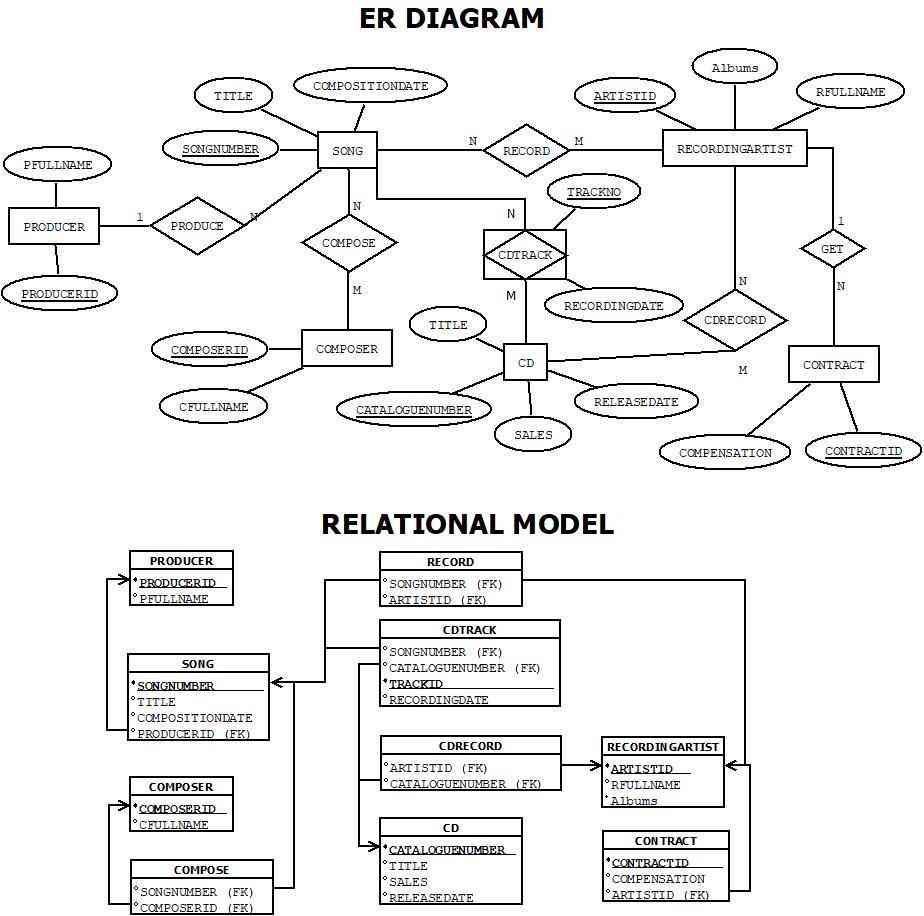
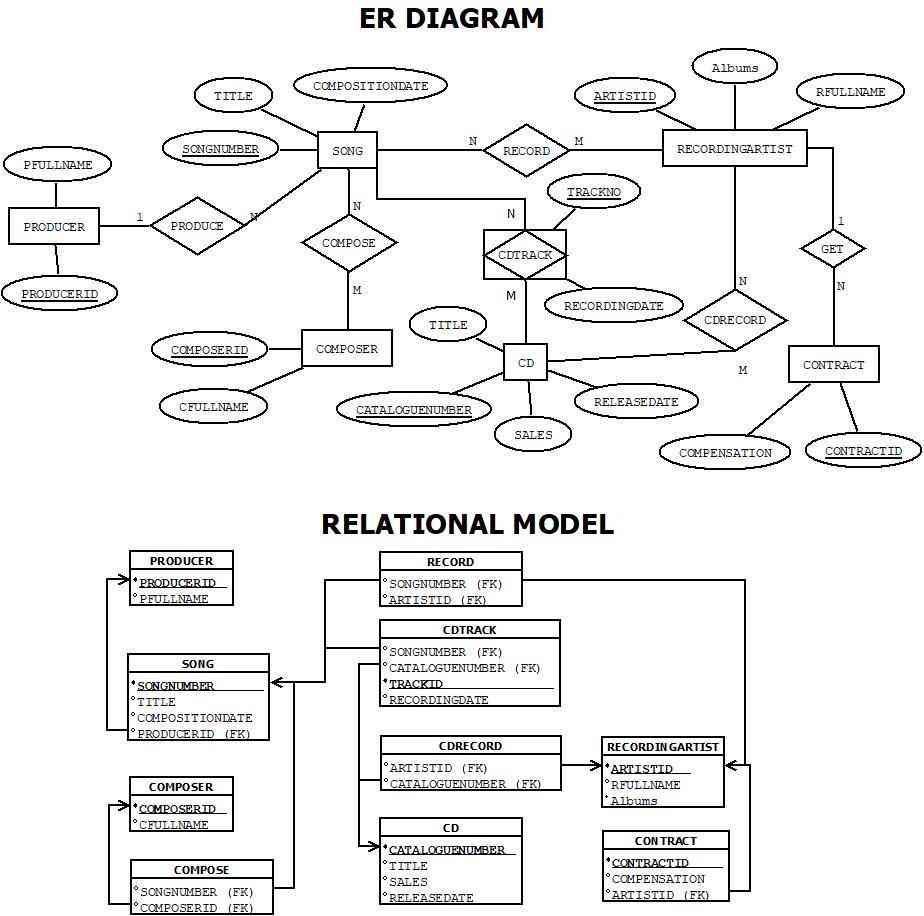
**ASSIGNMENT – 11**

**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.**





1. **Compile DDL and DML commands on the database created.**

-- CREATE TABLES

set linesize 50;

-- CREATE TABLE PRODUCER

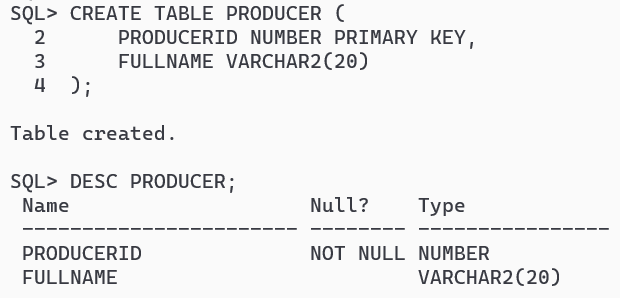
CREATE TABLE PRODUCER (

PRODUCERID NUMBER PRIMARY KEY,

FULLNAME VARCHAR2(20)

);

DESC PRODUCER;



-- CREATE TABLE SONG

CREATE TABLE SONG (

SONGNUMBER NUMBER PRIMARY KEY,

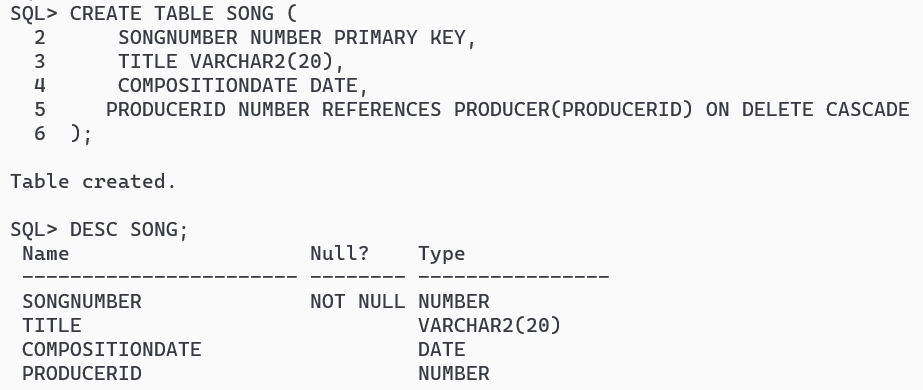
TITLE VARCHAR2(20),

COMPOSITIONDATE DATE,

PRODUCERID NUMBER REFERENCES PRODUCER(PRODUCERID) ON DELETE CASCADE

);

DESC SONG;



-- CREATE TABLE COMPOSER

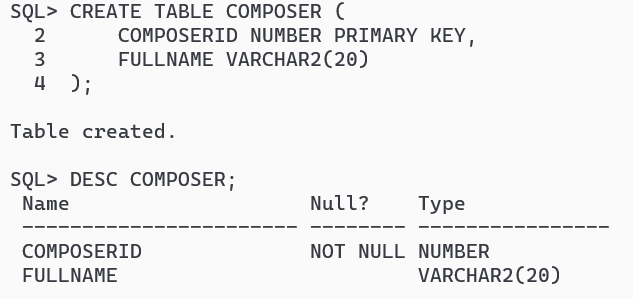
CREATE TABLE COMPOSER (

COMPOSERID NUMBER PRIMARY KEY,

FULLNAME VARCHAR2(20)

);

DESC COMPOSER;



-- CREATE TABLE RECORDINGARTIST

CREATE TABLE RECORDINGARTIST (

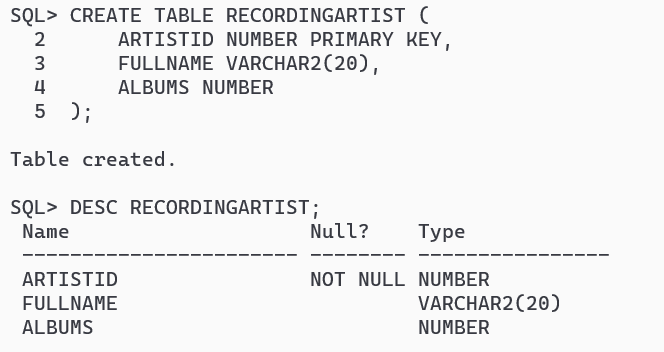
ARTISTID NUMBER PRIMARY KEY,

FULLNAME VARCHAR2(20),

ALBUMS NUMBER

);

DESC RECORDINGARTIST;



-- CREATE TABLE CD

CREATE TABLE CD (

CATALOGUENUMBER NUMBER PRIMARY KEY,

TITLE VARCHAR2(20),

PRODUCERID NUMBER,

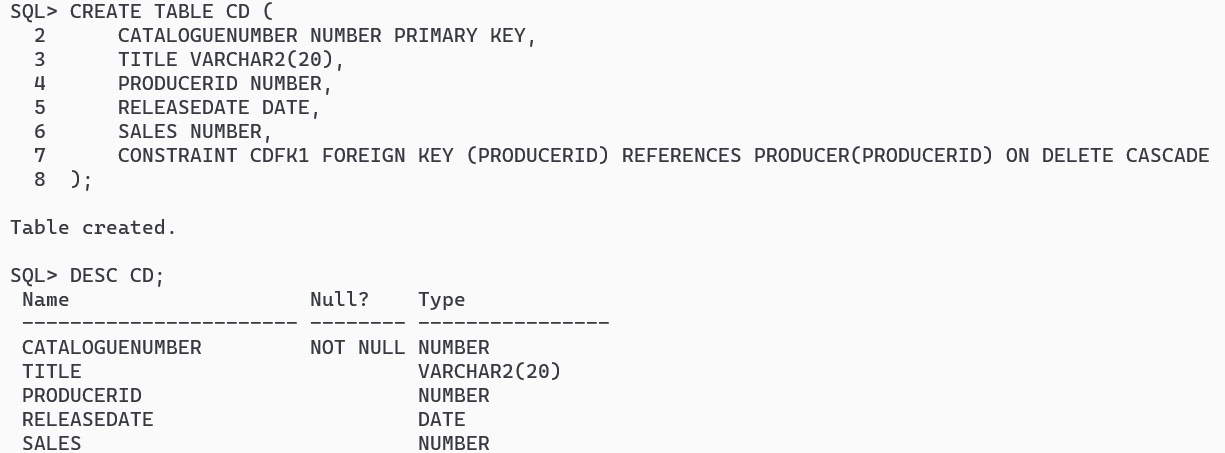
RELEASEDATE DATE,

SALES NUMBER,

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

);

DESC CD;



-- CREATE TABLE SONGCOMPOSER

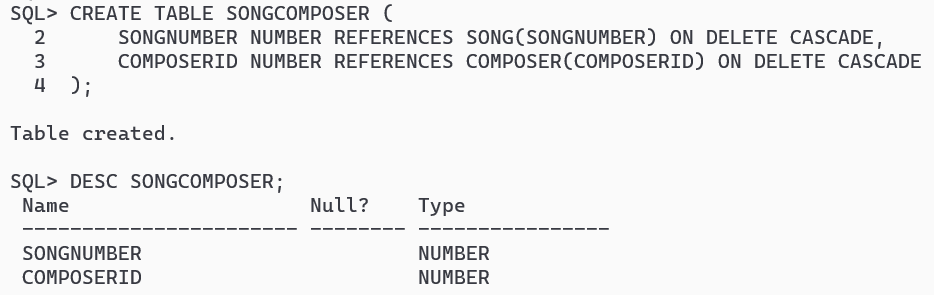
CREATE TABLE SONGCOMPOSER (

SONGNUMBER NUMBER REFERENCES SONG(SONGNUMBER) ON DELETE CASCADE,

COMPOSERID NUMBER REFERENCES COMPOSER(COMPOSERID) ON DELETE CASCADE

);

DESC SONGCOMPOSER;



-- CREATE TABLE SONGRECORDINGARTIST

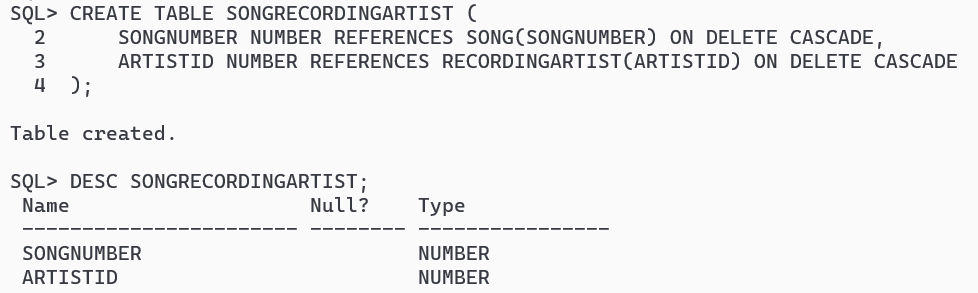
CREATE TABLE SONGRECORDINGARTIST (

SONGNUMBER NUMBER REFERENCES SONG(SONGNUMBER) ON DELETE CASCADE,

ARTISTID NUMBER REFERENCES RECORDINGARTIST(ARTISTID) ON DELETE CASCADE

);

DESC SONGRECORDINGARTIST;



-- CREATE TABLE CDTRACK

CREATE TABLE CDTRACK (

CATALOGUENUMBER NUMBER REFERENCES CD(CATALOGUENUMBER) ON DELETE CASCADE,

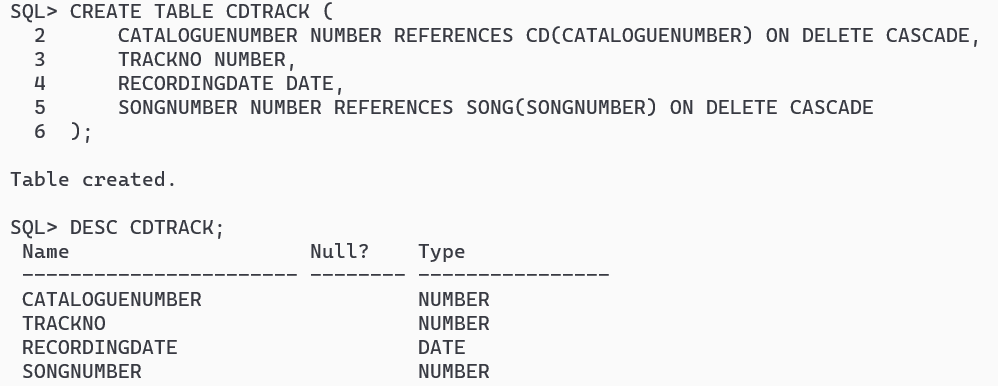
TRACKNO NUMBER,

RECORDINGDATE DATE,

SONGNUMBER NUMBER REFERENCES SONG(SONGNUMBER) ON DELETE CASCADE

);

DESC CDTRACK;



-- CREATE TABLE CDRECORDINGARTIST

CREATE TABLE CDRECORDINGARTIST (

CATALOGUENUMBER NUMBER,

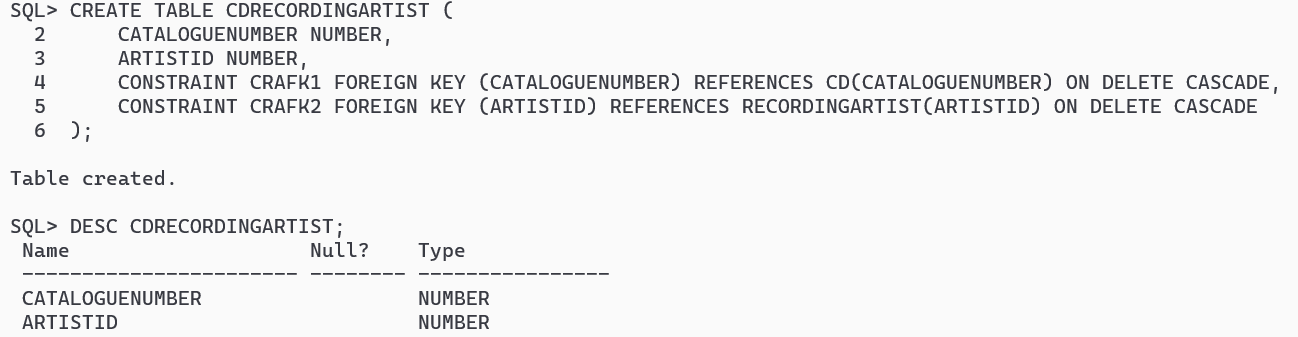
ARTISTID NUMBER,

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

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

);

DESC CDRECORDINGARTIST;



CREATE TABLE CONTRACT (

CONTRACTID NUMBER PRIMARY KEY,

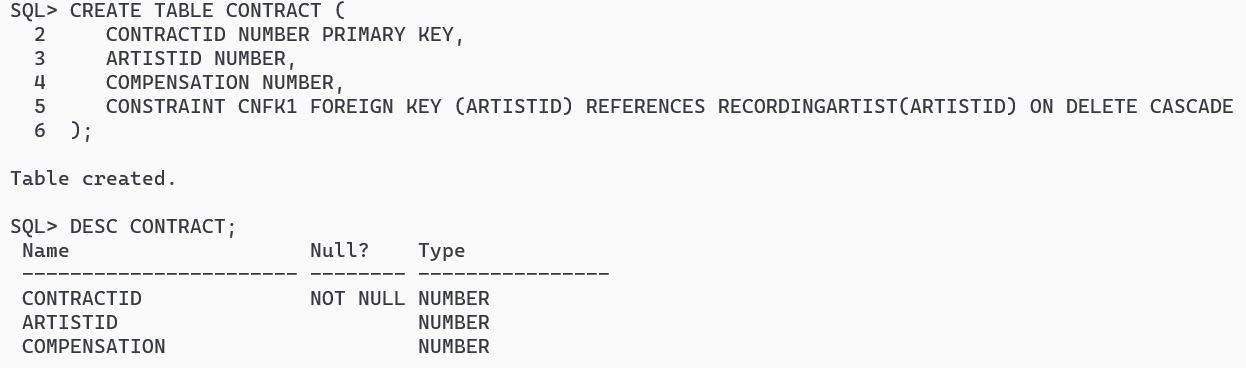
ARTISTID NUMBER,

COMPENSATION NUMBER,

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

);

DESC CONTRACT;



-- INSERT VALUES

SET LINESIZE 300;

-- Inserting values into PRODUCER table

INSERT ALL

INTO PRODUCER VALUES (1, 'Dr. Dre')

INTO PRODUCER VALUES (2, 'Pharrell Williams')

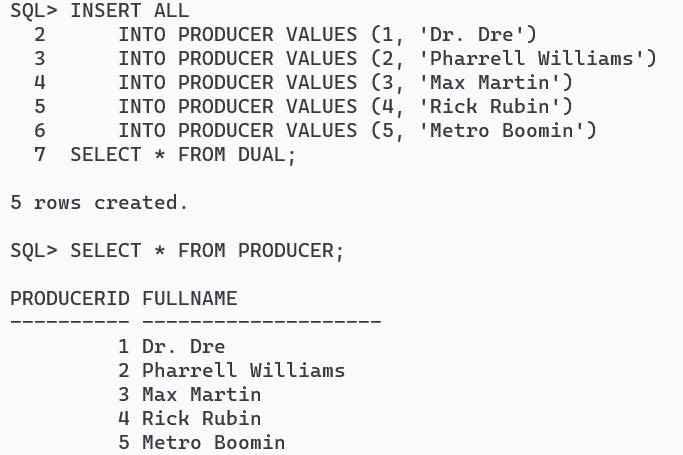
INTO PRODUCER VALUES (3, 'Max Martin')

INTO PRODUCER VALUES (4, 'Rick Rubin')

INTO PRODUCER VALUES (5, 'Metro Boomin')

SELECT \* FROM DUAL;

SELECT \* FROM PRODUCER;



-- Inserting values into SONG table

INSERT ALL

INTO SONG VALUES (1, 'Song1', TO\_DATE ('12-04-2025', 'DD-MM-YYYY'), 1)

INTO SONG VALUES (2, 'Song2', TO\_DATE ('14-08-2023', 'DD-MM-YYYY'), 2)

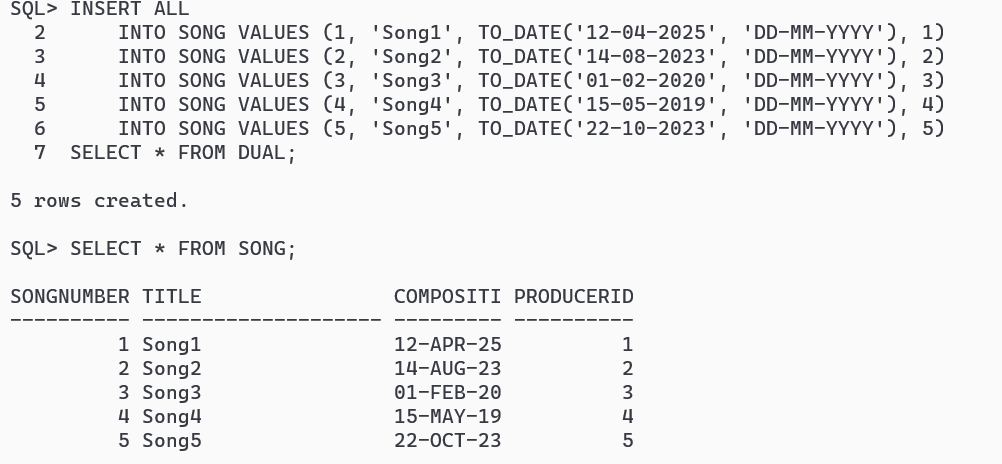
INTO SONG VALUES (3, 'Song3', TO\_DATE ('01-02-2020', 'DD-MM-YYYY'), 3)

INTO SONG VALUES (4, 'Song4', TO\_DATE ('15-05-2019', 'DD-MM-YYYY'), 4)

INTO SONG VALUES (5, 'Song5', TO\_DATE ('22-10-2023', 'DD-MM-YYYY'), 5)

SELECT \* FROM DUAL;

SELECT \* FROM SONG;



-- Inserting values into COMPOSER table

INSERT ALL

INTO COMPOSER VALUES (1, 'John Williams')

INTO COMPOSER VALUES (2, 'Hans Zimmer')

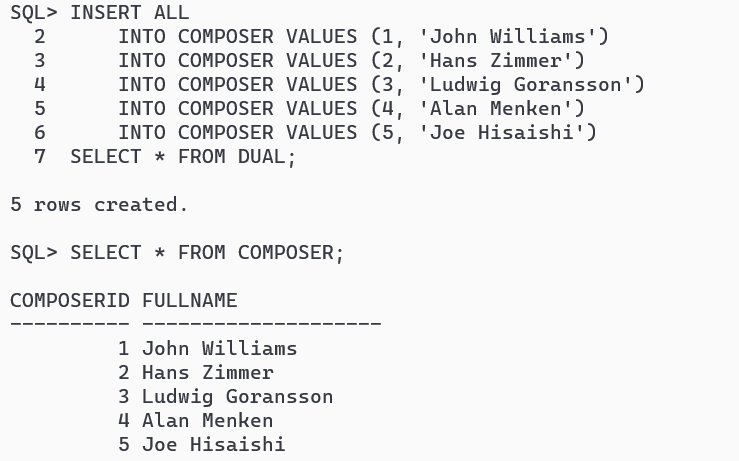
INTO COMPOSER VALUES (3, 'Ludwig Goransson')

INTO COMPOSER VALUES (4, 'Alan Menken')

INTO COMPOSER VALUES (5, 'Joe Hisaishi')

SELECT \* FROM DUAL;

SELECT \* FROM COMPOSER;



-- Inserting values into RECORDINGARTIST table

INSERT ALL

INTO RECORDINGARTIST VALUES (1, 'Adele', 2)

INTO RECORDINGARTIST VALUES (2, 'Drake', 3)

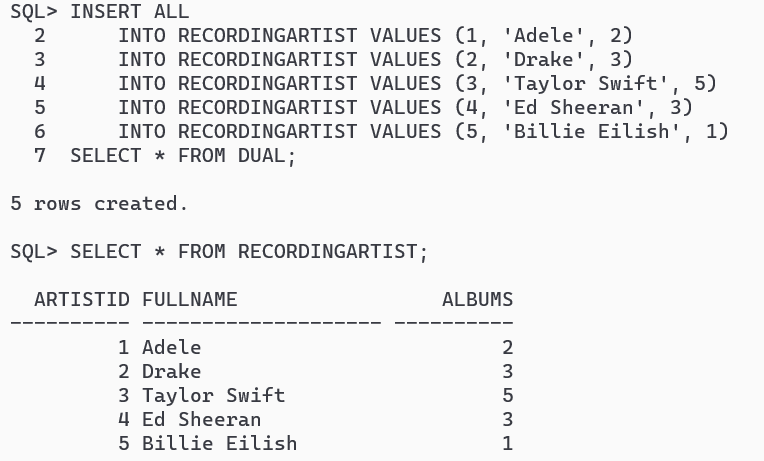
INTO RECORDINGARTIST VALUES (3, 'Taylor Swift', 5)

INTO RECORDINGARTIST VALUES (4, 'Ed Sheeran', 3)

INTO RECORDINGARTIST VALUES (5, 'Billie Eilish', 1)

SELECT \* FROM DUAL;

SELECT \* FROM RECORDINGARTIST;



-- Inserting values into CD table

INSERT ALL

INTO CD VALUES (1, 'CD1', 1, TO\_DATE ('01-01-2023', 'DD-MM-YYYY'), 960)

INTO CD VALUES (2, 'CD2', 2, TO\_DATE ('15-02-2023', 'DD-MM-YYYY'), 1300)

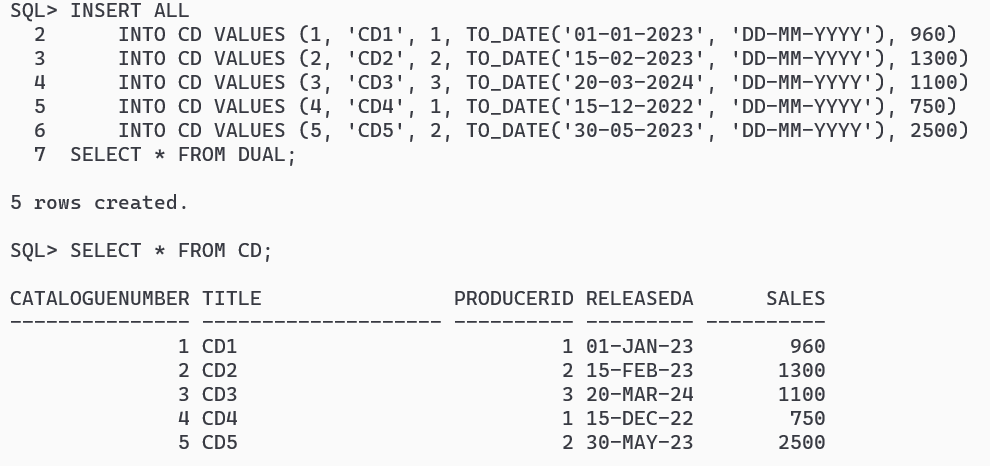
INTO CD VALUES (3, 'CD3', 3, TO\_DATE ('20-03-2024', 'DD-MM-YYYY'), 1100)

INTO CD VALUES (4, 'CD4', 1, TO\_DATE ('15-12-2022', 'DD-MM-YYYY'), 750)

INTO CD VALUES (5, 'CD5', 2, TO\_DATE ('30-05-2023', 'DD-MM-YYYY'), 2500)

SELECT \* FROM DUAL;

SELECT \* FROM CD;



-- Inserting values into SONGCOMPOSER table

INSERT ALL

INTO SONGCOMPOSER VALUES (1, 1)

INTO SONGCOMPOSER VALUES (2, 2)

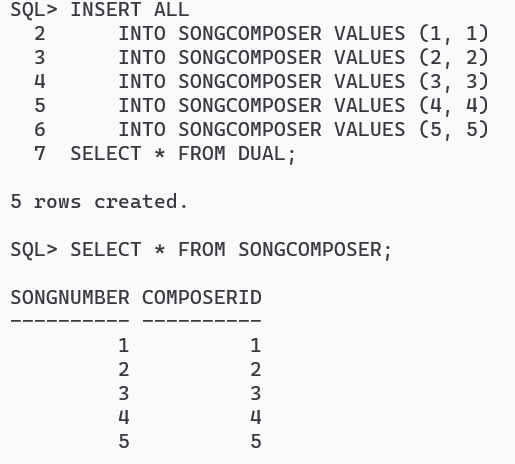
INTO SONGCOMPOSER VALUES (3, 3)

INTO SONGCOMPOSER VALUES (4, 4)

INTO SONGCOMPOSER VALUES (5, 5)

SELECT \* FROM DUAL;

SELECT \* FROM SONGCOMPOSER;



-- Inserting values into SONGRECORDINGARTIST table

INSERT ALL

INTO SONGRECORDINGARTIST VALUES (1, 1)

INTO SONGRECORDINGARTIST VALUES (2, 2)

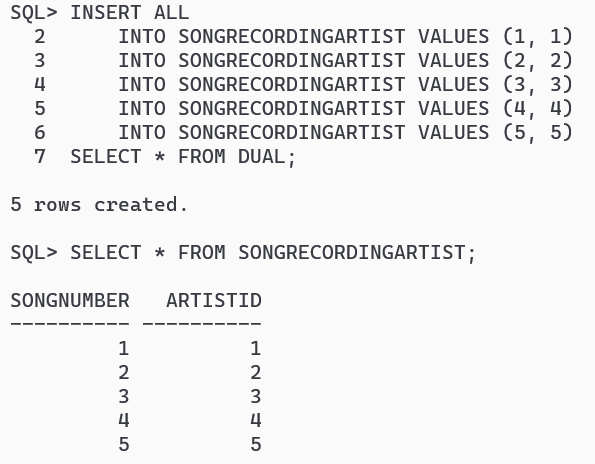
INTO SONGRECORDINGARTIST VALUES (3, 3)

INTO SONGRECORDINGARTIST VALUES (4, 4)

INTO SONGRECORDINGARTIST VALUES (5, 5)

SELECT \* FROM DUAL;

SELECT \* FROM SONGRECORDINGARTIST;



-- Inserting values into CDTRACK table

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)

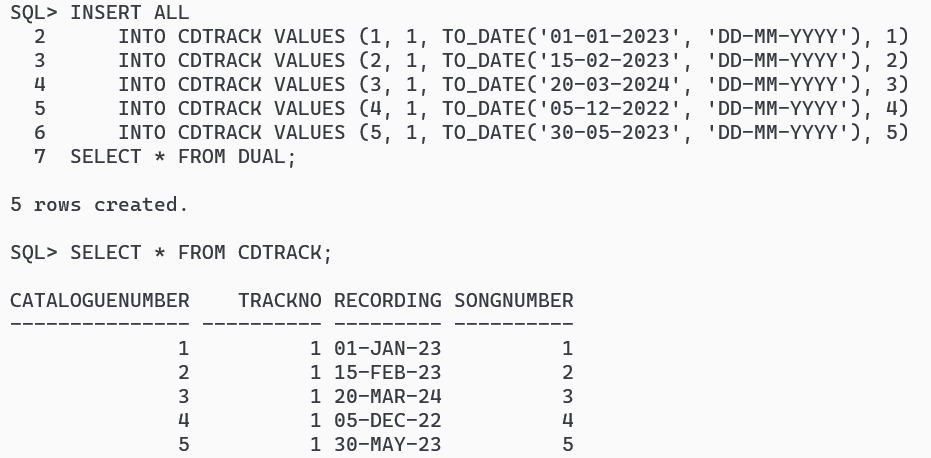
INTO CDTRACK VALUES (3, 1, TO\_DATE ('20-03-2024', 'DD-MM-YYYY'), 3)

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;

SELECT \* FROM CDTRACK;



-- Inserting values into CDRECORDINGARTIST table

INSERT ALL

INTO CDRECORDINGARTIST VALUES (1, 1)

INTO CDRECORDINGARTIST VALUES (2, 2)

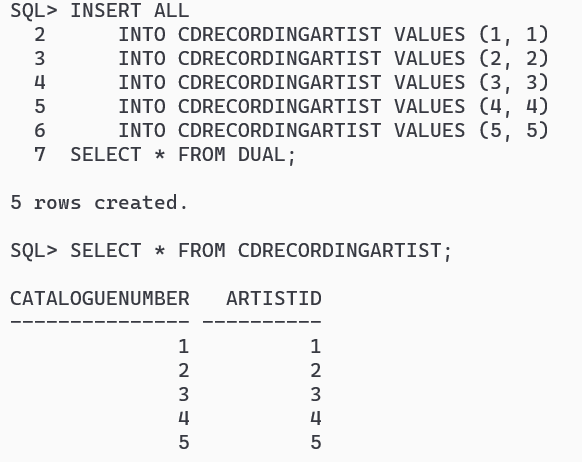
INTO CDRECORDINGARTIST VALUES (3, 3)

INTO CDRECORDINGARTIST VALUES (4, 4)

INTO CDRECORDINGARTIST VALUES (5, 5)

SELECT \* FROM DUAL;

SELECT \* FROM CDRECORDINGARTIST;



**SQL: -**

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

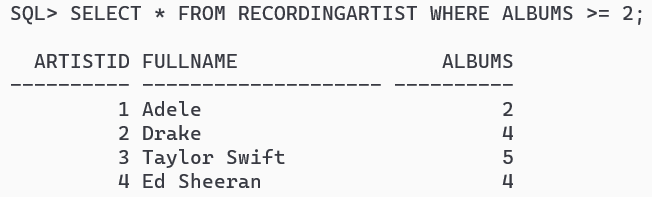
UPDATE RECORDINGARTIST SET ALBUMS = 4 WHERE ALBUMS = 3;

SELECT \* FROM RECORDINGARTIST;



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

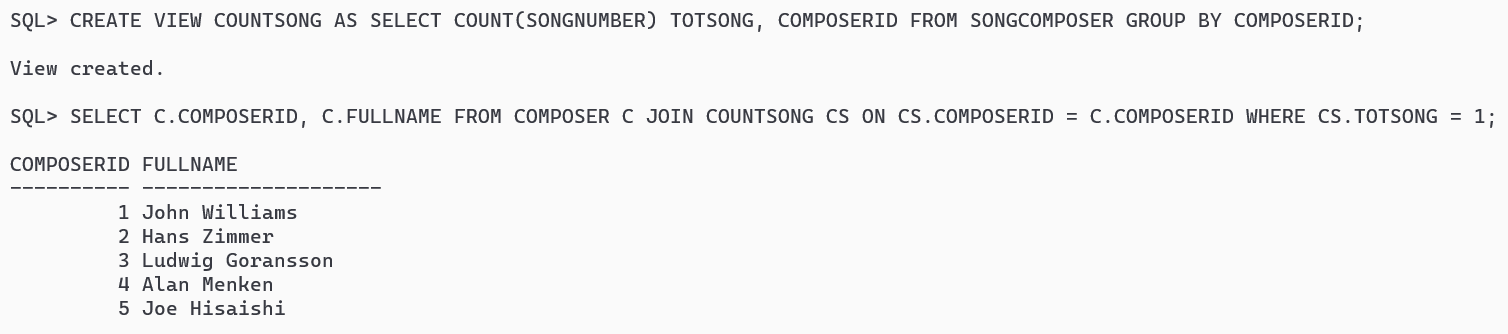
SELECT \* FROM RECORDINGARTIST WHERE ALBUMS >= 2;



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

CREATE VIEW 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;

/

EXEC INSERTCONTRACT(1, 1, 1000);

EXEC INSERTCONTRACT(2, 2, 2000);

EXEC INSERTCONTRACT(3, 3, 3000);

EXEC INSERTCONTRACT(4, 4, 4000);

EXEC INSERTCONTRACT(5, 5, 5000);

SELECT \* FROM CONTRACT;

