Islamic University of Technology (IUT)

Report on Lab 06

Submitted By

Shanta Maria (ID:200042172)

CSE 4308 Database Management Systems Lab

Submitted To

Md. Bakhtiar Hasan

Lecturer, Department of CSE

Zannatun Naim Srsity

Lecturer, Department of CSE

October 19, 2022

Introduction

In the lab class, we were asked to draw an ER Diagram by analysing a given problem set.

1 Task

Bhalo Basha Chai (BBC) is a housing agent in Bangladesh that publishes advertisements of properties that can be rented. Previously they stored all their information on paper. Recently they have decided to use a database. They have come up with the following requirements:

- There are many branches of BBC throughout the country. Each branch is located in a street of a city and has a postcode.

- Every branch is maintained by many employees. Upon joining the company, they provide their first name, last name, gender, and date of birth. They are also appointed a position (like manager, salesperson, etc.) in a specific branch. Their salaries are recorded for tax purposes.

- Numerous clients rent houses from BBC. They register by going to a certain branch and providing their first name, last name, telephone number, email, preferred accommodation type, and the maximum amount of rent they can afford. At that time, s/he is also assigned a staff member who is their contact person. A client can register in multiple branches.

- BBC stores information about the property owners who actually own the houses. The owners register by providing their first name, last name, telephone number, email, and password.

- BBC has multiple houses for rent under them. These houses are denoted by street, city, postcode, type, number of available rooms, and rent. Each property is associated with one owner, one contact person who is also a staff member, and the branch the staff works in.

- Each client can visit properties multiple times, but not twice in a day. A client can make some comments about the property during their visit. The date of their visit is also documented.

Now, your task is to:

1. Draw an ER Diagram, without any data redundancy, specifying the cardinality explicitly. You may add additional attributes only if it is needed.

2. Convert the ER Diagram into DDL using standard SQL denoting the appropriate constraints.

1.1 Solution

```sql
DROP TABLE propertyVisit;

DROP TABLE houseOwningInfo;

DROP TABLE owners;

DROP TABLE client;

DROP TABLE house;

DROP TABLE employee;

DROP TABLE branch;


CREATE TABLE branch
(
    branch_name VARCHAR(25) NOT NULL,

    street_name VARCHAR(25),

    city_name VARCHAR(25),

    postcode INT,

    CONSTRAINT pk_branch_name PRIMARY KEY(branch_name)
);
```

3

```sql
CREATE TABLE employee
(
    emp_id INT NOT NULL,

    first_name VARCHAR(25),

    last_name VARCHAR(25),

    gender VARCHAR(25),

    date_of_birth DATE,

    position VARCHAR(25),

    salary INT,

    branch_name VARCHAR(25) NOT NULL,

    CONSTRAINT pk_emp_id PRIMARY KEY(emp_id),

    CONSTRAINT fk_branch_name FOREIGN KEY(branch_name)

    REFERENCES branch(branch_name)
);


CREATE TABLE house
(
    house_id INT NOT NULL,

    street_name VARCHAR(25),

    city_name VARCHAR(25),

    postcode INT,

    type VARCHAR(25),

    num_of_rooms INT,

    rent NUMBER NOT NULL,

    CONSTRAINT pk_house_id_primary PRIMARY KEY(house_id)
);


CREATE TABLE client
```

```sql
(
    c_id INT NOT NULL,

    first_name VARCHAR(25),

    last_name VARCHAR(25),

    telephone_num NUMBER,

    email VARCHAR(25),

    accomodation_type VARCHAR(25),

    branch_name VARCHAR(25) NOT NULL,

    max_rent NUMBER NOT NULL,

    contact_person_id INT NOT NULL,

    house_id INT NOT NULL,

    CONSTRAINT pk_c_id PRIMARY KEY(c_id),

    CONSTRAINT fk_branch_name FOREIGN KEY(branch_name)

    REFERENCES branch(branch_name),

    CONSTRAINT fk_house_id FOREIGN KEY(house_id)

    REFERENCES house(house_id),

    CONSTRAINT fk_contact_person_id FOREIGN KEY(contact_person_id)

    REFERENCES employee(emp_id)
);

CREATE TABLE owners
(
    owner_id INT NOT NULL,

    first_name VARCHAR(25),

    last_name VARCHAR(25),

    telephone_num NUMBER,

    email VARCHAR(25),

    password VARCHAR(25),

    house_id INT NOT NULL,
```

```sql
    CONSTRAINT pk_owner_id PRIMARY KEY(owner_id),

    CONSTRAINT fk_house_id FOREIGN KEY(house_id)

    REFERENCES house(house_id)

);


CREATE TABLE houseOwningInfo

(

    house_id INT NOT NULL,

    owner_id INT NOT NULL,

    emp_id INT NOT NULL,

    CONSTRAINT pk_house_owning PRIMARY KEY(house_id, owner_id, emp_id),

    CONSTRAINT fk_house_id FOREIGN KEY(house_id)

    REFERENCES house(house_id),

    CONSTRAINT fk_owner_id FOREIGN KEY(owner_id)

    REFERENCES owners(owner_id),

    CONSTRAINT fk_emp_id FOREIGN KEY(emp_id)

    REFERENCES employee(emp_id)

);


CREATE TABLE propertyVisit

(

    house_id INT NOT NULL,

    date_of_visit DATE NOT NULL,

    comments VARCHAR(255),

    CONSTRAINT pk_property_visit PRIMARY KEY(house_id, date_of_visit),

    CONSTRAINT fk_house_id FOREIGN KEY(house_id)

    REFERENCES houseOwningInfo(house_id)

);
```
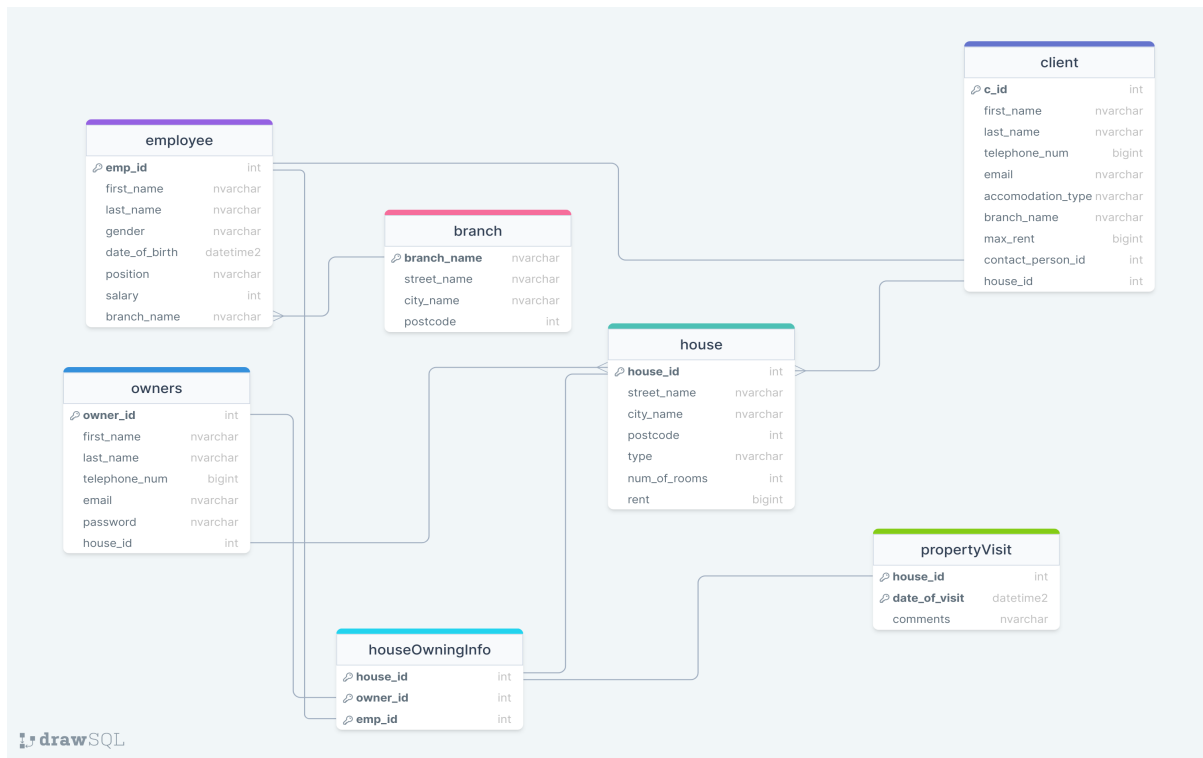
Figure 1: ER Diagram

1.2 Analysis and Explanation

For the solution, I have created 7 entities and linked them. First, there is the `branch` table. Here `branch_name` is the primary key because each branch is unique to a city.// Next, we have the `employee` table where `emp_id` is the primary key and the relationship between `branch` and `employee` is one to many since one branch can have many employees and they are linked by the foreign key `branch_name`.

Then, we have the `client` table where the primary key is `c_id` and the `house` table. The `client` table is linked with both the `employee` and `house` tables using the foreign keys `contact_person_id` and `house_id` respectively since every client is assigned a contact person who is a member of the staff in that particular branch. There is a one to one relationship between `employee` and `client` while a one to many relationship between `client` and `house` since one client can rent several houses from a branch.

Lastly, we have the `owners` table with primary key `owner_id`, `houseOwningInfo` table with 3 attributes as primary key and propertyVisit| with `house_id` and `date_of_visit` as primary key. There is a one to one relationship between `propertyVisit` and `houseOwningInfo` linked

by foreign key `house_id`, `owners` and `houseOwningInfo` tables using `owner_id` as foreign key and the tables `houseOwningInfo` and `employee` using foreign key `emp_id`.

1.3 Difficulties

I faced difficulties in figuring out how each entity is connected to one another and what the relationship is between them. No other mentionable issues were encountered.

Conclusion

As shown in the report, I have solved and tested the solutions for all four tasks given in the lab. All the commands used were written in notepad which was then saved with .sql extension. The .sql file was then run through the `SQL` command line to execute all the commands after executing the given `DDL+drop.sql` and `smallRelationsInsertFile.sql` files.