

문제 1      문제의 요구사항을 확인 후, 프로시저를 작성하세요.  
문제에서 사용할 테이블은 아래와 같습니다.

```
create table T매출 (  
    일자 nvarchar(8),  
    제품 nvarchar(30),  
    수량 int default 0,  
    primary key (일자, 제품)  
)  
go
```

```
insert into T매출(일자, 제품, 수량)  
values ('20200101', 'A1', 10), ('20200102', 'A2', 20)  
go
```

아래는 요구사항입니다.

update, delete 를 하나의 트랜잭션으로 관리합니다.  
각 DML 문장의 성공 결과는 시스템 변수를 확인합니다.  
참조할 시스템 변수는 @@ERROR, @@ROWCOUNT  
이고, 트랜잭션의 완료와 취소는 아래를 참고하세요.

@@ERROR <> 0 또는 @@ROWCOUNT <> 1 아니면  
모두 취소이고, 그외는 모두 완료가 되도록 합니다.

트랜잭션이 완료 또는 취소 후의 데이터를 출력합니다.

T매출 테이블에 대한 update 문장의 실행 조건입니다.  
조건의 일자, 제품은 각각 '20200101', 'A1' 입니다.

T매출 테이블에 대한 delete 문장의 실행 조건입니다.  
조건의 일자, 제품은 각각 '20200105', 'A5' 입니다.

문제 2      문제의 요구사항을 확인 후, 프로시저를 작성하세요.  
문제에서 사용할 임시 로컬 테이블은 아래와 같습니다.

```
CREATE TABLE #T제품 (  
    제품코드 NVARCHAR(30)  
    ,제품명 NVARCHAR(30)  
)  
  
INSERT INTO #T제품 (제품코드, 제품명)  
VALUES ('A2', '당근') , ('A1', '사과'), ('A4', '레몬'),  
        ('A3', '포도'), ('A5', '양파'), ('A9', '상추'),  
        ('A7', '감자'), ('A6', '고추'), ('A8', '버섯')
```

프로시저의 실행 결과가 아래와 같이 되도록 하세요.

	제품코드	제품명
1	A4	레몬
2	A5	양파
3	A6	고추
4	A7	감자
5	A8	버섯
6	A9	상추

	제품코드	제품명
1	A4	레몬
2	A5	양파

문제 3     문제의 요구사항을 확인 후, 프로시저를 작성하세요.  
아래의 프로시저는 커서를 사용하고 있습니다.  
아래의 코드에서 커서를 사용하지 않고 동일하게 실행이 되도록  
새로운 프로시저를 작성하세요.

```
CREATE PROCEDURE SP_CURSOR
AS
BEGIN
    CREATE TABLE #T작업1 (
        코드    NVARCHAR(10)
        ,수량    NUMERIC(18,0)
    )

    INSERT INTO #T작업1 (코드, 수량)
    VALUES ('A1', 10), ('A2', 20), ('A3', 30)

    DECLARE 커서1 CURSOR FOR
    SELECT A.코드, A.수량
    FROM #T작업1 A
    WHERE A.코드 < 'A3'
    ORDER BY 수량 DESC

    OPEN 커서1;

    DECLARE @커서1_코드    NVARCHAR(10)
        ,@커서1_수량    NUMERIC(18,0)

    WHILE (1 = 1) BEGIN
        FETCH NEXT FROM 커서1 INTO @커서1_코드, @커서1_수량

        IF @@FETCH_STATUS <> 0 BREAK

        SELECT 코드    = @커서1_코드,
            수량    = @커서1_수량

    END;

    CLOSE 커서1
    DEALLOCATE 커서1

    DROP TABLE #T작업1

END
```

문제 4      문제의 요구사항을 확인 후, 프로시저를 작성하세요.  
문제에서 사용할 임시 로컬 테이블은 아래와 같습니다.

```
CREATE TABLE #제품 (
    제품코드    NVARCHAR(20),
    제품명      NVARCHAR(20)
)
INSERT INTO #제품 (제품코드, 제품명)
VALUES ('A','사과'), ('B','포도'), ('C','딸기'),
       ('D','수박'), ('E','참외')

CREATE TABLE #매입 (
    제품코드    NVARCHAR(20),
    매입일자    NVARCHAR(20),
    매입수량    NUMERIC(18,0)
)
INSERT INTO #매입 (제품코드, 매입일자, 매입수량)
VALUES ('A', '20191201', 100), ('A', '20200103', 200),
       ('B', '20200201', 300), ('C', '20200105', 400),
       ('D', '20200107', 500)

CREATE TABLE #매출 (
    제품코드    NVARCHAR(20),
    매출일자    NVARCHAR(20),
    매출수량    NUMERIC(18,0)
)
INSERT INTO #매출 (제품코드, 매출일자, 매출수량)
VALUES ('A', '20191220', 10), ('A', '20200103', 20),
       ('B', '20200305', 30), ('B', '20200217', 40),
       ('C', '20200220', 50)
```

아래는 프로시저를 실행하여 특정 기간의 재고 수불 정보를 조회한 결과입니다. 재고수불 조회 검색 조건은 다음과 같습니다.

수불 검색 시작일 : 20200101

수불 검색 종료일 : 20200131

	제품코드	제품명	기초수량	매입수량	매출수량	기말수량
1	A	사과	90	200	20	270
2	B	포도	0	0	0	0
3	C	딸기	0	400	0	400
4	D	수박	0	500	0	500
5	E	참외	0	0	0	0

검색 조건은 프로시저 실행시 입력해야 합니다.

기초 수량은 수불 검색 시작일 이전의 매입 및 매출 데이터를 집계해서 출력하면 됩니다.

기말 수량은 기초수량, 매입수량, 매출수량이 모두 집계된 후 최종적으로 기말수량 = 기초수량 + 매입수량 - 매출수량으로 계산이 되도록 하면 됩니다.

