

```

--70
ALTER TABLE VIDEO
ADD VID_STATUS VARCHAR(4) NOT NULL CHECK(VID_STATUS IN ('IN', 'OUT', 'LOST')) DEFAULT
'IN'

--71
UPDATE VIDEO
SET VID_STATUS = 'OUT'
WHERE VID_NUM IN (SELECT VID_NUM
                  FROM DETAILRENTAL
                  WHERE DETAIL_RETURNDATE IS NULL
                  )

--72
ALTER TABLE PRICE
ADD PRICE_RENTDAYS NUMERIC(2) NOT NULL DEFAULT (3)

--73
UPDATE PRICE
SET PRICE_RENTDAYS = 5
WHERE PRICE_CODE = 1 OR PRICE_CODE = 3

UPDATE PRICE
SET PRICE_RENTDAYS = 7
WHERE PRICE_CODE = 4

--78
CREATE PROCEDURE PRC_NEW_DETAIL
(@VID_NUM INT, @RENT_NUM INT)
AS
BEGIN
    IF NOT (@VID_NUM IN (SELECT VID_NUM FROM VIDEO))
        PRINT 'VIDEO DOES NOT EXIST'
    ELSE
        PRINT 'VIDEO EXISTS'

    BEGIN
        IF NOT (@VID_NUM IN (SELECT VID_NUM FROM VIDEO WHERE VID_STATUS = 'IN'))
            PRINT 'VIDEOS RETURN MUST BE ENTERED'
        ELSE
            BEGIN
                DECLARE @PRICE_RENTFEE FLOAT
                DECLARE @PRICE_DAILYLATEFEE FLOAT
                DECLARE @PRICE_RENTDAYS INT

                SELECT @PRICE_RENTFEE = PRICE_RENTFEE, @PRICE_DAILYLATEFEE =
PRICE_DAILYLATEFEE, @PRICE_RENTDAYS = PRICE_RENTDAYS
                FROM PRICE

                DECLARE @DETAIL_DUEDATE DATETIME

                SET @DETAIL_DUEDATE =
DATEADD(DAY, @PRICE_RENTDAYS, GETDATE())

                INSERT INTO DETAILRENTAL (RENT_NUM, VID_NUM, DETAIL_FEE,
DETAIL_DUEDATE, DETAIL_DAILYLATEFEE)
                VALUES (@RENT_NUM, @VID_NUM, @PRICE_RENTFEE, @DETAIL_DUEDATE,
@PRICE_DAILYLATEFEE)
            END
        END
    END

```

```

        END
    END
END

--79
CREATE PROCEDURE PRC_RETURN_VIDEO
(@VID_NUM INT)
AS
BEGIN
    IF NOT (@VID_NUM IN (SELECT VID_NUM FROM VIDEO))
        PRINT 'VIDEO NUMBER PROVIDED WAS NOT FOUND'
    ELSE
        IF((SELECT COUNT(VID_NUM) FROM    DETAILRENTAL WHERE VID_NUM = @VID_NUM) <>
1)
            PRINT 'VIDEO HAS MULTIPLE OUTSTANDING RECORDS'
        ELSE
            IF (SELECT DETAIL_RETURNDATE FROM DETAILRENTAL WHERE VID_NUM =
@VID_NUM) IS NULL
                BEGIN
                    UPDATE VIDEO
                    SET          VID_STATUS = 'IN'
                    WHERE  VID_NUM = @VID_NUM

                    PRINT  'VIDEO HAS NO OUTSTANDING RENTALS AND IS NOW AVAILABLE
FOR RENTAL'
                END
            ELSE
                BEGIN
                    UPDATE  DETAILRENTAL
                    SET      DETAIL_RETURNDATE = GETDATE()
                    WHERE  VID_NUM = @VID_NUM

                    PRINT  'VIDEO WAS SUCCESSFULLY RETURNED'
                END
            END
        END
END

```