**ASSIGNMENT 11**

A record company wishes to use a computer database to help with its operations regarding its performers, recordings and song catalogue. A requirements analysis has elicited the following information: Songs have a unique song number, a non-unique title and a composition date. A song can be written by a number of composers; the composer’s full name is required. Songs are recorded by recording artists (bands or solo performers). A song is recorded as a track of a CD. A CD has many songs on it, called tracks. CDs have a unique record catalogue number, a title and must have a producer(the full name of the producer is required). Each track must have the recording date and the track number of the CD. A song can appear on many (or no) CDs, and be recorded by many different recording artists. The same recording artist might re-record the same song on different CDs. A CD must have only 1 recording artist appearing on it. CDs can be released a number of times, and each time the release date and associated number of sales is required.

1.Use this information to design an appropriate ER and relational model.

2.Compile DDL and DML commands on the database created.

SQL:-

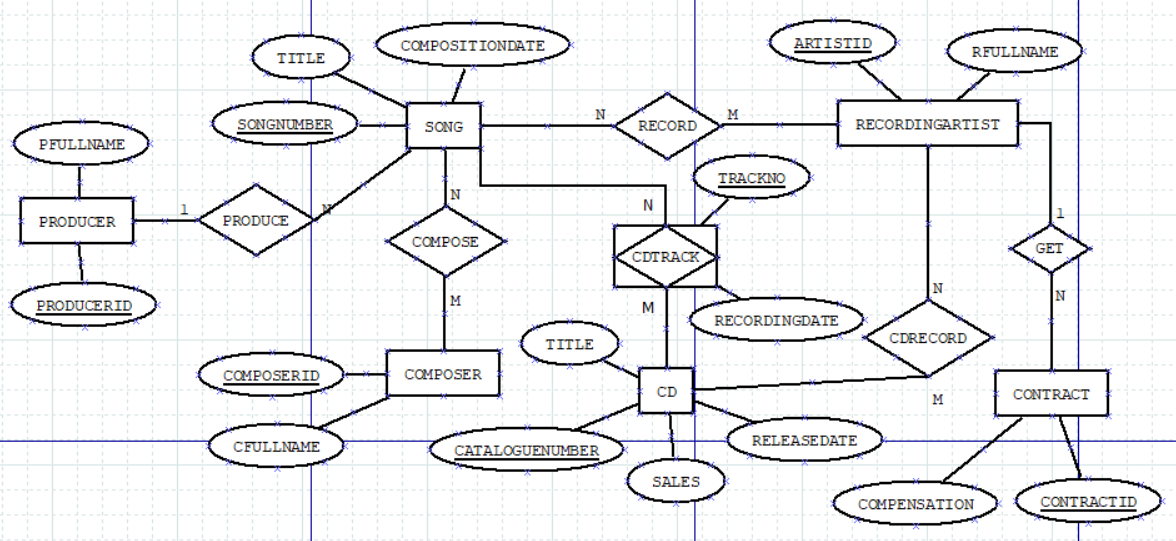
i>Update number of recorded albums to 4 for those artists who have recorded only 3.

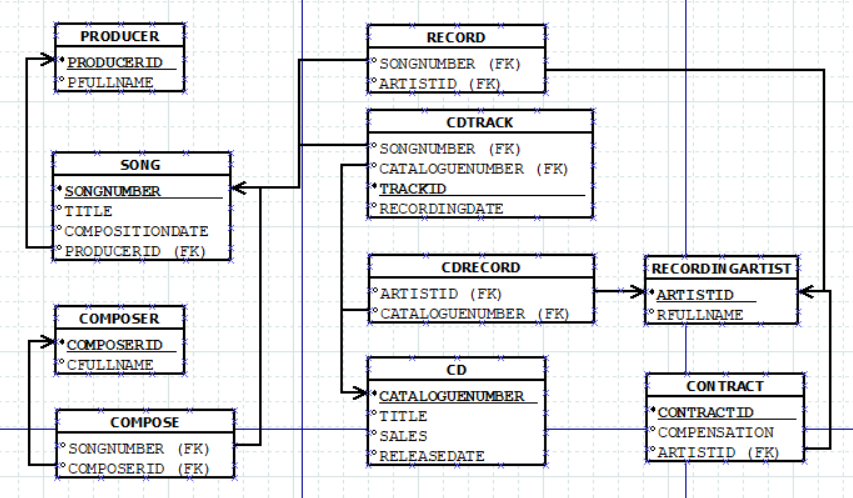
ii>Find all artists who have recorded at least two albums.

iii>Find all writers who have only written one song.

PL/SQL

i>Write Procedure to insert a new Contract into the Contract relation.





CREATE TABLE PRODUCER (

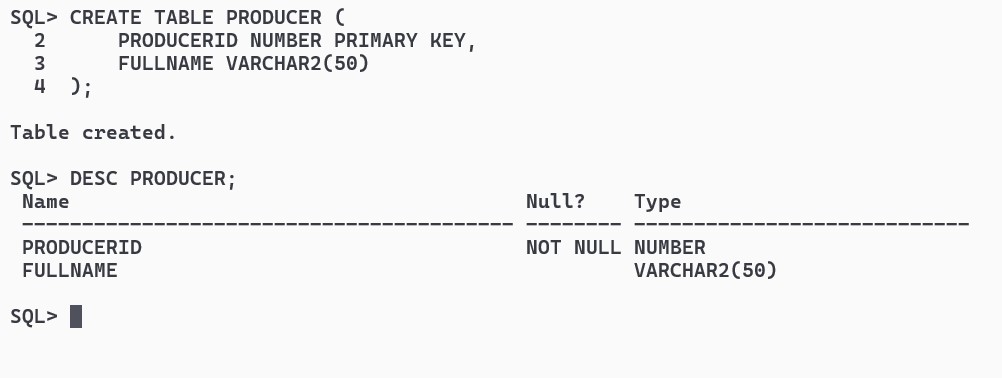
PRODUCERID NUMBER PRIMARY KEY,

FULLNAME VARCHAR2(50),

--SONGNUMBER NUMBER,

--CONSTRAINT PFK1 FOREIGN KEY (SONGNUMBER) REFERENCES SONG(SONGNUMBER) ON DELETE CASCADE

);



CREATE TABLE SONG (

SONGNUMBER NUMBER PRIMARY KEY,

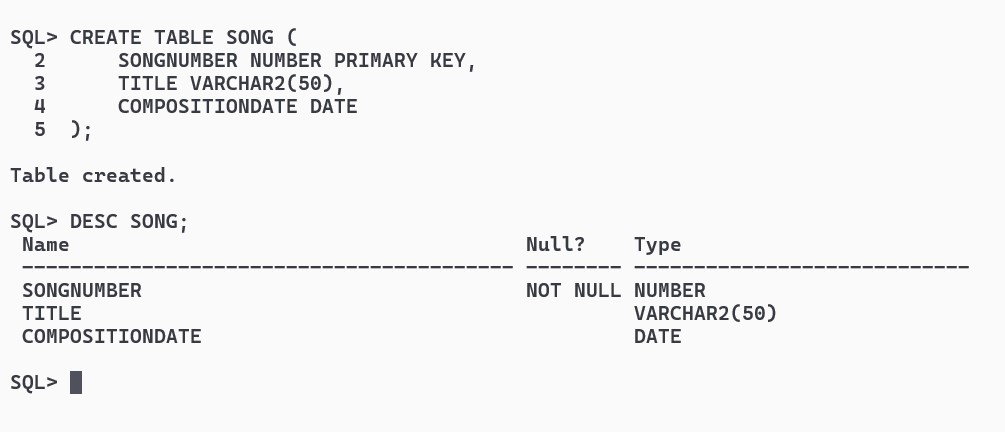
TITLE VARCHAR2(50),

COMPOSITIONDATE DATE,

PRODUCERID NUMBER,

CONSTRAINT SFK1 FOREIGN KEY (PRODUCERID) REFERENCES PRODUCER(PRODUCERID) ON DELETE CASCADE

);

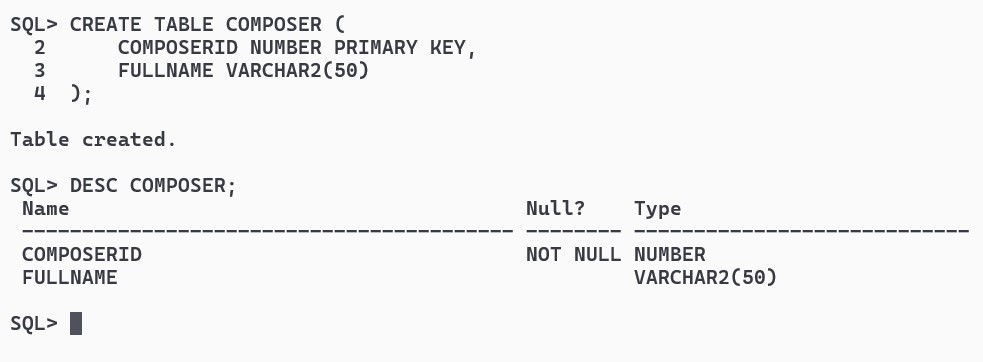


CREATE TABLE COMPOSER (

COMPOSERID NUMBER PRIMARY KEY,

FULLNAME VARCHAR2(50)

);

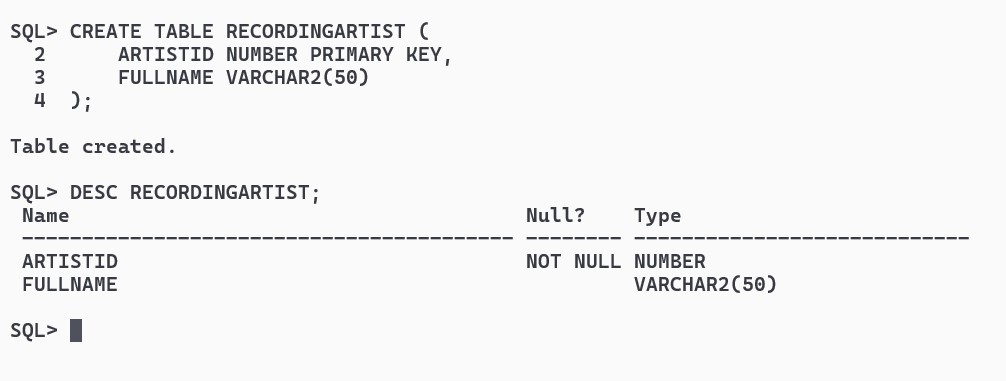


CREATE TABLE RECORDINGARTIST (

ARTISTID NUMBER PRIMARY KEY,

FULLNAME VARCHAR2(50)

);



CREATE TABLE CD (

CATALOGUENUMBER NUMBER PRIMARY KEY,

TITLE VARCHAR2(50),

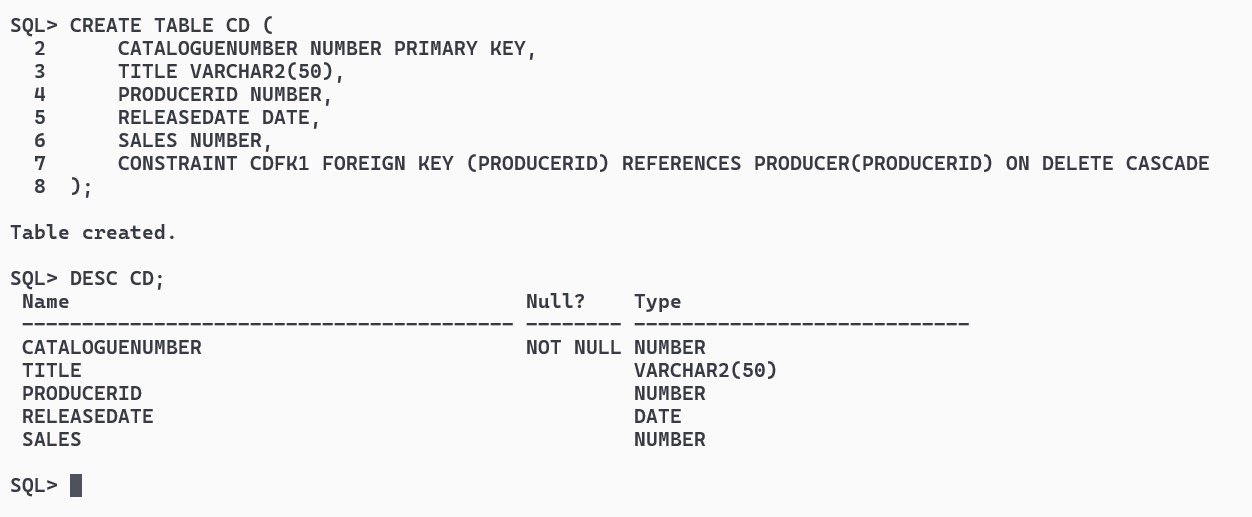
PRODUCERID NUMBER,

RELEASEDATE DATE,

SALES NUMBER,

CONSTRAINT CDFK1 FOREIGN KEY (PRODUCERID) REFERENCES PRODUCER(PRODUCERID) ON DELETE CASCADE

);



CREATE TABLE SONGCOMPOSER (

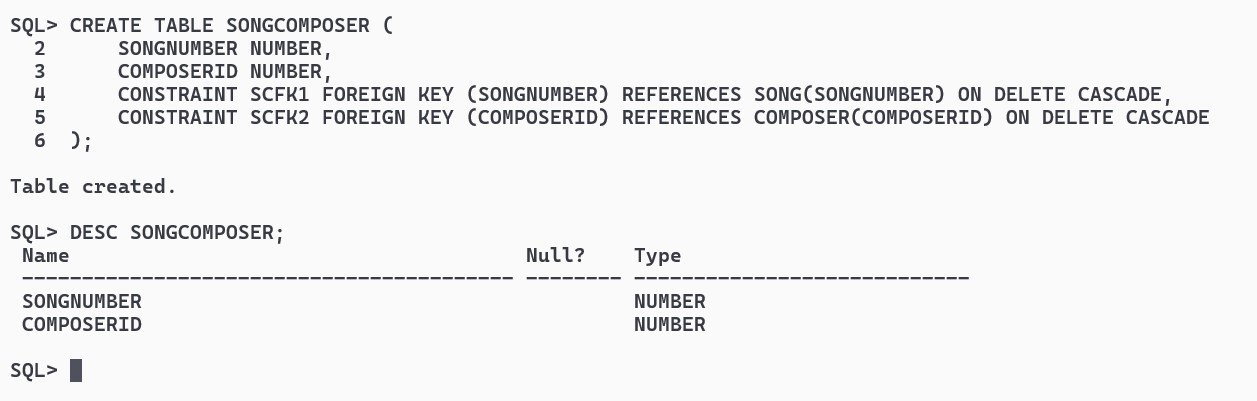
SONGNUMBER NUMBER,

COMPOSERID NUMBER,

CONSTRAINT SCFK1 FOREIGN KEY (SONGNUMBER) REFERENCES SONG(SONGNUMBER) ON DELETE CASCADE,

CONSTRAINT SCFK2 FOREIGN KEY (COMPOSERID) REFERENCES COMPOSER(COMPOSERID) ON DELETE CASCADE

);



CREATE TABLE SONGRECORDINGARTIST (

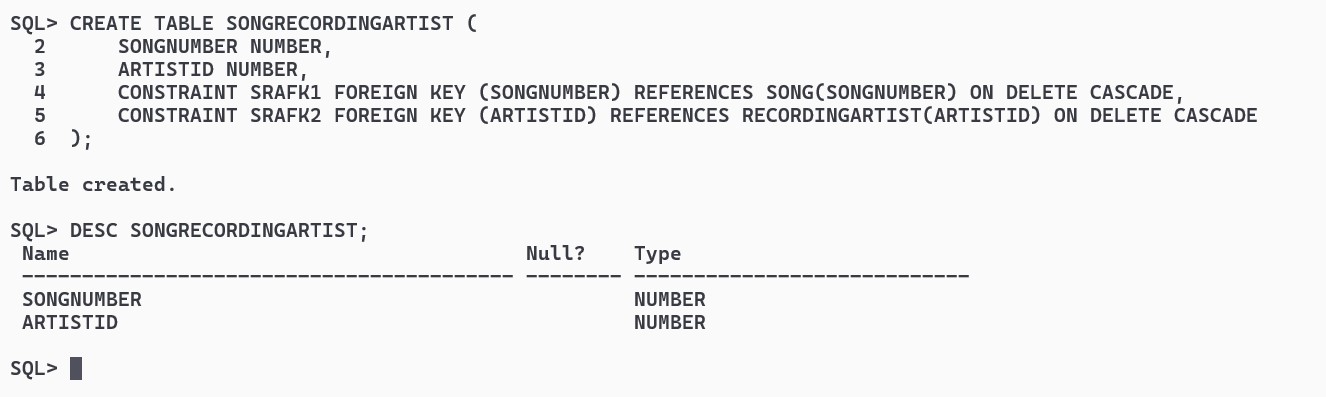
SONGNUMBER NUMBER,

ARTISTID NUMBER,

CONSTRAINT SRAFK1 FOREIGN KEY (SONGNUMBER) REFERENCES SONG(SONGNUMBER) ON DELETE CASCADE,

CONSTRAINT SRAFK2 FOREIGN KEY (ARTISTID) REFERENCES RECORDINGARTIST(ARTISTID) ON DELETE CASCADE

);



CREATE TABLE CDTRACK (

CATALOGUENUMBER NUMBER,

TRACKNO NUMBER,

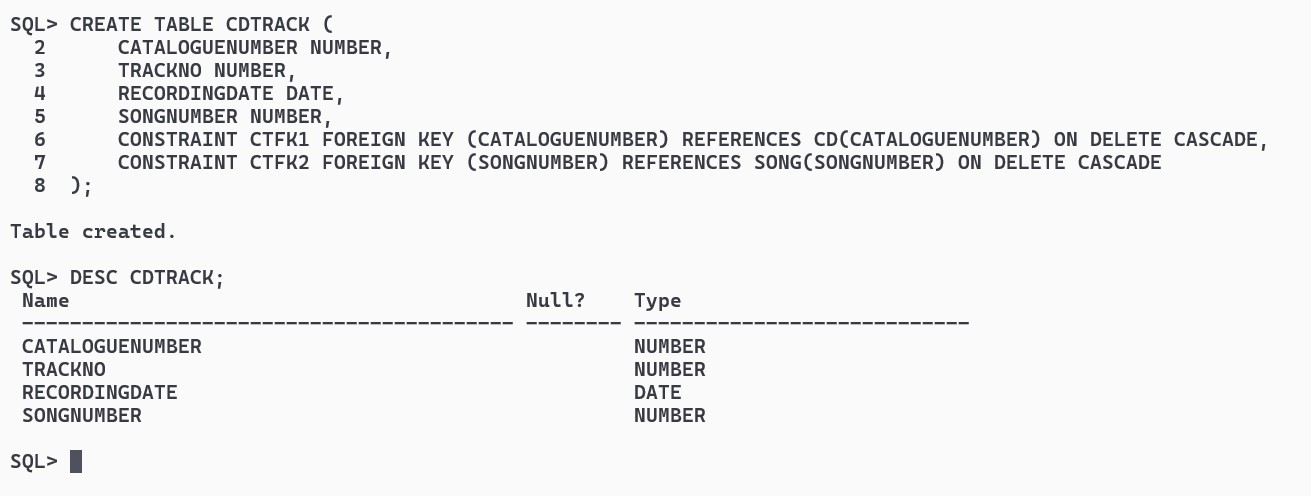
RECORDINGDATE DATE,

SONGNUMBER NUMBER,

CONSTRAINT CTFK1 FOREIGN KEY (CATALOGUENUMBER) REFERENCES CD(CATALOGUENUMBER) ON DELETE CASCADE,

CONSTRAINT CTFK2 FOREIGN KEY (SONGNUMBER) REFERENCES SONG(SONGNUMBER) ON DELETE CASCADE

);



CREATE TABLE CDRECORDINGARTIST (

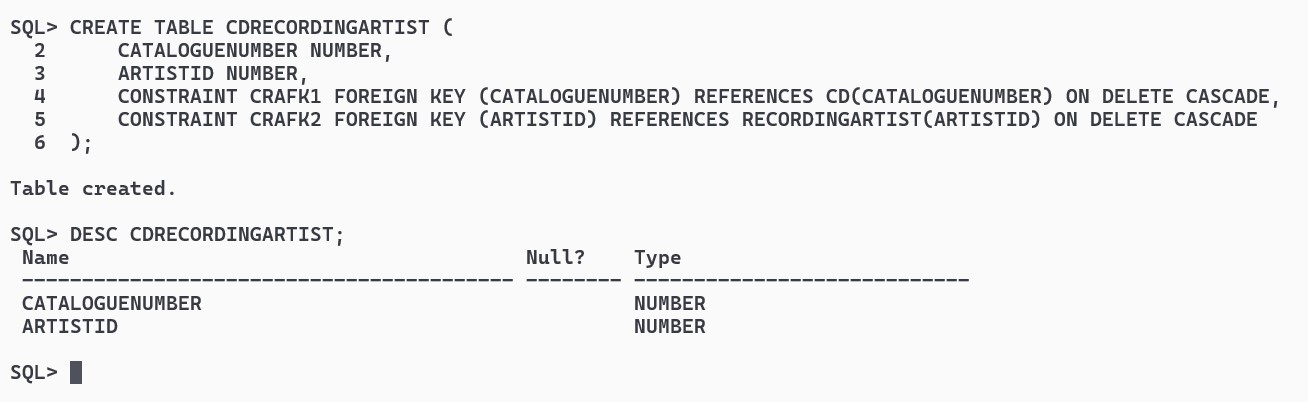
CATALOGUENUMBER NUMBER,

ARTISTID NUMBER,

CONSTRAINT CRAFK1 FOREIGN KEY (CATALOGUENUMBER) REFERENCES CD(CATALOGUENUMBER) ON DELETE CASCADE,

CONSTRAINT CRAFK2 FOREIGN KEY (ARTISTID) REFERENCES RECORDINGARTIST(ARTISTID) ON DELETE CASCADE

);



CREATE TABLE CONTRACT (

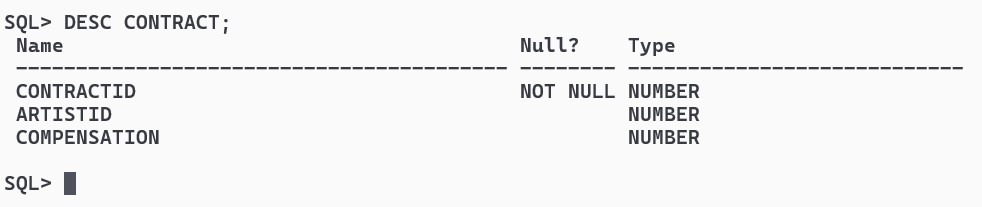
CONTRACTID NUMBER PRIMARY KEY,

ARTISTID NUMBER,

COMPENSATION NUMBER,

CONSTRAINT CNFK1 FOREIGN KEY (ARTISTID) REFERENCES RECORDINGARTIST(ARTISTID) ON DELETE CASCADE

);



INSERT ALL

INTO PRODUCER VALUES (1, 'Producer1')

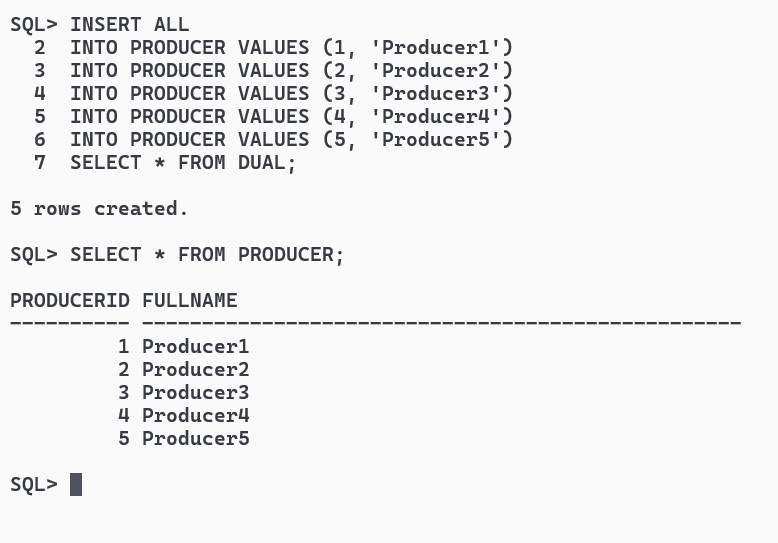
INTO PRODUCER VALUES (2, 'Producer2')

INTO PRODUCER VALUES (3, 'Producer3')

INTO PRODUCER VALUES (4, 'Producer4')

INTO PRODUCER VALUES (5, 'Producer5')

SELECT \* FROM DUAL;



INSERT ALL

INTO SONG VALUES (1, 'Song1', TO\_DATE('01-01-2023', 'DD-MM-YYYY'))

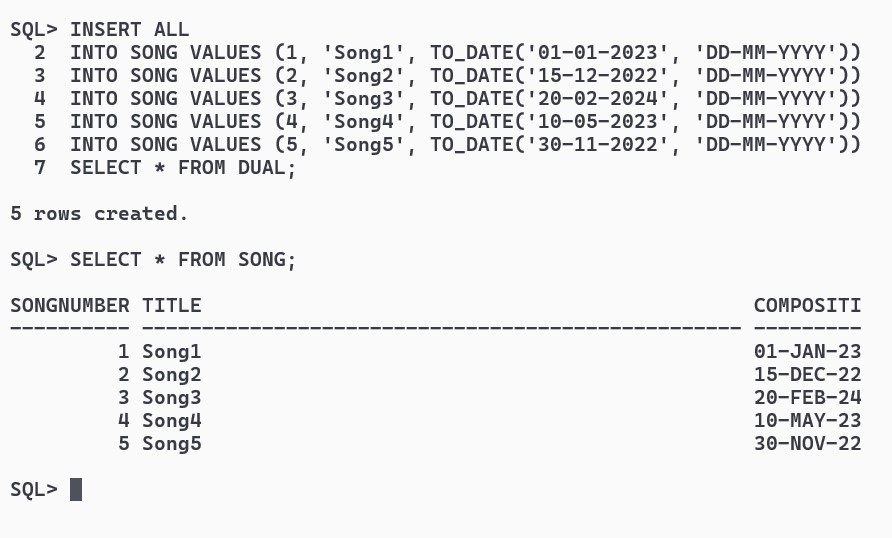
INTO SONG VALUES (2, 'Song2', TO\_DATE('15-12-2022', 'DD-MM-YYYY'))

INTO SONG VALUES (3, 'Song3', TO\_DATE('20-02-2024', 'DD-MM-YYYY'))

INTO SONG VALUES (4, 'Song4', TO\_DATE('10-05-2023', 'DD-MM-YYYY'))

INTO SONG VALUES (5, 'Song5', TO\_DATE('30-11-2022', 'DD-MM-YYYY'))

SELECT \* FROM DUAL;

INSERT ALL

INTO COMPOSER VALUES (1, 'Composer1')

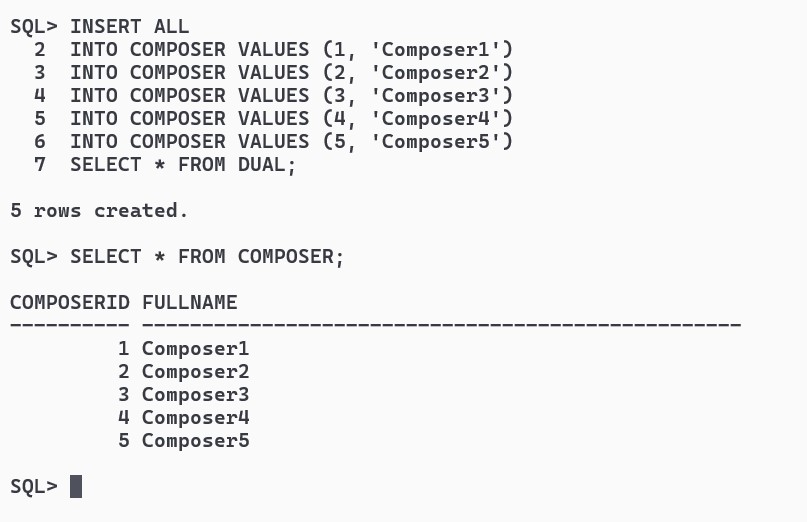
INTO COMPOSER VALUES (2, 'Composer2')

INTO COMPOSER VALUES (3, 'Composer3')

INTO COMPOSER VALUES (4, 'Composer4')

INTO COMPOSER VALUES (5, 'Composer5')

SELECT \* FROM DUAL;



INSERT ALL

INTO RECORDINGARTIST VALUES (1, 'Artist1')

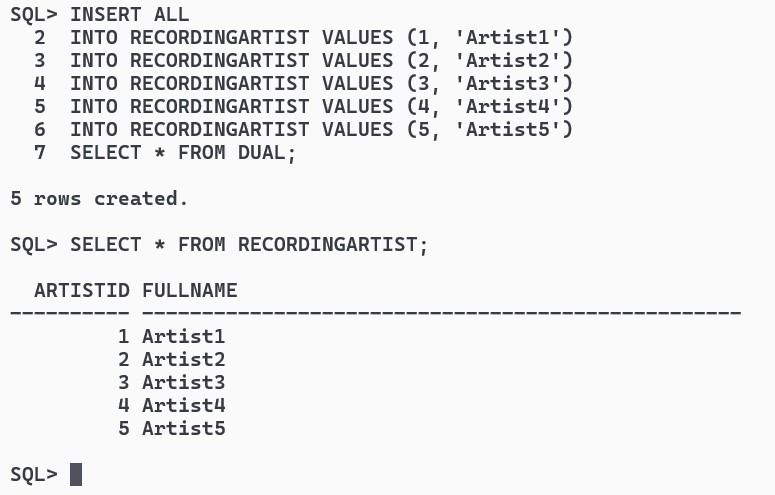
INTO RECORDINGARTIST VALUES (2, 'Artist2')

INTO RECORDINGARTIST VALUES (3, 'Artist3')

INTO RECORDINGARTIST VALUES (4, 'Artist4')

INTO RECORDINGARTIST VALUES (5, 'Artist5')

SELECT \* FROM DUAL;



INSERT ALL

INTO CD VALUES (1, 'CD1', 1, TO\_DATE('01-01-2023', 'DD-MM-YYYY'), 1000)

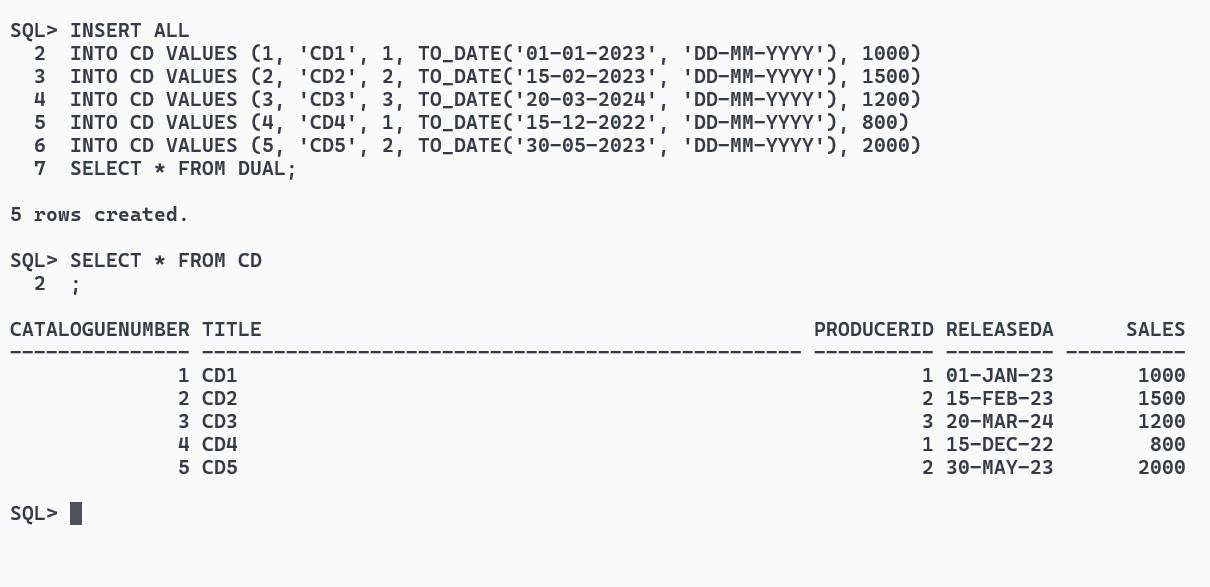
INTO CD VALUES (2, 'CD2', 2, TO\_DATE('15-02-2023', 'DD-MM-YYYY'), 1500)

INTO CD VALUES (3, 'CD3', 3, TO\_DATE('20-03-2024', 'DD-MM-YYYY'), 1200)

INTO CD VALUES (4, 'CD4', 1, TO\_DATE('15-12-2022', 'DD-MM-YYYY'), 800)

INTO CD VALUES (5, 'CD5', 2, TO\_DATE('30-05-2023', 'DD-MM-YYYY'), 2000)

SELECT \* FROM DUAL;



INSERT ALL

INTO SONGCOMPOSER VALUES (1, 1)

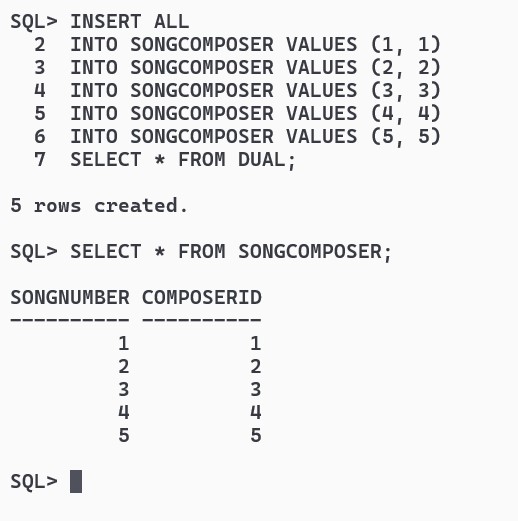
INTO SONGCOMPOSER VALUES (2, 2)

INTO SONGCOMPOSER VALUES (3, 3)

INTO SONGCOMPOSER VALUES (4, 4)

INTO SONGCOMPOSER VALUES (5, 5)

SELECT \* FROM DUAL;



INSERT ALL

INTO SONGRECORDINGARTIST VALUES (1, 1)

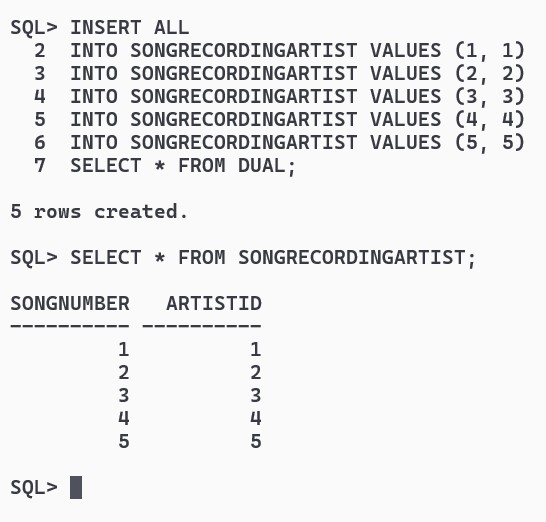
INTO SONGRECORDINGARTIST VALUES (2, 2)

INTO SONGRECORDINGARTIST VALUES (3, 3)

INTO SONGRECORDINGARTIST VALUES (4, 4)

INTO SONGRECORDINGARTIST VALUES (5, 5)

SELECT \* FROM DUAL;



INSERT ALL

INTO CDTRACK VALUES (1, 1, TO\_DATE('01-01-2023', 'DD-MM-YYYY'), 1)

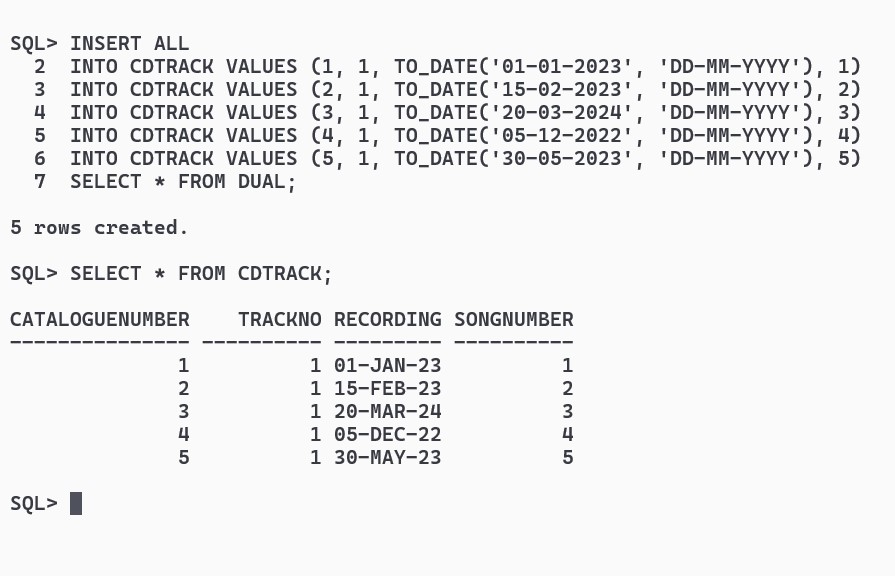
INTO CDTRACK VALUES (2, 1, TO\_DATE('15-02-2023', 'DD-MM-YYYY'), 2)

INTO CDTRACK VALUES (3, 1, TO\_DATE('20-03-2024', 'DD-MM-YYYY'), 3)

INTO CDTRACK VALUES (4, 1, TO\_DATE('05-12-2022', 'DD-MM-YYYY'), 4)

INTO CDTRACK VALUES (5, 1, TO\_DATE('30-05-2023', 'DD-MM-YYYY'), 5)

SELECT \* FROM DUAL;



INSERT ALL

INTO CDRECORDINGARTIST VALUES (1, 1)

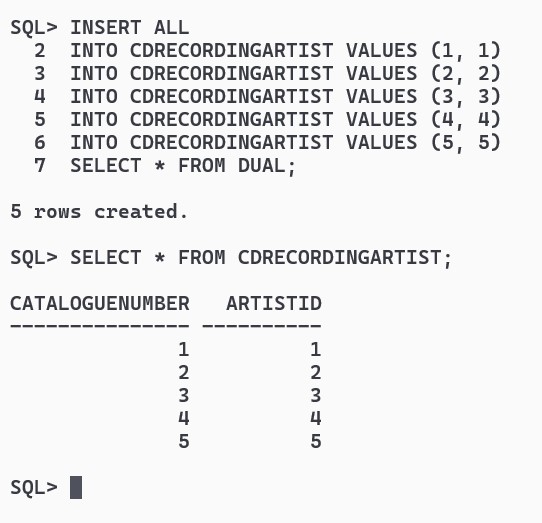
INTO CDRECORDINGARTIST VALUES (2, 2)

INTO CDRECORDINGARTIST VALUES (3, 3)

INTO CDRECORDINGARTIST VALUES (4, 4)

INTO CDRECORDINGARTIST VALUES (5, 5)

SELECT \* FROM DUAL;



SQL:-

i>Update number of recorded album to 4 for those artist who has recorded only 3.

ALTER TABLE RECORDINGARTIST ADD SALES NUMBER;

ALTER TABLE RECORDINGARTIST RENAME COLUMN SALES TO ALBUMS;

UPDATE RECORDINGARTIST SET ALBUMS = 1 WHERE ARTISTID = 1;

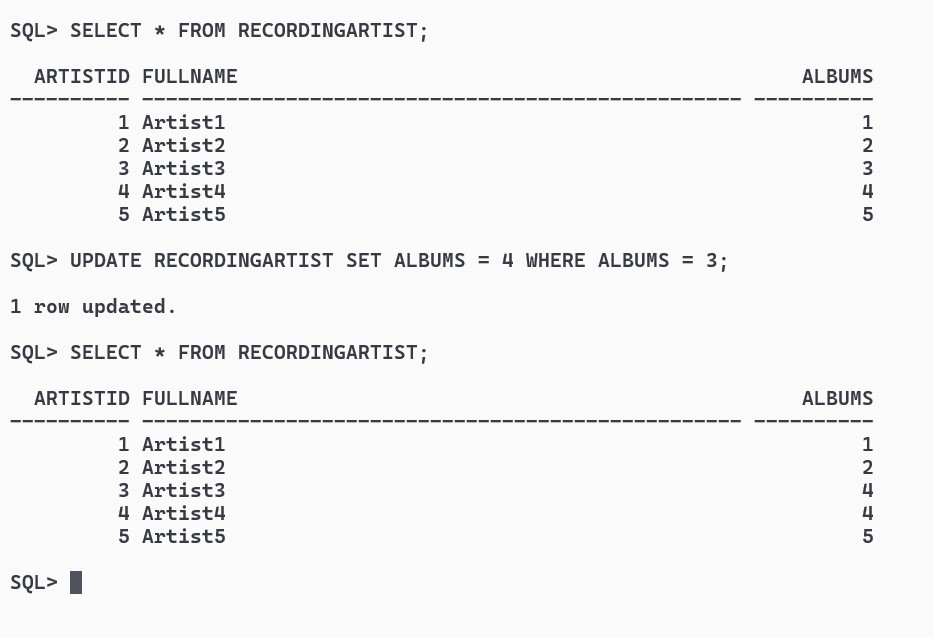
UPDATE RECORDINGARTIST SET ALBUMS = 2 WHERE ARTISTID = 2;

UPDATE RECORDINGARTIST SET ALBUMS = 3 WHERE ARTISTID = 3;

UPDATE RECORDINGARTIST SET ALBUMS = 4 WHERE ARTISTID = 4;

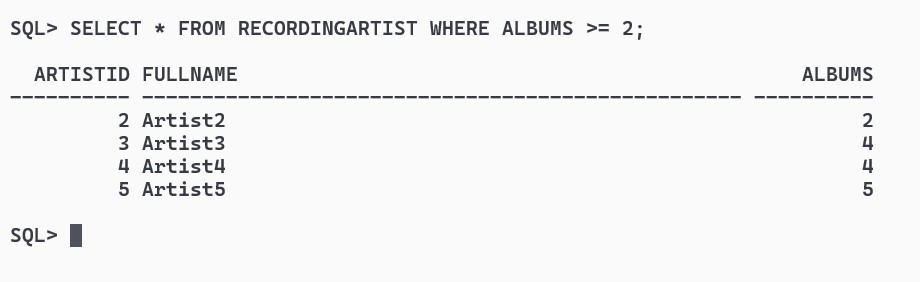
UPDATE RECORDINGARTIST SET ALBUMS = 5 WHERE ARTISTID = 5;

UPDATE RECORDINGARTIST SET ALBUMS = 4 WHERE ALBUMS = 3;



ii>Find all artists who have recorded at least two albums.

SELECT \* FROM RECORDINGARTIST WHERE ALBUMS >= 2;



iii>Find all writers who have only written one song.

CREATE TABLE COUNTSONG AS SELECT COUNT(SONGNUMBER) TOTSONG, COMPOSERID FROM SONGCOMPOSER GROUP BY COMPOSERID;

SELECT C.COMPOSERID, C.FULLNAME FROM COMPOSER C JOIN COUNTSONG CS ON CS.COMPOSERID = C.COMPOSERID WHERE CS.TOTSONG = 1;



PL/SQL

i>Write Procedure to insert a new Contract into the Contract relation.

CREATE OR REPLACE PROCEDURE INSERTCONTRACT (CID IN NUMBER, AID IN NUMBER, COMP IN NUMBER) AS

BEGIN

INSERT INTO CONTRACT VALUES(CID,AID,COMP);

COMMIT;

END;

/

