-- A11

-- 70. ADD VID\_STATUS TO VIDEO TABLE

ALTER TABLE VIDEO

ADD VID\_STATUS CHAR(4) CHECK (VID\_STATUS IN ('IN','OUT','LOST')) DEFAULT 'IN'

-- 71. UPDATE VID\_STATUS

UPDATE VIDEO

SET VID\_STATUS = 'OUT'

WHERE Vid\_Num IN (SELECT V.Vid\_Num

FROM DETAILRENTAL D LEFT OUTER JOIN VIDEO V ON D.VID\_NUM = V.VID\_NUM

WHERE D.DETAIL\_RETURNDATE IS NULL)

-- 72. ADD PRICE\_RENTDAYS

ALTER TABLE PRICE

ADD PRICE\_RENTDAYS INT NOT NULL

DEFAULT (3)

-- 73 UPDATE PRICE\_RENTDAYS

CREATE TABLE PRICE\_TO\_RENTDAYS(

PRICE\_CODE INT,

PRICE\_RENTDAYS INT,

);

INSERT INTO PRICE\_TO\_RENTDAYS VALUES (1,5);

INSERT INTO PRICE\_TO\_RENTDAYS VALUES (2,3);

INSERT INTO PRICE\_TO\_RENTDAYS VALUES (3,5);

INSERT INTO PRICE\_TO\_RENTDAYS VALUES (4,7);

UPDATE PRICE

SET PRICE\_RENTDAYS = PTR.PRICE\_RENTDAYS

FROM PRICE P INNER JOIN PRICE\_TO\_RENTDAYS PTR ON P.Price\_Code = PTR.PRICE\_CODE

-- 78 INSERT NEW RENTAL INTO DATABASE

CREATE PROC PRC\_NEW\_DETAIL

@VID\_NUM VARCHAR(20),

@RENT\_NUM INT

AS

BEGIN

-- DECLARING VARIABLES

DECLARE @RENT\_FEE FLOAT(8)

DECLARE @DAILY\_FEE FLOAT(8)

DECLARE @RENT\_DAYS int

DECLARE @TEMP\_DUEDATE date

-- PART C

IF @VID\_NUM IN (SELECT VID\_NUM FROM VIDEO WHERE VID\_STATUS = 'IN')

BEGIN

-- PART D

SET @RENT\_FEE = (SELECT PRICE\_RENTFEE

FROM PRICE P INNER JOIN MOVIE M ON P.Price\_Code = M.Price\_Code

INNER JOIN VIDEO V ON V.Movie\_Num = M.Movie\_Num

WHERE V.Vid\_Num = @VID\_NUM)

SET @DAILY\_FEE = (SELECT DAILY\_FEE

FROM PRICE P INNER JOIN MOVIE M ON P.Price\_Code = M.Price\_Code

INNER JOIN VIDEO V ON V.Movie\_Num = M.Movie\_Num

WHERE V.Vid\_Num = @VID\_NUM)

--PART E

select @TEMP\_DUEDATE = Getdate() + @RENT\_DAYS

-- PART F

INSERT INTO DETAILRENTAL

SELECT @RENT\_NUM, @VID\_NUM, @RENT\_FEE, @TEMP\_DUEDATE, null, @DAILY\_FEE

END

-- PART B

ELSE PRINT 'VIDEO NUMBER IS NOT IN THE DATABASE'

END

select \* from DETAILRENTAL where VID\_NUM = 12345

-- 79 STORED PROCEDURE TO RETURN VIDEO

CREATE PROC PRC\_RETURN\_VIDEO

@VID\_NUM VARCHAR(20)

AS

BEGIN

-- PART B

IF EXISTS (SELECT VID\_NUM FROM VIDEO WHERE @VID\_NUM = Vid\_Num)

BEGIN

-- PART C

IF ((SELECT COUNT(VID\_NUM) FROM DETAILRENTAL WHERE @VID\_NUM = VID\_NUM AND DETAIL\_RETURNDATE IS NULL) = 1 )

BEGIN

UPDATE VIDEO

SET VID\_STATUS = 'IN', Vid\_InDate = SYSDATETIME()

WHERE Vid\_Num = @VID\_NUM

PRINT 'VIDEO HAS BEEN RETURNED AND IS NOW AVAILABLE FOR RENTAL'

END

ELSE IF ((SELECT COUNT(VID\_NUM) FROM DETAILRENTAL WHERE @VID\_NUM = VID\_NUM) > 1)

BEGIN

PRINT 'VIDEO HAS MULTIPLE OUTSTANDING RENTALS'

END

ELSE

BEGIN

UPDATE VIDEO

SET VID\_STATUS = 'IN'

WHERE Vid\_Num = @VID\_NUM

PRINT 'VIDEO HAS NO OUTSTANDING RENTALS BUT IS NOW AVAILABLE TO RENT'

END

END

ELSE PRINT 'VIDEO WAS NOT FOUND'

END