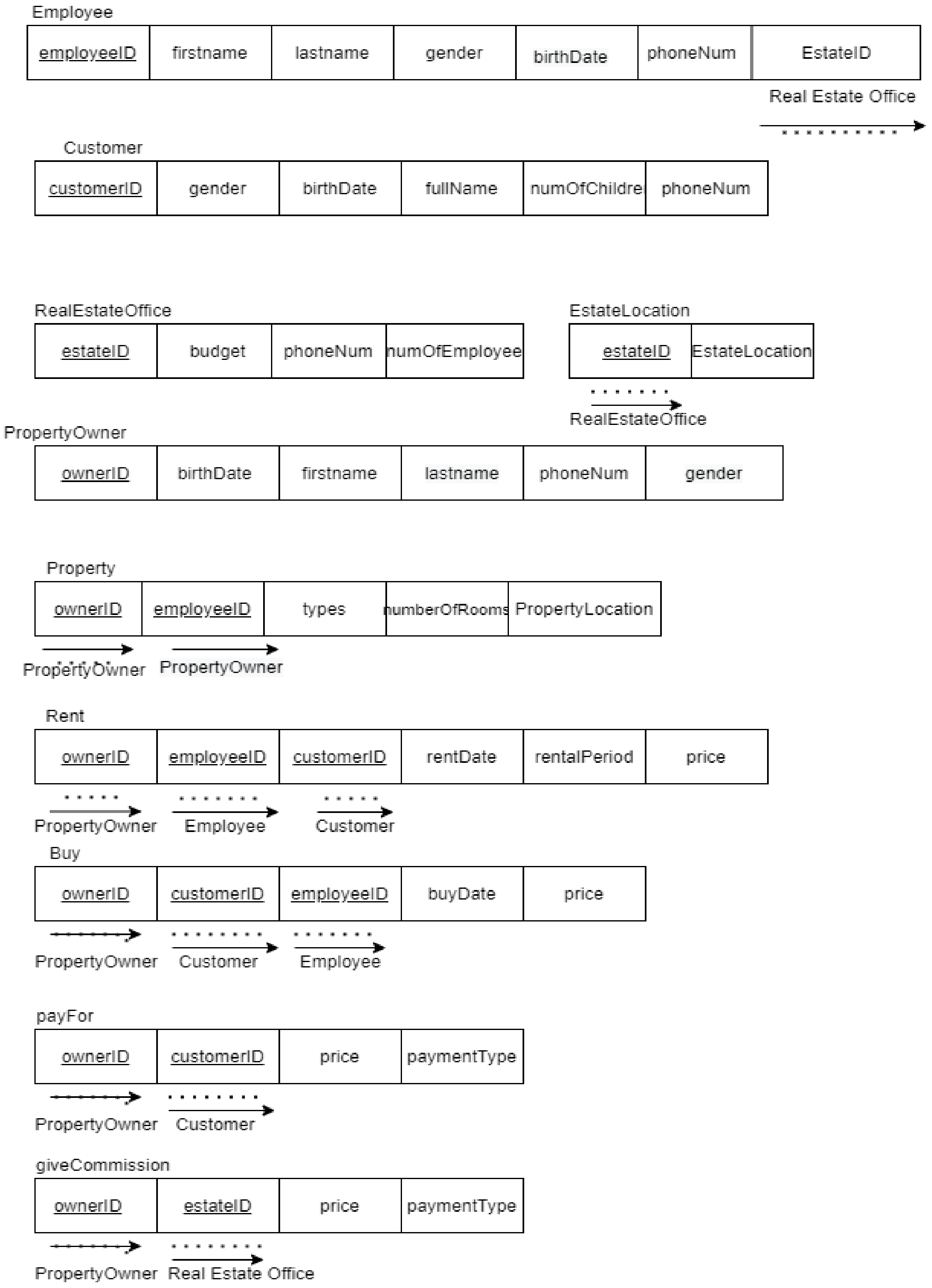
UMAY ŞAMLI	**52255193822**
SELİN DÖKMEN	**49159828400**
BEYBİN ÇİÇEK	**32933258128**

FINAL SCHEMA OF THE DIAGRAMS



Employee

employeeID	firstname	lastname	gender	birthDate	phoneNum	EstateID
						Real Estate Office →

Customer

customerID	gender	birthDate	fullName	numOfChildre	phoneNum

RealEstateOffice

estateID	budget	phoneNum	numOfEmployee

EstateLocation

estateID	EstateLocation
→ RealEstateOffice	

PropertyOwner

ownerID	birthDate	firstname	lastname	phoneNum	gender

Property

ownerID	employeeID	types	numberOfRooms	PropertyLocation
→ PropertyOwner	→ PropertyOwner			

Rent

ownerID	employeeID	customerID	rentDate	rentalPeriod	price
→ PropertyOwner	→ Employee	→ Customer			

Buy

ownerID	customerID	employeeID	buyDate	price
→ PropertyOwner	→ Customer	→ Employee		

payFor

ownerID	customerID	price	paymentType
→ PropertyOwner	→ Customer		

giveCommission

ownerID	estateID	price	paymentType
→ PropertyOwner	→ Real Estate Office		

Explanation of tables in the Database

As seen in the relational schema above, our database is consist of 10 tables, and 1 Views

RealEstateOffice table: This table keeps records of real estate office, each office can be identified with a unique estate ID. This ID serves as a primary key which other tables can use to refer to our real estate office table.

Employee Table: The employee table keeps records of employees who are customers in a office. This table is identified by the employee ID and the ID of the estate .Each employee is uniquely identified by an employee id. Other information such as gender and birth date are stored in this table.

Estate Location Table: This table keeps track of all locations in our system. A location record include a reference to the office to which the rent/buy is made, a reference to the,customer, and estate are also stored. Each estate record is uniquely identified by a estate number.

Property Owner Table: This table keeps records of all properties done against a owner. It also stores the a reference to the customer who made the payment, it also keeps stores the phone number and gender type.

Property Table: This table store all the estate types in our database. Each estate has a unique ID number and also a estate to how many room belongs to.

Customer table: This table keeps records of customers, each customer can be identified with a unique ID. This ID serves as a primary key which other tables can use to refer to our customer table.

Rent Table: This table keeps records of the available estates in our database, each estate has a unique rent date.

Buy Table: This table stores records about which estate when buyed or rented, the owner ID and the employee ID are used to reference owner and employee tables respectively. They also both serve as primary key to this table.

Pay For table: This table stores records of payment type.

Give CommissionTable: The table keeps records of the comisssions given to each estate office by the owner depending on the payment type.

Property Info Views Table: Since the views are multivalued attribute, a separate table is created and stores the views which are then linked to the room numbers.

UPDATES FROM PHASE 1

We put “on delete cascade” to the tables. This key word allows us to drop table that gives reference to other tables. With this keyword, we are allowed to drop tables that we want.

Explanation of the Java Application Program

We wrote a java program that basically interacts with the user by a simple command line. Outputs depend on the user’s inputs. This app basically does insert, update, delete and list operations on the database.

Our program contains just one class (main) and we manage all operations with this class. This class communicates with the database by using jdbc driver. In the beginning part of our code, we connect with the database.

You can enter numbers from 0 to 8 in the program. If you try to enter a different input, the program will ask to enter a different number.

->Entering 0 to program: If you enter zero to program, you will basically exit the program.

->Entering 1 to the program: By entering 1 to the program, you can see the active estate offices that work. To list active estates, program executes a query. That query brings us "realestateOffice" and "estateLocation" tables that we will use. To bring the requested data from both, we use joint operation. Than program searches and pulls the data from the information which we pulled from database via query. After that, we print the information about estate offices.

->Entering 2 to the program: By entering 2 to the program, you can print the houses in the database, by defining the district. Also, you can see the owners of the houses and the estate agent who is in charge. To do that, firstly program executes a query. That query brings us the needed data from"property", "employee" and "propertyowner" tables that we will use. To bring the needed information from all fields, we use joint operations twice. Than program searches and pulls the information from the information that we pulled from database via query. Then program prints the information about the estate offices.

->Entering 3 to the program: By entering 3 to the program, you can see all the workers personal information who work in the estate offices. Also, you can see where s/he is working and the phone number of the office. To do that, firstly program executes a query. That query brings us the needed data from "EstateLocation", "employee" and "realestateoffice" tables that we will use. To bring the needed data's from all, we use joint operations twice. Than program searches and pulls the information from the data that we pulled from database via query. Then program prints the information about the estate offices.

->Entering 4 to the program: By entering 4 to the program, you can see all the customer's personal information. To do that, firstly program executes a query. That query brings us the needed data from"Customer" table that we will use. To bring the needed data's from the table, we use joint operations twice. Than program searches and pulls the required data from the data's that we pulled from database via query. Then program prints the information about the customers.

->Entering 5 to the program: By entering 5 to the program, you can see all the Property owner's personal information. To do that, firstly program executes a query. That query brings us "PropertyOwner" tables that we will use. To extract the needed data from the table, we use joint operations twice. Than program searches and pulls the required data from the data that we pulled from database via query. Then program prints the information about estate offices.

->Entering 6 to the program: By entering 6 to the program, you can insert a customer to the database. Firstly, program has all the required information about the customer to insert the customer to the data base. Then program prints all the list of customer's before inserting operation. After that program inserts the customer, program prints all the list after inserting operation. To do these operations, we execute the query to find all customers data. Than program searches and pulls the required information from the data that we pulled from database via query. After that program lists all customer before insertion operation, then the program inserts the customer. Than program lists all the customers again. But this time program lists customers after the insertion operation.

->Entering 7 to the program: By entering 7 to the program, you can update the phone number of the specific property owner. Firstly, program requests the ID of the property owner. Than program wants from you to enter the new phone number. Than program prints the old data about the property owner, and then the program prints the new data. To do these operations, we execute a statement to find the specific property owner's data with that ID. Than program searches and pulls the required data from the data's that we pulled from database via statement. Then, program prints the specific property owner before the update operation. Then the program updates the phone number of that property owner. Finally the program prints the property owner's updated version of the data.

->Entering 8 to the program: By entering 8 to the program, you can delete a specific employee. Firstly, program requests the ID of the employee. After you enter the ID, the program searches and finds the employee with that specific ID. Then the program deletes this employee's data from the database. To do this operation, we execute a statement to find the employee that has the specific ID. Than program searches and pulls the required data from the info that we pull from the database via statement. Then program deletes the data in the database about that employee.