Lesson:

Types of Relationships

PW SKILLS

List of content :

1. Importance of Relationships in RDBMS.
2. Types of Relationships.

In the previous lecture, we had a detailed discussion on what is a relationship data model and its features. In this lecture, let's elaborate our discussion on the importance of relationships in RDBMS and look into the types of relationships in depth along with practical examples.

Importance of Relationships in RDBMS.

The concept of relationships in RDBMS is inspired by daily life human relationships. If you ask a group of people, what's their view on relationships, is it good or bad? Some say relationships are good and some say bad based on their experience. Along with this there also exists a group of people who answer diplomatically telling relationships can sometimes be good or bad, it varies from person to person.

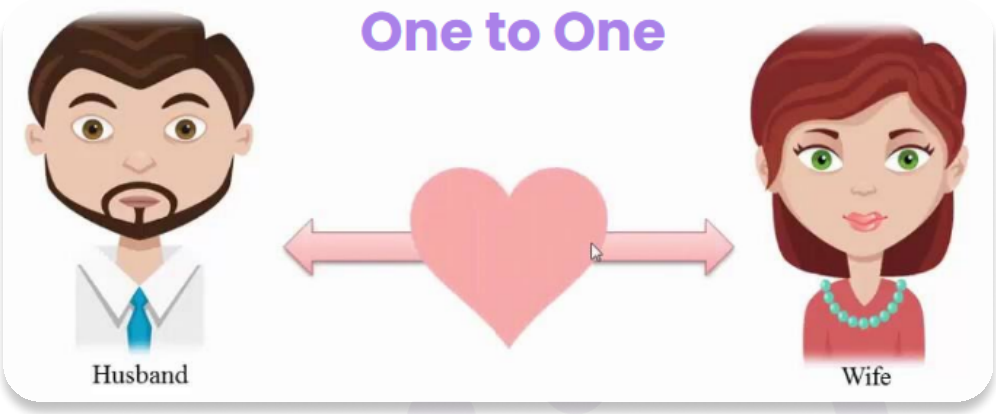
This question is quite controversial, and it is very hard to arrive at a conclusion. Most of the time, we agree upon the fact that relationships can sometimes be good or bad, and they varied person to person. This might not be true for all.

In the same way, if we ask "Are relationships important" anyone they surely will tell "Yes, relationships are important". They might be good or bad, but for humans, relationships play a very important role. In the same way in the relationship data model which is inspired by real-life relationships is important.

Types of Relationships.

Let's understand the types of relationships in RDBMS connecting to real-life relationships.

The first type of relationship that we have heard of is the relationships between people like romeo-juliet, and laila-majnu. These are some relationships that are often used to describe one love for life condition. Where an individual will be in a relationship only with the other one. We can also call this type of relationship a one-to-one relationship.

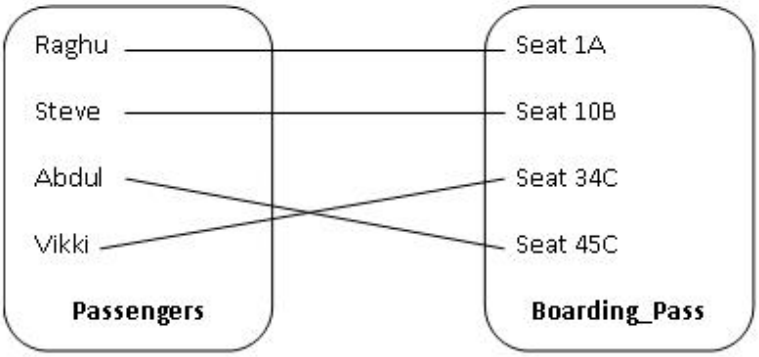


In the same way in RDBMS we have this concept of a one-to-one relationship where among the two tables, each row of one table has exactly one relationship with only row data of the other table and vice versa. We can also put this as the relationship, each record in one table is uniquely associated with only one record in another table, and vice versa.

Let's look at an example to understand better.

Globally we know that the ideal case is that each citizen in a country will be issued one passport. If this data is stored using an RDBMS it will look something like this. There will be two tables namely "Citizen" and "Passport". The relationship between the "Citizen" and "Passport" tables is established in such a way that every citizen in the "Citizen" table will have a unique passport number in the "Passport" table, and one passport in the "Passport" table will be associated with only one citizen in the "Citizen" table. This type of relationship is called a one-to-one relationship.

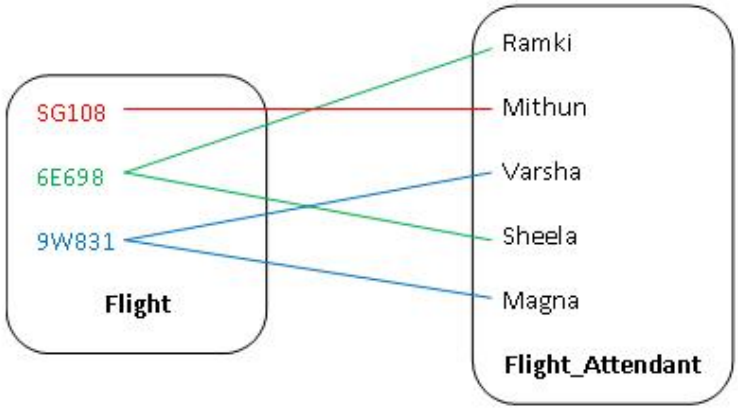
In the same way, we can also have another example where each passenger will be provided with one boarding pass. The relationship can be visualized like this.



This type of relationship where each record in one table is uniquely associated with only one record in another table, and vice versa is called a one-to-one relationship.
Moving further we have a one-to-many relationship. In this type of relationship, one record from a table is associated with multiple records in another table.

For example, a country can have multiple residents. Here in the country table, each record is associated with multiple records in the resident's table. This type of relationship is called as a one-to-many relationship.

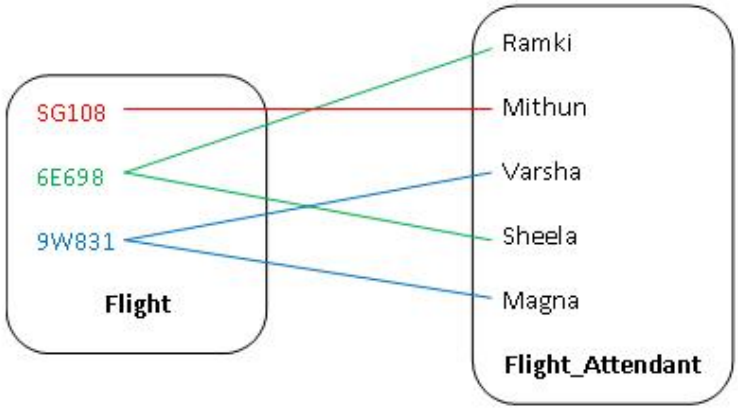
Another example to visualize the relationship is between flight and flight attendants. Each flight can have multiple flight attendants.



In the same way as the one-to-many relationship, we have another relationship type similar to this. In the one-to-many relationship typically a record in Table 1 will be associated with multiple records in Table 2. Similar to this we have another type of relationship where multiple records in Table 2 will be associated with only one record in Table 1. This type of relationship is called as the many-to-one relationship.

We can see this in the above-mentioned example too. Each country has multiple residents, this is a one-to-many relationship. In the same way, we will be having multiple residents living in one country, this is a many-to-one relationship.

Another example is between flights and flight_attendants. Each flight can have multiple attendants, this is a one-to-many relationship and multiple flight attendants will be on a single flight, this is a many-to-one relationship.

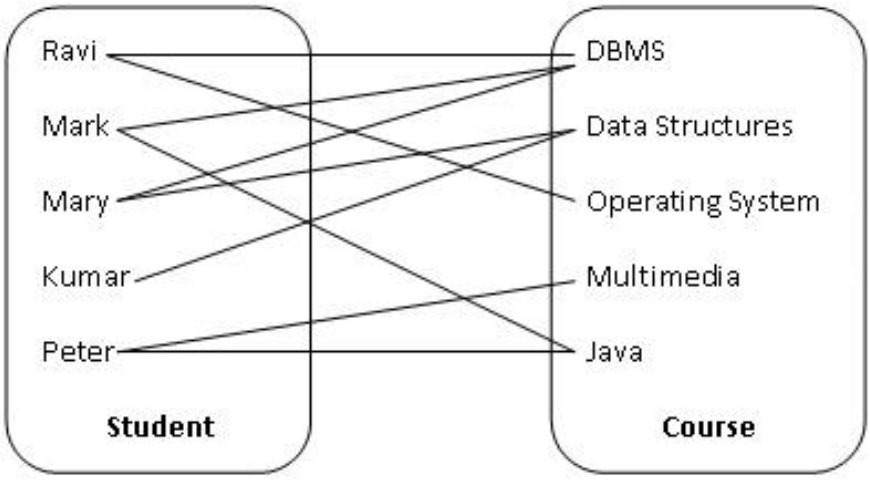


Both one-to-many and many-to-one are the same, only the difference is in which perspective we see them They are mirror images of each other.

After looking at the one-to-one relationship, one-to-many relationships, and many-to-one relationships, we have one last type of relationship, which is the many-to-many relationship.

In this type of relationship, records in each table are associated with multiple records in another table and vice versa.

The best example to understand this relationship is the relationship between the students and the courses.



As we can see in the above image, each student can be a part of multiple courses and each course can have multiple students. This is an example of a many-to-many relationship.

We can also see this type of relationship in applications like Tinder. Let's assume on Tinder there exist two entities males and females. Each male can select multiple females and each female can also select multiple males. This is also an example of a many-to-many relationship.