Lorenzo Scaturchio

21 October 2022

Homework 4

1. Explain what requirements this ER diagram is implementing. List the requirements (what the database is for, the main focus of the database, and explain entities and relationships) in plain sentences.

Requirements that are needed for this ER diagram are:

- Determine car vehicle owners and which vehicle(s) they own
- Searching for records of potential accidents in the past with associated report number
- Determining the damage amount based on the person and the type of vehicle they are driving
- There are a multitude of relationships that are at play for this database:
    - One person can own many cars
    - One car could be associated with multiple accidents
    - One person can have multiple accidents per multiple cars
    - A driver and a car could both simultaneously participate in an accident

2. Suppose you are given the following requirements for a simple database for Twitter messages:

   • there are many Twitter accounts (account entity) and many messages (message entity)

   • each account has an account ID, name, location, account description, and can

have many received messages

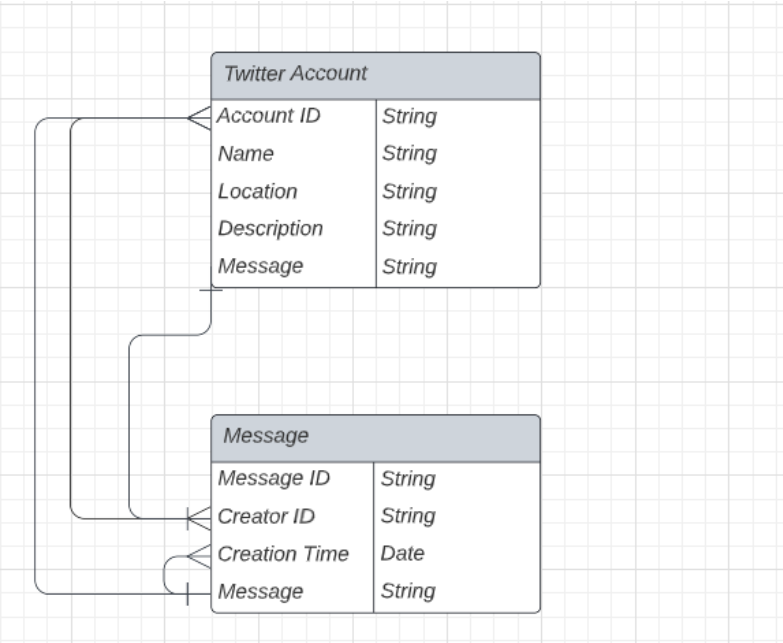
• each message has a message ID, creator account ID, creation time, textual message

• one message belongs to only one original creator account

• one message can be delivered to multiple accounts at different times

• one message can belong to multiple accounts (followers)

a. Construct a clean and concise ER diagram for this database



b. Create tables to implement your database design

```
CREATE TABLE Account (
    AccountID varchar(255),
    Name varchar(255),
    Location varchar(255),
    Description varchar(255),
```

```
    Message varchar(255)
    PRIMARY KEY (AccountID)
    FOREIGN KEY (Message) REFERENCES Message(Message)
);

CREATE TABLE Message (
    MessageID varchar(255),
    CreatorID varchar(255),
    Creation_time varchar(255),
    Message varchar(255)
    PRIMARY KEY (Message)
    FOREIGN KEY (CreatorID) REFERENCES Account(AccountID)
);
```

c. Explain how to search the name(s) of account holder(s) who is (are) the receiver(s) of a specific message in your database

In order to search for the name(s) of the account holder(s), you have to create a query that would select for the entity that contains an attribute, in this case the Twitter account entity searching for the name attribute. In order to determine which account is associated with a specific message, we would have to join the two entities that contain the message and match the message attribute with the string that contains the content of the message. If the contents match from the Twitter

account message (and is not associated with the Creator ID) attribute then that confirms who would be the receiver.

3. Using the following tables in the movie database, write SQL queries:

    i. Actor (id, fname, lname, age, gender, nationality)

    ii. Movie (id, name, year, rank, revenue, studio)

    iii. Director (id, fname, lname, gender, income)

b. List all the male actors (i.e., gender = ‘M’)

```
SELECT a.gender
FROM Actor a
WHERE a.gender = ‘M’
```

c. Find the actor whose first name is ‘John’ and from ‘Spain’.

```
SELECT a.fname
FROM Actor a
WHERE a.fname = ‘John’
AND f.nationality = ‘Spain’
```

d. List first name and last name of all the actors whose nationality is ‘USA’.

```
SELECT a.fname, a.lname
FROM Actor a
WHERE a.nationality = ‘USA’
```

e. List the name and revenue of movies made by “Universal Studio”.

```
SELECT m.name, m.revenue
FROM Movie m
```

WHERE m.studio = ‘Universal Studio’

f. List the total number of movies released in 2018.

SELECT COUNT(m.id)

FROM Movie m

WHERE m.year = 2018

g. List max income of all directors.

SELECT MAX(d.income)

FROM Director d

h. List the name of movies in descending order of revenue, i.e., the highest first.

SELECT m.name, m.revenue

FROM Movie m

ORDER BY m.revenue DESC

i. List the average income of female directors.

SELECT AVG(d.income)

FROM Directors d

WHERE d.gender = ‘F’