
EDAT exercises.


6. Write SQL queries to obtain the following information from the database of exercise 3:

a. Airlines without flights departing from London.

```
select *
from airline
where not exists (select *
                  from flight, airport
                  where flight.airline=IATA_code and origin=code and city='London')
```

b. Sold out flights, indicating the date.

```
select number, date
from flight, (select count(*) as c, f.number
              from ticket as t, flight as f
              where t.flight=f.number
              group by t.flight, f.number) as x
where seats=x.c and x.number=number
```



c. Empty flights (with no sold tickets) on January 1 st 2020.

```
select *
from flight
where date>2020-01-01
and not exists (
    select *
    from ticket
    where ticket.flight=flight.number
)
```

d. Airlines that only operate flights departing from or arriving in Madrid.

```
select *
from airline
where IATA_code not in (
    select airline
    from flights, airport
    where city='Madrid'
    and origin<>code and destination<>code
)
```

8. Write SQL queries to obtain the following information from the database of exercise 3:

a. Airport with most traffic (counting departures and arrivals).

```
select count(*) as c, code
from airport, flight
where code=destination or code=origin
group by code
order by c desc
limit 1
```

b. Airlines ordered by the total number of tickets sold in the flights they operate.

```
select count(*) as c, name
from airline, ticket
where airline=IATA_code
order by c desc
```

c. Name of the city from which the very first flight of the day departs.

```
select city
from airport, flight
where date(timestamp)='YYYY-MM-DD' and code=origin
order by timestamp asc
limit 1
```

d. Average total spending by passenger.

```
select avg(price), name
from passenger as p, ticket as t
where t.passenger=p.id
group by passenger, name
```

e. Total billing by airline and departure airport.

--displays total billing of an airline for each departure airport

```
select sum(price), a1.name, a2.code
from flight as f, airline as a1, airport as a2, ticket as t
where f.origin=a2.code and f.number=t.flight and a1.IATA_code=f.airline
group by a1.name, a2.code
```

10. Design an ER diagram that describes in detail the entities, attributes, relationships and keys to store the

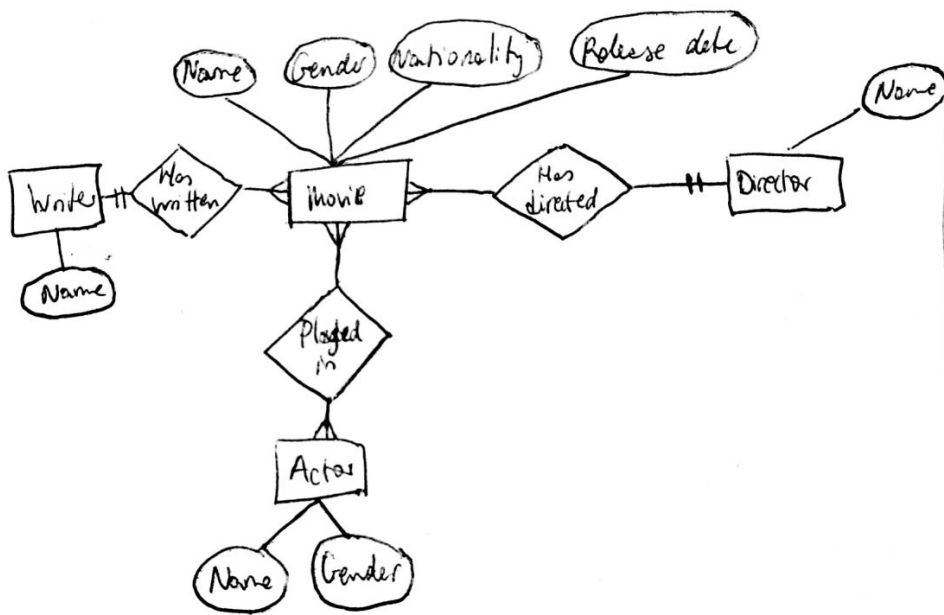
following information in a relational database:

- Movies with title, nationality, genre, cast, DIRECTOR, WRITER, release date.
- Actors, with name, gender, and movies they have played in (indicating the role).
- Directors, with name and directed movies.
- Writers, with name and movies the script of which they have written.

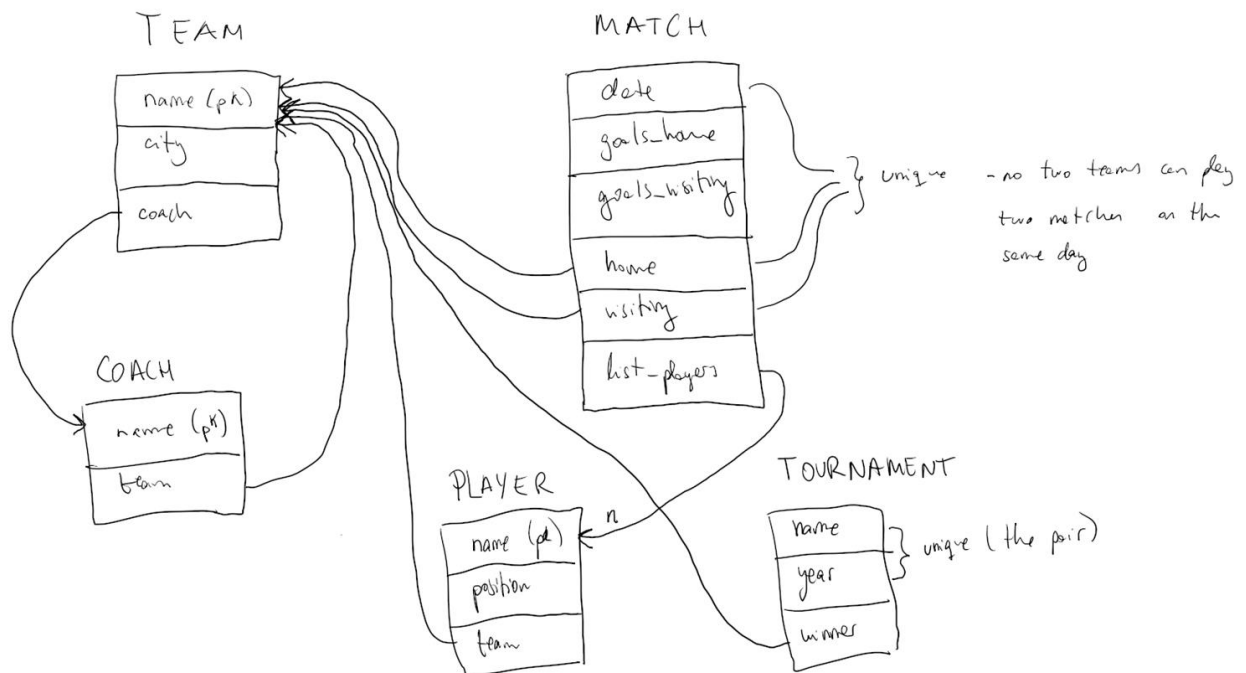
Indicate in detail the cardinality constraints, as well as the keys that the schemas may contain.

To simplify, consider initially:

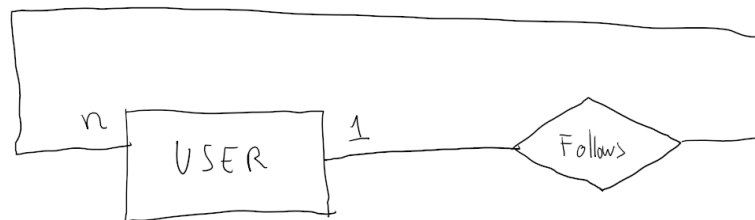
- That two movies cannot have the same title, and two people cannot have the same name.
- That no one can be an actor and director or writer at the same time, but can only one have of the three jobs.



16. The following ER model describes database structures to store information about teams, players and results of a football league season. To simplify, we shall assume that players and coaches do not switch between teams during the course of a season; and that two players or two coaches cannot have the same name.



18. Design an ER model for the structures of exercise 1.



19. Design an ER model for the structures of exercise 3.

