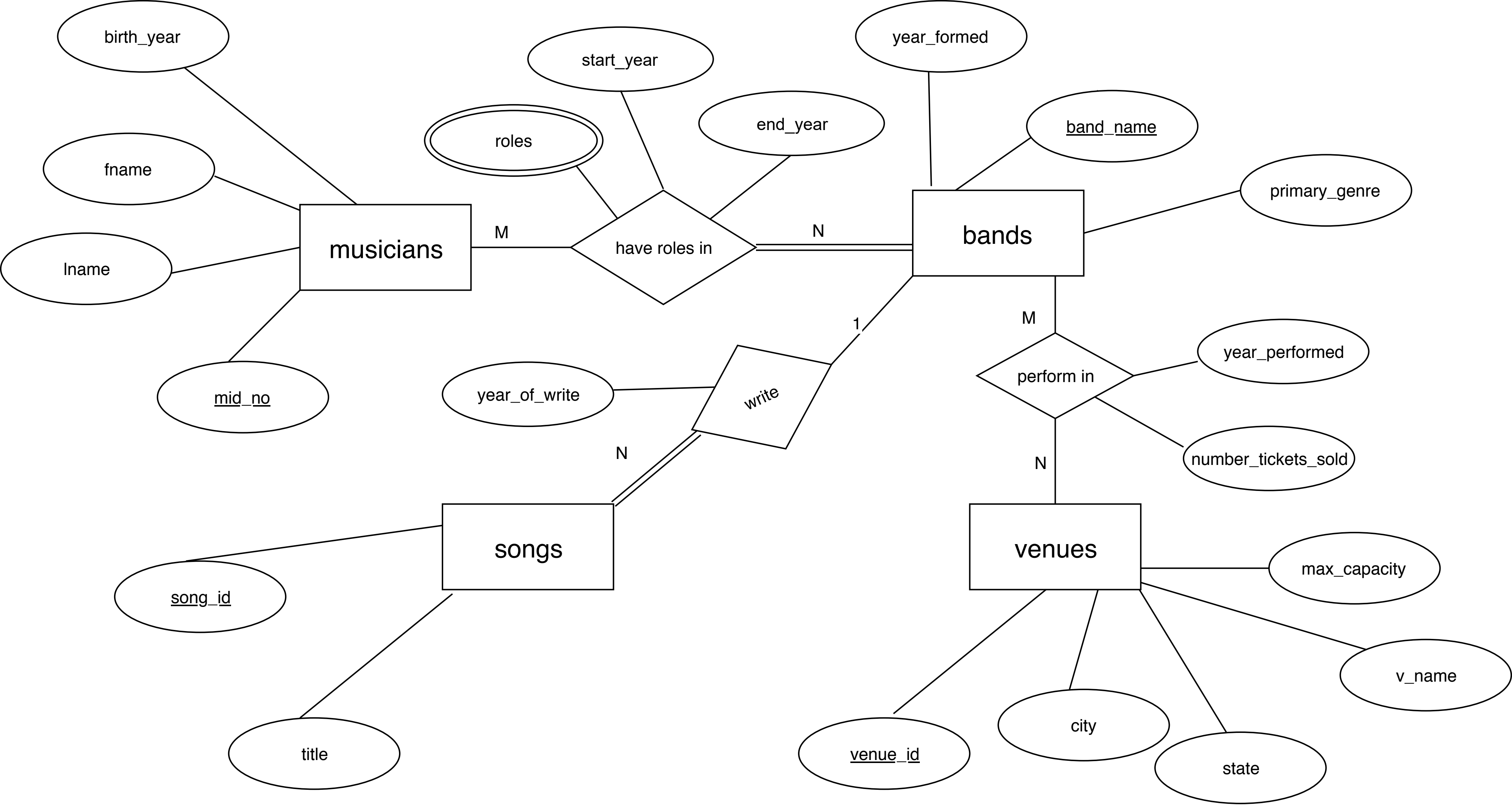
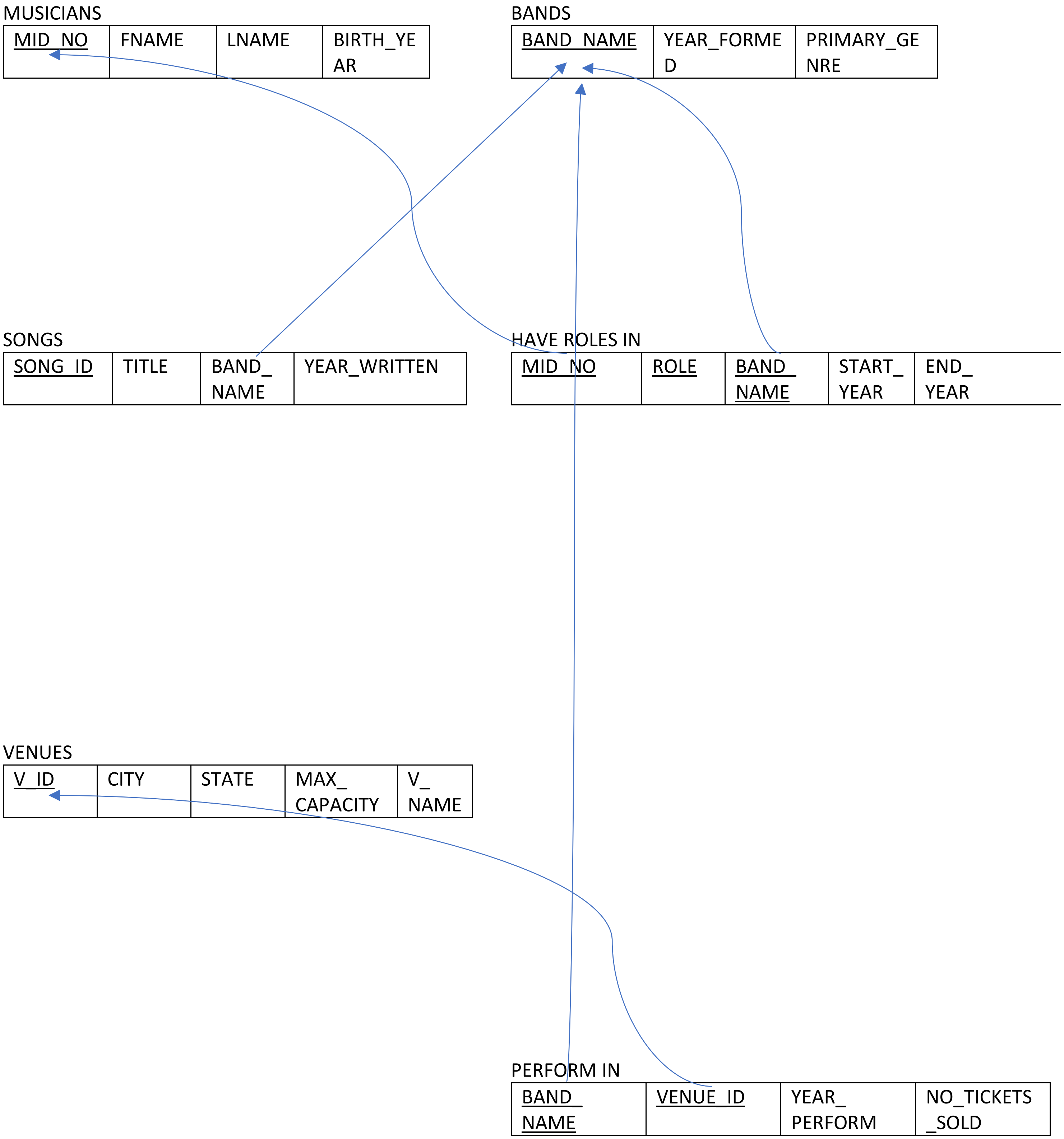

birth_year
start_year
year_formed
roles
end_year
band_name
fname
primary_genre
M
N
musicians
have roles in
bands
lname
1
M
year_performed
perform in
year_of_write
write
mid_no
N
N
number_tickets_sold
songs
venues
max_capacity
song_id
v_name
city
title
venue_id
state



MUSICIANS

MID_NO	FNAME	LNAME	BIRTH_YEAR

BANDS

BAND_NAME	YEAR_FORMED	PRIMARY_GENRE

SONGS

SONG_ID	TITLE	BAND_NAME	YEAR_WRITTEN

HAVE ROLES IN

MID_NO	ROLE	BAND_NAME	START_YEAR	END_YEAR

VENUES

V_ID	CITY	STATE	MAX_CAPACITY	V_NAME

PERFORM IN

BAND_NAME	VENUE_ID	YEAR_PERFORM	NO_TICKETS_SOLD

```
create table BANDS (
   BAND_NAME varchar(30) ,
   YEAR_FORMED year ,
   PRIMARY_GENRE varchar (20) ,
   primary key (BAND_NAME)
);
```

```
create table MUSICIANS (
   MID_NO int ,
   FNAME varchar (20) ,
   LNAME varchar (20) ,
   BIRTH_YEAR year ,
   primary key (MID_NO)
) ;
```

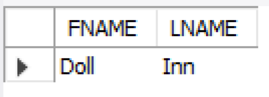
```
create table SONGS (
    SONG_ID int ,
    TITLE text ,
    BAND_NAME varchar (30) ,
    YEAR_WRITTEN year,
    check (YEAR_WRITTEN >= BANDS
(YEAR_FORMED) ) ,
    primary key (SONG_ID),
    foreign key (BAND_NAME) references BANDS
(BAND_NAME)
) ;
```

```
create table HAVE_ROLES_IN (
     MID_NO int ,
     BAND_NAME varchar (30) ,
     START_YEAR year ,
     END_YEAR year,
     ROLE varchar (20) ,
     check (START_YEAR <= END_YEAR) ,
     check (START_YEAR >= MUSICIANS
(BIRTH_YEAR) ),
     check (START_YEAR >= BANDS
(YEAR_FORMED) ),
     primary key (MID_NO, BAND_NAME, ROLE),
     foreign key (MID_NO) references MUSICIANS
(MID_NO),
     foreign key (BAND_NAME) references BANDS
(BAND_NAME) );
```

```
create table VENUES (
     V_ID int ,
     CITY varchar (20) not null ,
     STATE char (2) not null,
     MAX_CAPACITY int,
     V_NAME varchar (20),
     primary key (V_ID)
) ;
```

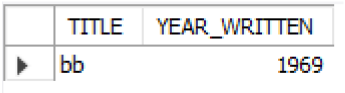
```
create table PERFORM_IN (
    BAND_NAME varchar (30) ,
    VENUE_ID int,
    YEAR_PERFORM year,
    NO_TICKETS_SOLD int,
    primary key (BAND_NAME, VENUE_ID),
    check (NO_TICKETS_SOLD >= 0) ,
    check (NO_TICKETS_SOLD <= VENUES
(MAX_CAPACITY) ) ,
    check (YEAR_PERFORM >=
BANDS(YEAR_FORMED) ) ,
    foreign key (BAND_NAME) references BANDS
(BAND_NAME),
    foreign key (VENUE_ID) references VENUES
(V_ID)
) ;
```

```
SELECT M.FNAME, M.LNAME
FROM MUSICIANS M
WHERE M.BIRTH_YEAR > '1980' and
      M.MID_NO IN (SELECT H.MID_NO
                   FROM HAVE_ROLES_IN H
                   WHERE H.ROLE='drummer') ;
```



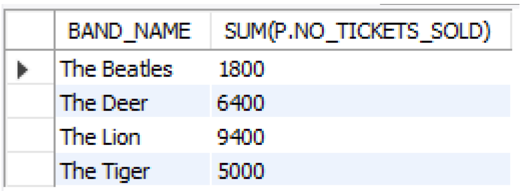
	FNAME	LNAME
▶	Doll	Inn

```
SELECT S.TITLE, S.YEAR_WRITTEN
FROM SONGS S
WHERE S.BAND_NAME='The Beatles' and
      S. YEAR_WRITTEN > '1968' ;
```



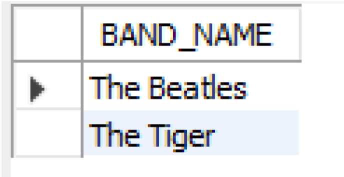
	TITLE	YEAR_WRITTEN
▶	bb	1969

```
SELECT B.BAND_NAME,
SUM(P.NO_TICKETS_SOLD)
FROM BANDS B, PERFORM_IN P, VENUES V
WHERE B.BAND_NAME=P.BAND_NAME AND
      V.V_ID=P.VENUE_ID AND
      V.MAX_CAPACITY >= 1000
      GROUP BY P.BAND_NAME;
```



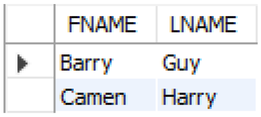
	BAND_NAME	SUM(P.NO_TICKETS_SOLD)
▶	The Beatles	1800
	The Deer	6400
	The Lion	9400
	The Tiger	5000

```
SELECT distinct B.BAND_NAME
FROM BANDS B, PERFORM_IN P
WHERE B.BAND_NAME=P.BAND_NAME AND
      B.BAND_NAME NOT IN (SELECT P.BAND_NAME
                                   FROM PERFORM_IN P
                          WHERE P.YEAR_PERFORM >'1990' AND
                                P.YEAR_PERFORM < '1995' ) ;
```



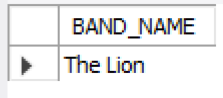
	BAND_NAME
▶	The Beatles
	The Tiger

```
SELECT distinct M.FNAME, M.LNAME
FROM MUSICIANS M, HAVE_ROLES_IN H, PERFORM_IN P, VENUES V
WHERE M.MID_NO=H.MID_NO AND
      H.BAND_NAME=P.BAND_NAME AND
      V.V_ID=P.VENUE_ID AND
      H.ROLE='guitar' and
      P.BAND_NAME NOT IN (SELECT P.BAND_NAME
                          FROM PERFORM_IN P, BANDS B, VENUES V
                          WHERE P.BAND_NAME=B.BAND_NAME AND
                                            P.VENUE_ID=V.V_ID AND
                                V.V_NAME='Red Rocks') ;
```



	FNAME	LNAME
▶	Barry	Guy
	Camen	Harry

```
SELECT B.BAND_NAME

FROM BANDS B, PERFORM_IN P1, PERFORM_IN P2

WHERE B.BAND_NAME=P1.BAND_NAME AND

      P1.BAND_NAME=P2.BAND_NAME AND

      P1.VENUE_ID=002 AND

      P2.VENUE_ID=003 AND

      P1.NO_TICKETS_SOLD > P2.NO_TICKETS_SOLD AND

      P1.YEAR_PERFORM IN (SELECT P.YEAR_PERFORM

                          FROM BANDS B, PERFORM_IN P, VENUES V

                          WHERE B.BAND_NAME=P.BAND_NAME AND

                                P.VENUE_ID=V.V_ID AND

                                V.V_NAME = 'Carolina Theater' AND

                                P.YEAR_PERFORM IN ( SELECT P.YEAR_PERFORM

                                                    FROM BANDS B, PERFORM_IN P, VENUES V

                                                    WHERE B.BAND_NAME=P.BAND_NAME AND

                                                          P.VENUE_ID=V.V_ID AND

                                                          V.V_NAME= 'Wolf Trap') ) ;
```



	BAND_NAME
▶	The Lion