드는 딱 봐도 MySQL 문법이야.

그런데 ****MSSQL(SQL Server)****에서는 문법이 꽤 달라서 바로 쓸 수는 없어 ☹

트리거 문법, 동적 SQL 처리 방식, 변수 선언 등이 모두 다르거든.

✔ 정리해서 알려줄게!

▣ 1. 문제점 요약

너가 작성한 트리거는 MySQL 문법이라서 MSSQL에선 아래 요소들이 문제야:

항목	문제 내용
DELIMITER \$\$	▣ MSSQL에는 이런 문법 없음 (MySQL에서만 사용)
CREATE TRIGGER ... BEFORE INSERT	▣ MSSQL은 BEFORE 트리거를 지원하지 않아! AFTER INSERT 또는 INSTEAD OF INSERT만 가능
DECLARE, SET, CONCAT, DATE_FORMAT, NOW() 등	▣ 전부 MySQL 전용 함수와 문법
PREPARE, EXECUTE ... USING	▣ 동적 SQL 처리 방식이 MSSQL에서는 sp_executesql 방식으로 완전 다름

결론은 **☹**SQL Server**에서는 이 코드를 통째로 사용할 수 없어☹**.

▣ 단계별로 바꿔보자! (MSSQL용으로 변환)

▣ 기본 구조 재작성

1. MSSQL에서는 **INSTEAD OF INSERT** 트리거를 이용해서 INSERT 자체를 가로채야 해.
2. `sp_executesql`을 사용해서 동적 SQL로 중복 검사를 실행한다.
3. 테이블 존재여부는 `INFORMATION_SCHEMA.TABLES` 또는 `sys.objects` 사용.
4. 중복이면 새로 들어오는 값의 status를 3으로 세팅하고,
중복이 아니면 기존 값으로 유지되도록 처리.

✓ MSSQL 버전 트리거 예시 (주석 포함)

```
-- ▣ 트리거: MMS_MSG 테이블에 중복 메시지를 방지하기 위한 트리거
-- ▣ 중복 판단: 현재월, 이전월 테이블에 동일한 phone + msg + status=3 가 있는지 검사
-- ✓ SQL Server에서 트리거는 BEFORE INSERT가 없어서 INSTEAD OF INSERT로 구현함

CREATE TRIGGER trg_handle_duplicate_mms
ON MMS_MSG
INSTEAD OF INSERT
AS
BEGIN
    SET NOCOUNT ON;

    -- 테이블 존재 여부 확인용 변수
    DECLARE @currentTableName NVARCHAR(100);
    DECLARE @prevTableName NVARCHAR(100);
    DECLARE @sql NVARCHAR(MAX);
    DECLARE @found INT;

    -- 현재 월 및 이전 월 계산
    SET @currentTableName = 'MMS_LOG_' + FORMAT(GETDATE(), 'yyyyMM');
    SET @prevTableName = 'MMS_LOG_' + FORMAT DATEADD(MONTH, -1, GETDATE()), 'yyyyMM');

    -- 결과 저장할 테이블 만들기 (중간 처리용 테이블 변수)
    DECLARE @InsertRows TABLE (
        id INT,
        phone NVARCHAR(50),
        msg NVARCHAR(MAX),
        status INT
    );

    -- 트리거는 다중 행 입력도 처리해야 함
    DECLARE cur CURSOR FOR
        SELECT id, phone, msg, status FROM inserted;

    DECLARE @id INT, @phone NVARCHAR(50), @msg NVARCHAR(MAX), @status INT;

    OPEN cur;
    FETCH NEXT FROM cur INTO @id, @phone, @msg, @status;

    WHILE @@FETCH_STATUS = 0
    BEGIN
```

```

SET @found = 0;

-- 1. 현재 월 테이블 검사
IF EXISTS (
    SELECT 1
    FROM INFORMATION_SCHEMA.TABLES
    WHERE TABLE_NAME = @currentTableName
)
BEGIN
    SET @sql = '
        SELECT @out_found = COUNT(*)
        FROM ' + QUOTENAME(@currentTableName) + '
        WHERE phone = @phone AND msg = @msg AND status = 3';

    EXEC sp_executesql @sql,
        N'@phone NVARCHAR(50), @msg NVARCHAR(MAX), @out_found INT OUTPUT',
        @phone = @phone, @msg = @msg, @out_found = @found OUTPUT;
END

-- 2. 현재 월에서 못 찾았으면 이전 월 테이블 검사
IF @found = 0 AND EXISTS (
    SELECT 1
    FROM INFORMATION_SCHEMA.TABLES
    WHERE TABLE_NAME = @prevTableName
)
BEGIN
    SET @sql = '
        SELECT @out_found = COUNT(*)
        FROM ' + QUOTENAME(@prevTableName) + '
        WHERE phone = @phone AND msg = @msg AND status = 3';

    EXEC sp_executesql @sql,
        N'@phone NVARCHAR(50), @msg NVARCHAR(MAX), @out_found INT OUTPUT',
        @phone = @phone, @msg = @msg, @out_found = @found OUTPUT;
END

-- 3. 중복이면 status = 3 으로, 아니면 원래 값으로 저장
IF @found > 0
    SET @status = 3;

-- 4. 결과 테이블에 저장
INSERT INTO @InsertRows (id, phone, msg, status)
VALUES (@id, @phone, @msg, @status);

FETCH NEXT FROM cur INTO @id, @phone, @msg, @status;
END;

CLOSE cur;
DEALLOCATE cur;

-- 5. 실제 INSERT 수행
INSERT INTO MMS_MSG (id, phone, msg, status)
SELECT id, phone, msg, status
FROM @InsertRows;
END;

```

★ 요약하자면

항목	내용
트리거 유형	INSTEAD OF INSERT 사용
MySQL → MSSQL 차이점	BEFORE INSERT, DECLARE/SET, PREPARE/EXECUTE 문법 다름
동적 SQL 처리	sp_executesql + 변수 전달 방식 사용
주의사항	다중 행 INSERT도 처리하기 위해 CURSOR 사용

궁금할 수도 있는 점

- ? phone, msg, status 컬럼 이름 바꾸면 안 돼?
▢ 바뀌도 되는데 그에 맞게 트리거에서도 같이 바꿔줘야 해!
- ? 동작이 느릴 수도 있지 않나?
▢ 월별 파티셔닝이라면 테이블 2개만 검사하니까 괜찮은 편이야.
다만 메시지 수가 엄청 많다면 인덱스 최적화는 꼭 필요해!

혹시 이해 안 되는 부분 있으면,

"이 부분 다시 설명해줘!" 라고 편하게 말해줘 ▢

지금까지 용감한 변환작업이었어 ☺