SQL/PL: Generating Data

```
--#SET DELIMITER !

-- ============================================================
ECHO Making table Book#1.!

CREATE TABLE Book#1 (
    book#     BIGINT                        NOT NULL PRIMARY KEY,
    language  VARCHAR(10)  DEFAULT NULL,
    genre     VARCHAR(15)  DEFAULT NULL,
    publisher VARCHAR(25)  DEFAULT NULL,
    price     DECIMAL(5,2) DEFAULT 0.00,
    lg#       BIGINT       DEFAULT 0,
    floor     FLOAT        DEFAULT 0.0,
    ceiling   FLOAT        DEFAULT 0.0
)!
```

SQL/PL: Generating Data (p.2)

```
ECHO Populating...!
BEGIN ATOMIC
DECLARE #books_ BIGINT  DEFAULT 14837;
INSERT INTO Book#1 (book#, language, genre, publisher)
WITH
    BookRow (row#, guess) AS (
        VALUES
            (BIGINT(1), rand())
        UNION ALL
        SELECT row# + 1, rand()
            FROM BookRow
            WHERE row# < #books_
    ),
    BookLGP (row#, language, genre, publisher) AS (
        SELECT B.row#, PLG.language, PLG.genre, PLG.publisher
            FROM BookRow B, PubLangGen#1 PLG
            WHERE B.guess >= PLG.floor AND B.guess < PLG.ceiling
    )
SELECT row#, language, genre, publisher
    FROM BookLGP;
END!
```

SQL/PL: Generating Data (p.3)

```
ECHO Adding a price for each...!
BEGIN ATOMIC
DECLARE price_ FLOAT  DEFAULT   0.00;
DECLARE upper_ FLOAT  DEFAULT 999.95;
DECLARE lower_ FLOAT  DEFAULT   5.00;
FOR B AS
    SELECT book#, price_mu, price_sig
        FROM Book#1 B, Genre#1 G
        WHERE B.genre = G.genre
DO
    SET price_ = price_mu + (gauss_rand() * price_sig);
    WHILE price_ > upper_ OR price_ < lower_ DO
        SET price_ = price_mu + (gauss_rand() * price_sig);
    END WHILE;

    UPDATE Book#1 A
        SET price = DECIMAL(price_, 5, 2)
        WHERE A.book# = B.book#;
END FOR;
END!
```