

Database Organization - CS425

Project Phase 2

Group - 7

Translating E-R Model To Relational Schema

Translating Strong Entities:

Property(Property_ID, Type, Description, Availability, Price)

Renter(Renters_ID, User_ID, Move_In_Date, PreferredLocation, Budget)

PropertyBooking(Booking_ID, Property_ID, Renters_ID, Booking_Date, CreditCard_ID)

CreditCard(CreditCard_ID, CardNumber, Renters_ID, Address, CardExpiryDate)

Agents(Agents_ID, User_ID, RealEstateAgency, Contact, Job_Title)

House(House_ID, Property_ID, Rooms, Sqfoot_Area)

Apartment(Apts_ID, Property_ID, Rooms, Sqfoot_Area)

CommercialBuilding(Building_ID, Property_ID, BusinessType, Sqfoot_Area)

Users(User_ID, First_Name, Middle_Name, Last_Name, User_Type, Address, Email_ID)

Translating Multivalued Composite Attributes:

Property(Property_ID, Type, Description, Availability, Price)

Property_Address(Property_ID, City, State, Location)

Renter(Renters_ID, User_ID, Move_In_Date, PreferredLocation, Budget)

PropertyBooking(Booking_ID, Property_ID, Renters_ID, Booking_Date, CreditCard_ID)

CreditCard(CreditCard_ID, CardNumber, Renters_ID, Address, CardExpiryDate)
Agents(Agents_ID, User_ID, RealEstateAgency, Contact, Job_Title)
House(House_ID, Property_ID, Rooms, Sqfoot_Area)
Apartment(Apts_ID, Property_ID, Rooms, Sqfoot_Area)
CommercialBuilding(Building_ID, Property_ID, BusinessType, Sqfoot_Area)
Users(User_ID, First_Name, Middle_Name, Last_Name, User_Type, Address, Email_ID)

Translating One to One Relationships:

Property(Property_ID, Type, Description, Availability, Price)
Property_Address(Property_ID, City, State, Location)
Renter(Renters_ID, User_ID, Move_In_Date, PreferredLocation, Budget)
PropertyBooking(Booking_ID, Property_ID, Renters_ID, Booking_Date, CreditCard_ID)
CreditCard(CreditCard_ID, CardNumber, Renters_ID, Address, CardExpiryDate)
Agents(Agents_ID, User_ID, RealEstateAgency, Contact, Job_Title)
House(House_ID, Property_ID, Rooms, Sqfoot_Area)
Apartment(Apts_ID, Property_ID, Rooms, Sqfoot_Area)
CommercialBuilding(Building_ID, Property_ID, BusinessType, Sqfoot_Area)
Users(User_ID, First_Name, Middle_Name, Last_Name, User_Type, Address, Email_ID)
PropertyCredit(Booking_ID, Property_ID, CreditCard_ID)

Translating One to Many/ Many to One Relationships:

Property(Property_ID, Type, Description, Availability, Price)
Property_Address(Property_ID, City, State, Location)
Renter(Renters_ID, User_ID, Move_In_Date, PreferredLocation, Budget)
PropertyBooking(Booking_ID, Property_ID, Renters_ID, Booking_Date, CreditCard_ID)

CreditCard(CreditCard_ID, CardNumber, Renters_ID, Address, CardExpiryDate)

Agents(Agents_ID, User_ID, RealEstateAgency, Contact, Job_Title, Booking_ID, Property_ID , Renters_ID)

House(House_ID, Property_ID, Rooms, Sqfoot_Area)

Apartment(Apts_ID, Property_ID, Rooms, Sqfoot_Area)

CommercialBuilding(Building_ID, Property_ID, BusinessType, Sqfoot_Area)

Users(User_ID, First_Name, Middle_Name, Last_Name, User_Type, Address, Email_ID)

PropertyCredit(Booking_ID, Property_ID, CreditCard_ID)

Translating Many to Many Relationships:

Property(Property_ID, Type, Description, Availability, Price)

Property_Address(Property_ID, City, State, Location)

Renter(Renters_ID, User_ID, Move_In_Date, PreferredLocation, Budget)

PropertyBooking(Booking_ID, Property_ID, Renters_ID, Booking_Date, CreditCard_ID)

CreditCard(CreditCard_ID, CardNumber, Renters_ID, Address, CardExpiryDate)

Agents(Agents_ID, User_ID, RealEstateAgency, Contact, Job_Title, Booking_ID, Property_ID , Renters_ID)

House(House_ID, Property_ID, Rooms, Sqfoot_Area)

Apartment(Apts_ID, Property_ID, Rooms, Sqfoot_Area)

CommercialBuilding(Building_ID, Property_ID, BusinessType, Sqfoot_Area)

Users(User_ID, First_Name, Middle_Name, Last_Name, User_Type, Address, Email_ID)

PropertyCredit(Booking_ID, Property_ID, CreditCard_ID)

User_Renter(Renters_ID, User_ID)

User_Agent(Agents_ID, User_ID)

Searches(Renters_ID, Property_ID)

PropertyHouse(Property_ID, House_ID)

PropertyApts(Property_ID, Apts_ID)

PropertyBuild(Property_ID, Building_ID)

The Final Schema:

Property(Property_ID, Type, Description, Availability, Price)

Property_Address(Property_ID, City, State, Location)

Renter(Renters_ID, User_ID, Move_In_Date, PreferredLocation, Budget)

PropertyBooking(Booking_ID, Property_ID, Renters_ID, Booking_Date, CreditCard_ID)

CreditCard(CreditCard_ID, CardNumber, Renter_ID, Address, CardExpiryDate)

Agents(Agents_ID, User_ID, RealEstateAgency, Contact, Job_Title, Booking_ID, Property_ID , Renters_ID)

House(House_ID, Property_ID, Rooms, Sqfoot_Area)

Apartment(Apts_ID, Property_ID, Rooms, Sqfoot_Area)

CommercialBuilding(Building_ID, Property_ID, BusinessType, Sqfoot_Area)

Users(User_ID, First_Name, Middle_Name, Last_Name, User_Type, Address, Email_ID)

PropertyCredit(Booking_ID, Property_ID, CreditCard_ID)

User_Renter(Renters_ID, User_ID)

User_Agent(Agents_ID, User_ID)

Searches(Renters_ID, Property_ID)

PropertyHouse(Property_ID, House_ID)

PropertyApts(Property_ID, Apts_ID)

PropertyBuild(Property_ID, Building_ID)

Schema In PostgreSQL

```
-- drop table property cascade;
-- drop table property_address cascade;
-- drop table renter cascade;
-- drop table property_booking cascade;
-- drop table credit_card cascade;
-- drop table agents cascade;
-- drop table house cascade;
-- drop table apartment cascade;
-- drop table commercial_building cascade;
-- drop table users cascade;
-- drop table property_credit cascade;
-- drop table user_renter cascade;
-- drop table user_agent cascade;
-- drop table searches cascade;
-- drop table property_house cascade;
-- drop table property_apts cascade;
-- drop table property_build cascade;
```

```
create table property(
property_id varchar(8) not null,
type varchar(20),
description varchar(200),
availability varchar(20),
price numeric(10, 2),
primary key(property_id)
);
```

```
create table property_address(
property_id varchar(8) not null,
```

```
city varchar(30) not null,  
state varchar(20) not null,  
location decimal(9, 6) not null,  
primary key(property_id, city, state, location),  
foreign key(property_id) references property,  
);
```

```
create table users(  
user_id varchar(8) not null,  
first_name varchar(10),  
middle_name varchar(10),  
last_name varchar(10),  
user_type varchar(10) not null,  
address varchar(30),  
email_id varchar(50) not null,  
primary key(user_id)  
);
```

```
create table renter(  
renters_id varchar(8) not null,  
user_id varchar(8) not null,  
move_in_date DATE,  
preferred_location decimal(9, 6),  
budget numeric(10, 2),  
primary key(renters_id),  
foreign key(user_id) references users  
);
```

```
create table credit_card(  
credit_card_id varchar(8) not null,  
card_number integer(10) not null,  
renters_id varchar(8) not null,  
address varchar(30) not null,  
card_expiry_date DATE not null,  
primary key(credit_card_id),  
foreign key(renters_id) references renter  
);
```

```
create table property_booking(  
booking_id varchar(8) not null,
```

```
property_id varchar(8) not null,  
renters_id varchar(8) not null,  
booking_date DATE not null,  
credit_card_id varchar(8),  
primary key(booking_id, property_id, renters_id),  
foreign key(property_id) references property,  
foreign key(renters_id) references renter,  
foreign key(credit_card_id) references credit_card  
);
```

```
create table agents(  
agents_id varchar(8) not null,  
user_id varchar(8) not null,  
real_estate_agency varchar(50),  
contact integer(10),  
job_title varchar(30),  
booking_id varchar(8) not null,  
property_id varchar(8) not null,  
renters_id varchar(8) not null,  
primary key(agents_id),  
foreign key(user_id) references users,  
foreign key(booking_id) references property_booking,  
foreign key(property_id) references property,  
foreign key(renters_id) references renter  
);
```

```
create table house(  
house_id varchar(8) not null,  
property_id varchar(8) not null,  
rooms integer(5),  
sqfoot_area numeric(6, 2),  
primary key(house_id),  
foreign key(property_id) references property,  
);
```

```
create table apartment(  
apts_id varchar(8) not null,  
property_id varchar(8) not null,  
rooms integer(5),
```

```
sqfoot_area numeric(6, 2),  
primary key(apts_id),  
foreign key(property_id) references property,  
);
```

```
create table commercial_building(  
building_id varchar(8) not null,  
property_id varchar(8) not null,  
business_type varchar(30),  
sqfoot_area numeric(6, 2),  
primary key(building_id),  
foreign key(property_id) references property,  
);
```

```
create table property_credit(  
booking_id varchar(8) not null,  
property_id varchar(8) not null,  
credit_card_id varchar(8) not null,  
primary key(booking_id, property_id),  
foreign key(booking_id) references property_booking,  
foreign key(property_id) references property,  
foreign key(credit_card_id) references credit_card  
);
```

```
create table user_renter(  
renters_id varchar(8) not null,  
user_id varchar(8) not null,  
primary key(renters_id, user_id),  
foreign key(renters_id) references renter,  
foreign key(user_id) references users  
);
```

```
create table user_agent(  
agents_id varchar(8) not null,  
user_id varchar(8) not null,  
primary key(agents_id, user_id),  
foreign key(agents_id) references agents,  
foreign key(user_id) references users  
);
```



```
create table searches(  
renters_id varchar(8) not null,  
property_id varchar(8) not null,  
primary key(renters_id, property_id),  
foreign key(property_id) references property,  
foreign key(renters_id) references renter  
);
```

```
create table property_house(  
house_id varchar(8) not null,  
property_id varchar(8) not null,  
primary key(house_id, property_id),  
foreign key(property_id) references property,  
foreign key(house_id) references house  
);
```

```
create table property_apts(  
apts_id varchar(8) not null,  
property_id varchar(8) not null,  
primary key(apts_id, property_id),  
foreign key(property_id) references property,  
foreign key(apts_id) references apartment  
);
```

```
create table property_build(  
building_id varchar(8) not null,  
property_id varchar(8) not null,  
primary key(building_id, property_id),  
foreign key(property_id) references property,  
foreign key(building_id) references commercial_building  
);
```