

# Real Estate Database Management System

Nicole Pang  
916558934  
Github Username: npng16

Milestone/Version	Date
M1	06/07/20
Revised M1	06/25/20
M2	07/16/20

## Table of Contents

Section I: Project Description.....	2
Section II: Use Cases .....	3
Section III: Database Requirements (Business Rules) .....	4
Section IV: Detailed List of Main Entities, Attributes and Keys .....	8
Section V: Entity Relationship Diagram (ERD).....	12
Section VI: Testing Table .....	16
Section VII: Database Model .....	24
Section VIII: Forward Engineering .....	27
Section IX: Inserting Data.....	28
Section X: Testing.....	29
Section XI: Testing Table .....	30

## Section I: Project Description

My motivation to create this real estate database management system is to help organize large amounts of client and property information for company and client use. I want businesses and real estate agents to be able to access needed information efficiently and with ease. This database is going to help businesses create more sales and build a strong client base. They will be able to market themselves and their properties successfully which will increase revenue over time. Clients are very important is a business's success. Companies should be able to reach out to former and current clients more efficiently through mail, email, or telephone which will help in keeping a strong client base. Having a strong client base can help with sales since it is easier to make sales through existing clients rather than new clients. Companies should also be able to effectively market current and upcoming properties to current/former/prospective clients. This database should assist prospective and former clients in easily accessing information about available and upcoming properties offered by the company. Clients will be able to access information regarding real estate agents at a company such as: their experience, credentials, and contact information. My real estate database management system will help companies increase sales and maintain a strong client base by providing convenient and effortless access of information to the companies as well as clients.

## Section II: Use Cases

1. Jordan is a real estate agent at a company who wants to inform his former and current clients of a property that was just listed for sale. He logs into the system with his username and password, then pulls up his past transactions with former clients. Each transaction lists the client's name, phone number, and email so he can contact them. He goes down his list of clients contacting them, one by one. When the client asks the agent questions, the agent can provide quick and informative answers by pulling up the listing on his computer.
2. Ann is searching for available homes in her area through Company X's website. She notices that if she creates an account on the website, she can favorite certain property listings and can revisit them later. Ann chooses to create an account on Company X's website, but it is asking what area she is looking into. She isn't able to continue registering for an account without specifying where she wants to look for properties. She continues registering for an account after confirming the desired area and entering her full name, email, password, and phone number.
3. Belle, a prospective client comes into the Company X office and is greeted by an agent. Belle expresses that she is interested in seeing a property listing she previously viewed on their company website. They both converse and the agent asks about her preferences such as: the property type, property size, desired neighborhoods or area, and her budget. The agent tells Belle that she needs to have a down payment ready, proof of the down-payment, and pre-approval letter if she plans on loaning money from a financial institution or she will not be able to view the property in person. Belle does not have all the necessary documentation, so the agent shows her the property. The agent and Belle exchange contact information. He tells Belle to make an appointment with him when she is ready to view properties. After she leaves, he starts to compile a list of available properties based on her preferences and plans to present his list at their next meeting.
4. A company has a new property listing available for purchase. David, a real estate agent at the company advertises this new listing on the company website. He also reaches out to clients through telephone and email about this new listing and gives them a date for chance to view the property in person. David holds the open house and many people stop by, both current and prospective clients. Even realtors from other companies stop by the open house to collect property information for their clients. He exchanges contact information with the prospective clients and other agents so he can reach out to them after the open house ends.
5. A client goes to a company to sell her property. The agent working there asks the client what kind of property is and the client states that it is residential property she inherited from her parents. The client tells the agent that her property has already been appraised and gives the agent all the details of the house. The details include when the house was built, the number of bedrooms and bathrooms, the total number of square feet of the property, and any additional amenities. The realtor logs into the computer and posts a new property listing on his company's website for the client.

## Section III: Database Requirements (Business Rules)

1. Company:
  1. A company shall have one name.
  2. A company shall operate at least one website.
  3. A company shall work with at least one employee.
  4. A company shall have one unique id number.
2. Employee:
  1. An employee shall work for only one company.
  2. An employee shall have only one unique id number.
  3. An employee shall have one first name.
  4. An employee shall have one last name.
  5. An employee shall have many clients.
  6. An employee shall host at least one open house.
  7. An employee shall be able to post multiple property listings.
  8. An employee shall manage at least one property.
3. Website:
  1. A website shall have at least one account.
  2. A website shall be accessed by multiple users at once.
  3. A website shall have only one unique id.
  4. A website shall be operated by only one company.
  5. A website shall have at least one property listing.
  6. A website shall have many sessions at once.
4. User:
  1. A user shall be able to view at least one property listing.
  2. A user shall be able to create only one account.
  3. A user shall have one first name.
  4. A user shall have one last name.
  5. A user shall have one unique id number.
  6. A user shall be able to have only one session at a time.
  7. A registered user shall be able to favorite multiple property listings at a time.
  8. A user shall be able search using one search engine at a time.
  9. A user shall access many websites at once.
5. Session:
  1. A session shall belong to only one user.
  2. A session shall expire after a period of time.
  3. A session shall belong to only one website.
6. Account:
  1. An account shall belong to only one user.
  2. An account shall create only one profile.
  3. An account shall have one unique id.
  4. An account shall have at least one email address.
  5. An account shall belong to only one website.
  6. An account shall belong to only one client.
  7. An account shall be able to create many lists.

## 7. Property:

1. A property shall be owned by only one landlord at a time.
2. A property shall be managed by one and only one employee.
3. A property shall have only one property listing.
4. A property shall be assigned at least one property value.
5. A property shall belong to only one neighborhood.
6. A property shall have at least one open house.
7. A property shall be viewed by many clients.
8. A property shall be sold by only one landlord.
9. A property shall be bought by only one landlord.

## 8. Landlord:

1. A landlord shall have only one unique id.
2. A landlord shall own at least one property.
3. A landlord shall be able to sell at least one property.
4. A landlord shall be able to buy at least one property.
5. A landlord shall own at least one piece of land.
6. A landlord shall own at least one store.
7. A landlord shall own at least one apartment.
8. A landlord shall own at least one house.
9. A landlord shall own at least one unit.
10. A landlord shall own at least one room.

## 9. Client:

1. A client shall be assigned to only one employee at a time.
2. A client shall be able to view one property at a time.
3. A client shall be able create at most one account.
4. A client shall have only one unique id.
5. A client shall take out at most one loan.
6. A client shall have at least one bank account.
7. A client shall make at least one payment.

## 10. Property Listing:

1. A property listing shall be posted on only one website.
2. A property listing shall have only one state at a time.
3. A property listing shall have only one unique id.
4. A property listing shall be viewed by multiple users at once.
5. A property listing shall belong to many lists.
6. A property listing belongs to only one property.
7. A property listing shall be posted by at least one employee.
8. A property listing shall be favorited multiple users.

## 11. Payment:

1. A payment shall have one unique payment id.
2. A payment shall have only one payment type.
3. A payment shall have a payment date.
4. A payment shall be made by only one client.

## 12. Payment Type:

1. A payment type shall belong to only one payment.
  2. A payment type is credit or cash.
  3. A payment type shall have one unique payment type id.
  4. A payment type shall have one billing address.
  5. A payment type shall have one city.
  6. A payment type shall have one unique state.
  7. A payment type shall have one zip code.
  8. A payment type shall have one country.
13. Loan:
1. A loan shall belong to only one client.
  2. A loan shall have one loan amount.
  3. A loan shall have one unique loan id.
  4. A loan shall be granted by only one bank.
14. Bank:
1. A bank shall grant multiple loans at once.
  2. A bank shall have at least one client.
  3. A bank shall contain many bank accounts.
15. Neighborhood:
1. A neighborhood shall have at least one house.
  2. A neighborhood shall have at least one apartment.
  3. A neighborhood shall have at least one store.
  4. A neighborhood shall have at least one piece of land.
16. Profile:
1. A profile shall be created by only one account.
  2. A profile shall have only one unique profile id.
17. List:
1. A list shall contain at least one property listing.
  2. A list shall be created by only one account.
  3. A list shall be private or public.
18. Bank Account:
1. A bank account shall have one unique id.
  2. A bank account shall belong to only one client at once.
  3. A bank account shall belong to only one bank.
19. Search Engine
1. A search engine shall be used by many users at one.
  2. A search engine shall have one unique address id.
  3. A search engine shall have one area.
  4. A search engine shall have one city.
  5. A search engine shall have one state.
  6. A search engine shall have one zip code.
  7. A search engine shall have one country.
20. House
1. A house shall be owned by only one landlord.
  2. A house shall belong to only one neighborhood.

3. A house shall have many rooms.
21. Apartment
  1. An apartment shall be owned by only one landlord.
  2. An apartment shall belong to only one neighborhood.
  3. An apartment shall have many units.
22. Unit
  1. A unit shall be owned by only one landlord.
  2. A unit shall belong to only one apartment.
23. Store
  1. A store shall be owned by only one landlord.
  2. A store shall belong to only one neighborhood.
24. Land
  1. A piece of land shall be owned by only one landlord.
  2. A piece of land shall belong to only one neighborhood.
25. Room
  1. A room shall be owned by only one landlord.
  2. A room shall belong to only one house.
26. Open House:
  1. An open house shall be hosted by only one employee.



## Section IV: Detailed List of Main Entities, Attributes and Keys

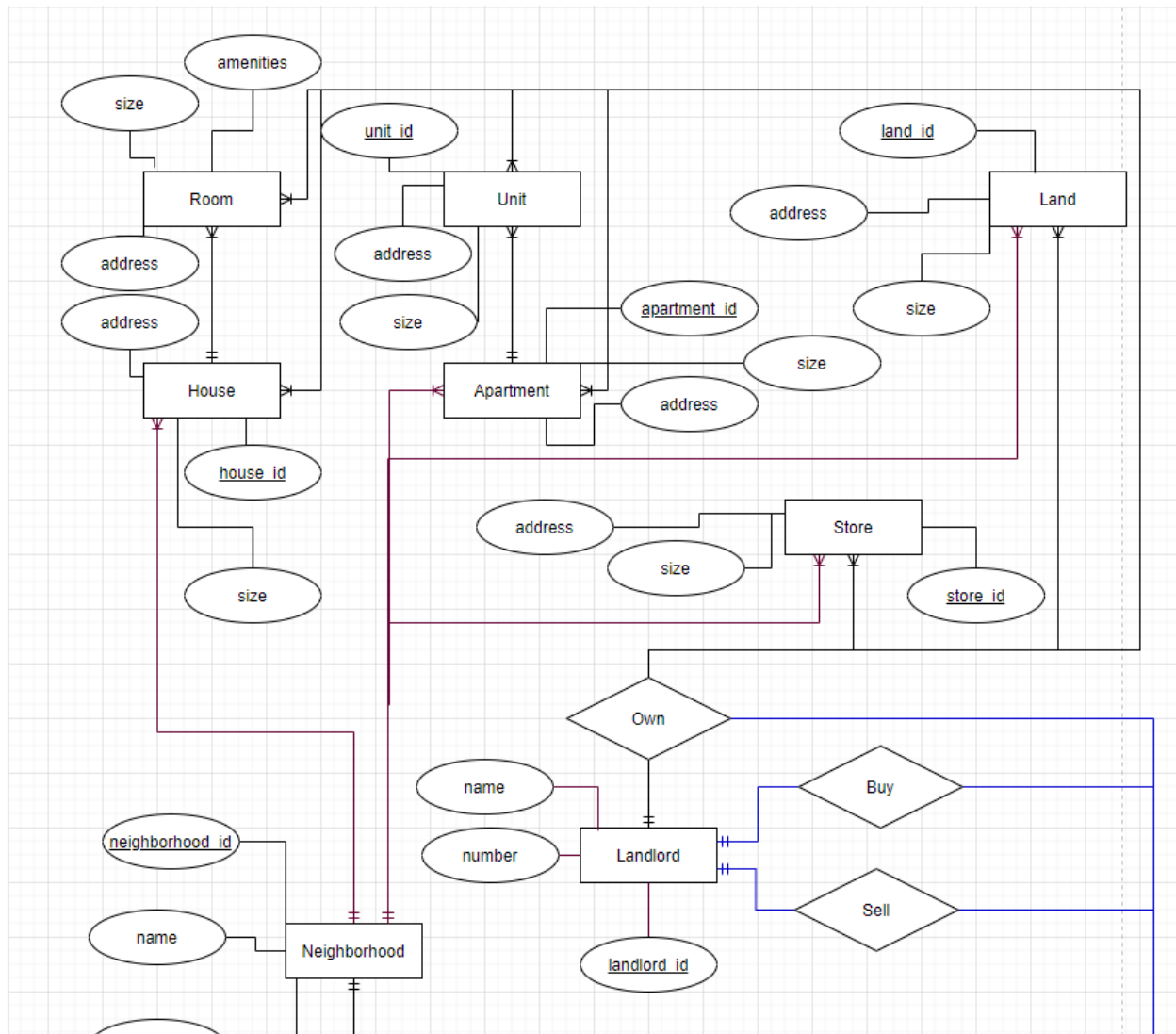
1. Company: Strong
  1. Company\_id: key, numeric
  2. name: alphanumeric
  3. description: alphanumeric
  4. Num\_employees: numeric
  5. Num\_clients: numeric
2. Employee: Strong
  1. Employee\_id: key, numeric
  2. email: key, alphanumeric
  3. name: alphanumeric
  4. number: numeric
  5. extension: numeric
3. Website: Strong
  1. Website\_id: key, numeric
  2. Num\_sessions: numeric
  3. Num\_accounts: numeric
  4. Num\_users: numeric
4. User: Strong
  1. User\_id: key, numeric
  2. name: alphanumeric
  3. Last\_name: alphanumeric
  4. Email: key, alphanumeric
5. Session: Strong
  1. Session\_id: key, numeric
  2. User: alphanumeric
  3. Duration: alphanumeric
6. Account: Strong
  1. Account\_id: key, numeric
  2. Email: key, alphanumeric
  3. Password: alphanumeric
  4. Created: alphanumeric
7. Property: Strong
  1. property\_id: key, numeric
  2. property\_type: alphanumeric
  3. address: alphanumeric, multi-value, composite
    1. street
    2. zipcode
    3. state
    4. country
  4. property\_sq\_ft: alphanumeric
8. Landlord: Weak
  1. landlord\_id: key, numeric
  2. name: alphanumeric

3. number: numeric
  4. email: alphanumeric
  5. address: alphanumeric
9. Client: Strong
  1. client\_id: key, numeric
  2. name: alphanumeric
  3. number: numeric
  4. email: alphanumeric
  5. address: alphanumeric
10. PropertyListing: Strong
  1. PropertyListing\_id: key, alphanumeric
  2. date: alphanumeric
  3. price: alphanumeric
  4. description: alphanumeric
11. Payment: Strong
  1. Payment\_id: key, numeric
  2. Payment\_type: composite, multivalue
    1. Cash
    2. Credit
  3. date: alphanumeric
12. PaymentType: Weak
  1. Payment\_type\_id: key, numeric
  2. Billing\_address: alphanumeric
  3. Billing\_city: alphanumeric
  4. Billing\_state: alphanumeric
  5. Billing\_zipcode: numeric
  6. Billing\_county: alphanumeric
13. Loan: Strong
  1. Loan\_id: key, numeric
  2. Loan\_approval: alphanumeric
  3. Borrower\_name: alphanumeric
  4. amount: alphanumeric
  5. term: alphanumeric
  6. interest\_rate: alphanumeric
  7. balance: alphanumeric
14. Bank: Strong
  1. Bank\_id: key, numeric
  2. name: alphanumeric
  3. Loan\_amount: numeric
  4. Loan\_date: alphanumeric
  5. Num\_clients: numeric
15. Neighborhood: Strong
  1. Neighborhood\_id: key, alphanumeric
  2. name: alphanumeric

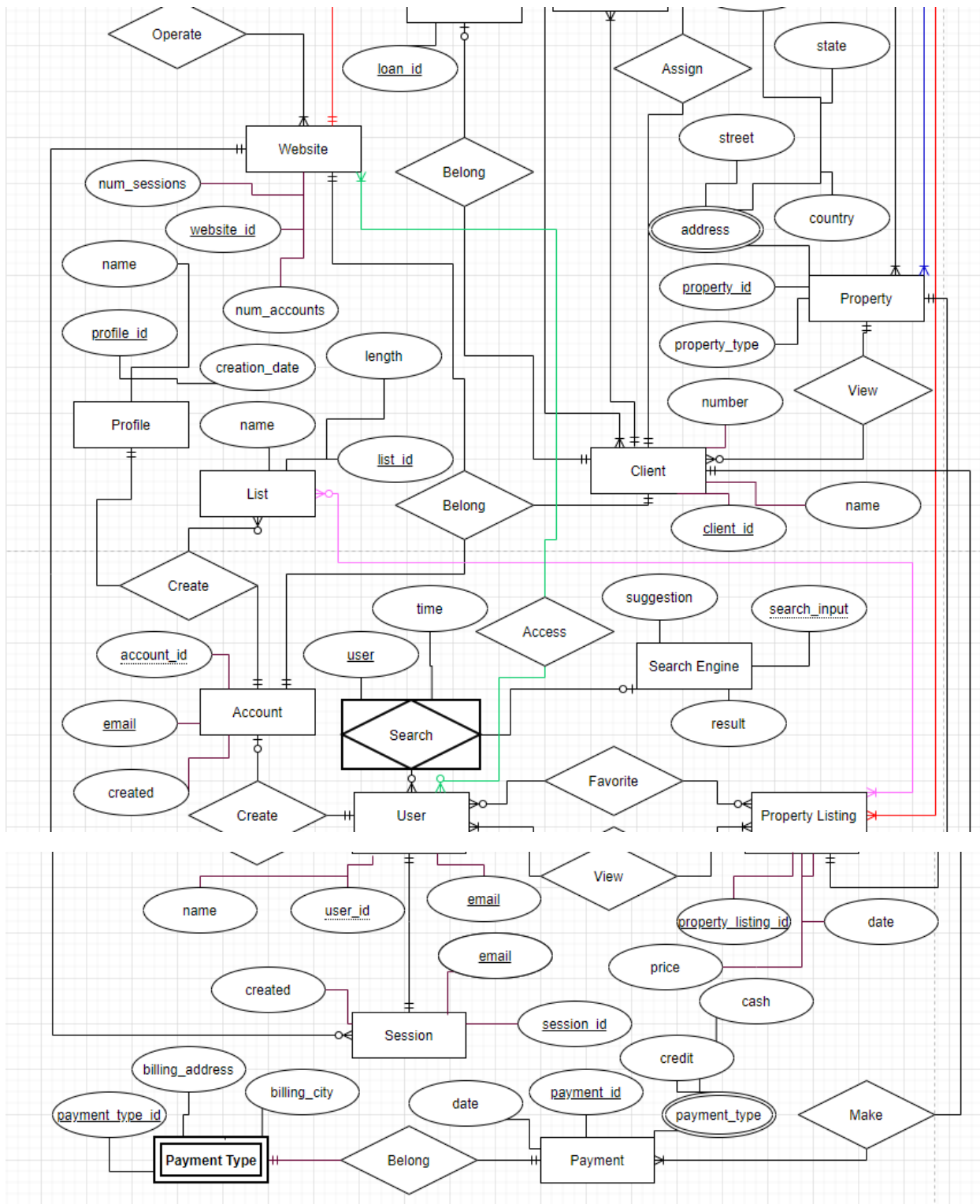
3. size: numeric
4. population: numeric
16. Profile: Strong
  1. Profile\_id: key, numeric
  2. name: alphanumeric
  3. creation\_date: alphanumeric
17. List: Strong
  1. List\_id: key, numeric
  2. name: alphanumeric
  3. length: numeric
  4. creation\_date: alphanumeric
18. BankAccount: Strong
  1. Account\_id: key, alphanumeric
  2. name: alphanumeric
  3. routing: numeric
19. SearchEngine: Strong
  1. SearchEngine\_id: key, numeric
  2. Result: alphanumeric
  3. input: alphanumeric
20. House: Strong
  1. House\_id: key, numeric
  2. Address: alphanumeric
  3. Size: alphanumeric
  4. num\_rooms: numeric
21. Apartment: Strong
  1. Apartment\_id: key, numeric
  2. Address: alphanumeric
  3. Size: alphanumeric
  4. Num\_units: numeric
  5. Num\_rooms: numeric
  6. Amenities: alphanumeric
22. Unit: Strong
  1. Unit\_id: key, numeric
  2. Address: alphanumeric
  3. Size: alphanumeric
  4. num\_rooms: numeric
  5. amenities: alphanumeric
23. Store: Strong
  1. Store\_id: key, alphanumeric
  2. Address: alphanumeric
  3. Size: alphanumeric
24. Land: Strong
  1. Land\_id: key, numeric
  2. Address: alphanumeric

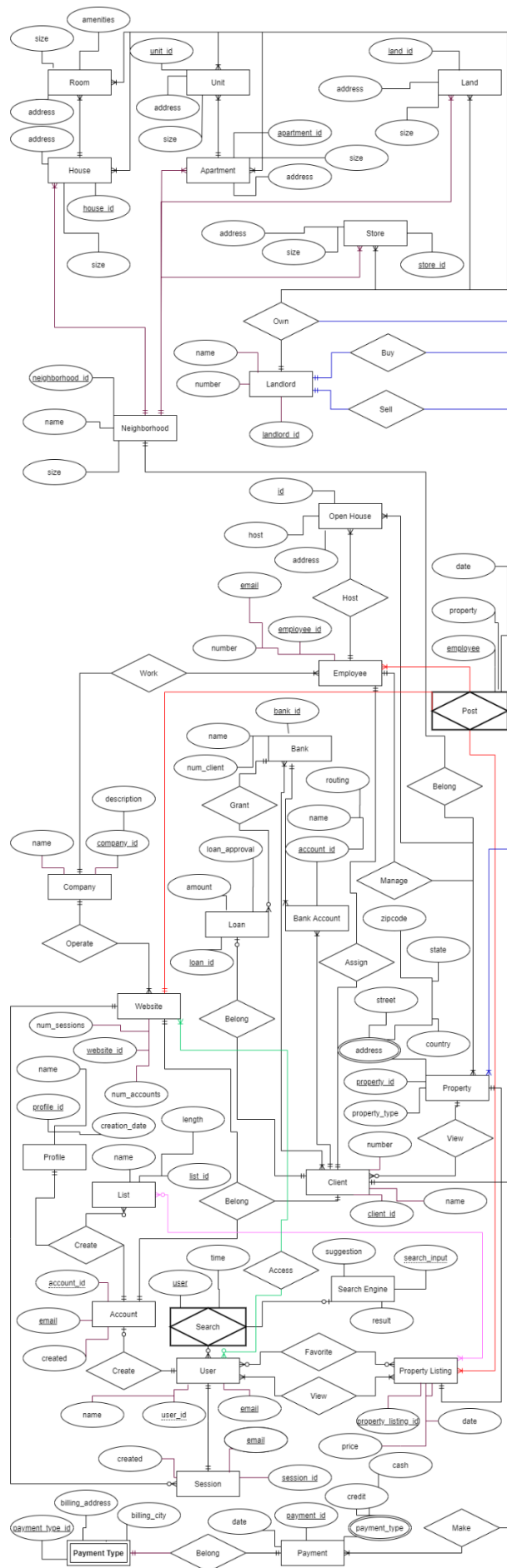
3. Size: alphanumeric
25. Room: Strong
  1. Room\_id: key, numeric
  2. Size: alphanumeric
  3. Amenities: alphanumeric
26. OpenHouse: Strong
  1. Id: key, numeric
  2. Host: alphanumeric
  3. Address: alphanumeric
  4. Date: alphanumeric
  5. Duration: alphanumeric
27. Post: Weak
  1. employee: key, alphanumeric
  2. property: alphanumeric
  3. date: alphanumeric
28. Search: Weak
  1. user: key, alphanumeric
  2. listing: alphanumeric
  3. time: numeric

## Section V: Entity Relationship Diagram (ERD)











## Section VI: Testing Table

Rule	Entity A	Relation	Entity B	Cardinality	Pass/Fail	Error Description
1	User	Create	Account	1:1	Fail	A user shall be able to create only one account. Changed from one and one only to zero or one. (ERD)
2	User	View	Property Listing	1:M	Fail	A user shall be able to view at least one property listing. Changed from many to one or many. (ERD)
3	User	Have	Session	1:1	Fail	A user shall be able to have only one session at a time. Changed from one to one and only one. (ERD)
4	User	Favorite	Property Listing	1:M	Fail	A user shall be able to favorite multiple property listings at a time. Changed from many to zero or many. (ERD)
5	User	Search	Search Engine	1:1	Fail	A user shall be able to search using one search engine at a time. Changed from one to zero or one. (ERD)
6	User	Access	Website	1:1	Fail	A user shall access many websites at once. Changed from one to zero or many.
7	Account	Create	User	1:1	Fail	An account shall belong to only one user. Changed from zero or one to one and only one. (ERD)
8	Account	Create	List	1:M	Fail	An account shall be able to create many lists. Changed from many to zero and many. (ERD)
9	Account	Create	Profile	1:1	Fail	An account shall create only one profile. Changed from one to one and only one. (ERD)
10	Account	Belong	Website	1:1	Fail	An account shall belong to only one website. Changed from one to one and only one. (ERD)

11	Account	Belong	Client	1:1	Fail	An account shall belong to only one client. Changed from one to one and only one. (ERD)
12	Session	Belong	User	1:1	Fail	A session shall belong to only one user. Changed from one to one and only one. (ERD)
13	Client	Have	Bank	1:M	Fail	A client shall belong to at least one bank. Changed from one to one or many. (ERD).
14	Client	Assign	Employee	1:1	Fail	A client shall be assigned to only one employee at a time. Changed from one to one and only one. (ERD)
15	Client	View	Property	1:1	Pass	A client shall be able to view one property at a time.
16	Client	Belong	Account	1:1	Fail	A client shall create at most one account. Changed from one to zero or one. (ERD)
17	Client	Belong	Loan	1:1	Fail	A client shall be able to take out at most one loan.
18	Client	Have	Bank Account	1:M	Fail	A client shall have at least one bank account. Changed from many to one or many. (ERD)
19	Client	Make	Payment	1:M	Fail	e payment. Changed from zero or many to one or many.
20	Session	Have	Website	1:1	Fail	A session shall belong to only one website. Changed from one to one and only one. (ERD)
21	List	Create	Account	1:1	Fail	A list shall be created by only account. Changed from one to one and only one. (ERD)
22	Profile	Create	Account	1:1	Fail	A profile shall be created by only one account. Changed from one to one and only one (ERD)

23	Website	Operate	Company	1:1	Fail	A website shall be operated by only one company. Changed from one to one and only one. (ERD)
24	Website	Have	Account	1:M	Fail	A website shall have at least one account. Changed from many to one or many. (ERD)
25	Website	Access	User	1:M	Fail	A website shall be accessed by many users at once. Changed from many to one or many. (ERD)
26	Website	Have	Session	1:M	Fail	A website shall have many sessions at once. Changed from many to zero or many.
27	Company	Operate	Website	1:M	Fail	A company shall operate at least one website. Changed from many to one or many. (ERD)
28	Company	Work	Employee	1:M	Fail	A company shall work with at least one employee. Changed from zero or many to one or many. (ERD)
29	Employee	Work	Company	1:1	Fail	An employee shall work for only one company. Changed from zero or one to one and only one. (ERD)
30	Employee	Manage	Property	1:M	Fail	An employee shall manage at least one property. Changed from zero or many to one or many. (ERD)
31	Property	Manage	Employee	1:1	Fail	A property shall be managed by one and only one employee. Changed from one to one and only one. (ERD)
32	Property	Have	Property Listing	1:1	Pass	A property shall have only one listing.
33	Property	Belong	Neighborhood	1:1	Fail	A property shall belong to only one neighborhood. Changed from one to one and only one. (ERD)

34	Property	Have	Open House	1:M	Fail	A property shall have at least one open house. Changed from one to one or many. (ERD)
35	Property	View	Client	1:M	Fail	A property shall be viewed by many clients. Changed from many to zero or many. (ERD)
36	Payment	Make	Client	1:1	Fail	A payment shall be made by only one client. Changed from zero or one to one and only one. (ERD)
37	Payment	Have	Payment Type	1:1	Pass	A payment shall have only one payment type.
38	Payment Type	Belong	Payment	1:1	Fail	A payment type shall belong to only one payment. Changed from one to one and only one. (ERD)
39	Loan	Belong	Client	1:1	Fail	A loan shall belong to only one client. Changed from one to one and only one. (ERD)
40	Loan	Grant	Bank	1:1	Pass	A loan shall be granted by only one bank.
41	Bank	Grant	Loan	1:M	Fail	A bank shall grant multiple loans at once. Changed from many to zero or many. (ERD)
42	Bank	Have	Client	1:M	Fail	A bank shall have at least one client. Changed from many to one or many. (ERD)
43	Bank	Contain	Bank Accounts	1:M	Fail	A bank shall contain at least one bank account. Changed from many to one or many. (ERD)
44	Bank Account	Belong	Bank	1:1	Pass	A bank account shall belong to only one bank.
45	Bank Account	Belong	Client	1:1	Fail	A bank account shall belong to only one client at once. Changed from one to one and only one. (ERD)
46	Search Engine	Search	User	1:M	Fail	A search engine shall be used by many users at once.

						Changed from many to zero or many. (ERD)
47	Employee	Host	Open House	1:M	Fail	An employee shall host at least one open house. Changed from zero or many to one or many. (ERD)
48	Open House	Host	Employee	1:1	Fail	An open house shall be hosted by only one employee. Changed from one to one and only one. (ERD)
49	Property Listing	Belong	List	1:M	Fail	A property listing shall belong to many lists. Changed from many to zero or many. (ERD)
50	List	Belong	Property Listing	1:1	Fail	A list shall contain at least one property listing. Changed from zero or many to one or many. (ERD)
51	Employee	Post	Property Listing	1:M	Pass	An employee shall be able to post multiple property listings.
52	Property Listing	Post	Employee	1:M	Pass	A property listing shall be posted by at least one employee.
53	Property Listing	Belong	Property	1:1	Fail	A property listing belongs to only one property. Changed from one to one and only one. (ERD)
54	Property Listing	Post	Website	1:1	Pass	A property listing shall be posted on the only one website.
55	Property Listing	Favorite	User	1:M	Fail	A property listing shall be favorited by multiple users. Changed from many to zero or many. (ERD)
56	Property Listing	View	User	1:M	Fail	A property listing shall be viewed by multiple users at once. Changed from many to one or many. (ERD)
57	Website	Post	Property Listing	1:M	Pass	A website shall have at least one property listing.

58	Landlord	Own	Property	1:M	Fail	A landlord shall own at least one property. Changed from many to one or many. (ERD)
59	Landlord	Sell	Property	1:M	Fail	A landlord shall be able to sell at least one property. Changed from many to one or many. (ERD)
60	Landlord	Buy	Property	1:M	Fail	A landlord shall be able to buy at least one property. Changed from many to one or many. (ERD)
61	Landlord	Own	House	1:M	Fail	A landlord shall own at least one house. Changed from many to one or many. (ERD)
62	Landlord	Own	Apartment	1:M	Fail	A landlord shall own at least one apartment. Changed from many to one or many. (ERD)
63	Landlord	Own	Unit	1:M	Fail	A landlord shall own at least one unit. Changed from many to one or many. (ERD)
64	Landlord	Own	Store	1:M	Fail	A landlord shall own at least one store. Changed from many to one or many. (ERD)
65	Landlord	Own	Land	1:M	Fail	A landlord shall own at least one piece of land. Changed from many to one or many. (ERD)
66	Landlord	Own	Room	1:M	Fail	A landlord shall own at least one room. Changed from many to one or many. (ERD)
67	Property	Own	Landlord	1:1	Fail	A property shall be owned by only one landlord at a time. Changed from one to one and only one. (ERD)
68	Property	Sell	Landlord	1:1	Fail	A property shall be sold by only one landlord. Changed from one to one and only one. (ERD)
69	Property	Buy	Landlord	1:1	Fail	A property shall be bought by only one landlord.

						Changed from one to one and only one. (ERD)
70	House	Own	Landlord	1:1	Fail	A house shall be owned by only one landlord. Changed from one to one and only one. (ERD)
71	House	Belong	Room	1:M	Pass	A house shall have many rooms.
72	House	Belong	Neighborhood	1:1	Pass	A house shall belong to only one neighborhood.
73	Apartment	Own	Landlord	1:1	Fail	An apartment shall be owned by only one landlord. Changed from one to one and only one. (ERD)
74	Apartment	Belong	Unit	1:M	Pass	An apartment shall have many units.
75	Apartment	Belong	Neighborhood	1:1	Fail	An apartment shall belong to only one neighborhood. Changed from one to one and only one. (ERD)
76	Unit	Own	Landlord	1:1	Fail	A unit shall be owned by only one landlord. Changed from one to one and only one. (ERD)
77	Unit	Belong	Apartment	1:1	Fail	A unit shall belong to only one apartment. Changed from one to one and only one. (ERD)
78	Store	Own	Landlord	1:1	Fail	A store shall be owned by only one landlord. Changed from one to one and only one. (ERD)
79	Store	Belong	Neighborhood	1:1	Fail	A store shall belong to only one neighborhood. Changed from one to one and only one. (ERD)
80	Land	Own	Landlord	1:1	Fail	A piece of land shall be owned by only one landlord. Changed from one to one and only one. (ERD)
81	Land	Belong	Neighborhood	1:1	Pass	A piece of land shall belong to only one neighborhood. (ERD)

82	Room	Own	Landlord	1:1	Fail	A room shall be own by only one landlord. Changed from one to one and only one. (ERD)
83	Room	Belong	House	1:1	Fail	A room shall belong to only one house. Changed from one to one and only one. (ERD)
84	Neighborhood	Belong	House	1:M	Pass	A neighborhood shall have at least one house.
85	Neighborhood	Belong	Apartment	1:M	Pass	A neighborhood shall have at least one apartment.
86	Neighborhood	Belong	Store	1:M	Fail	A neighborhood shall have at least one store. Changed from many to one or many. (ERD)
87	Neighborhood	Belong	Land	1:M	Fail	A neighborhood shall have at least one piece of land. Changed from many to one or many. (ERD)



## Section VII: Database Model

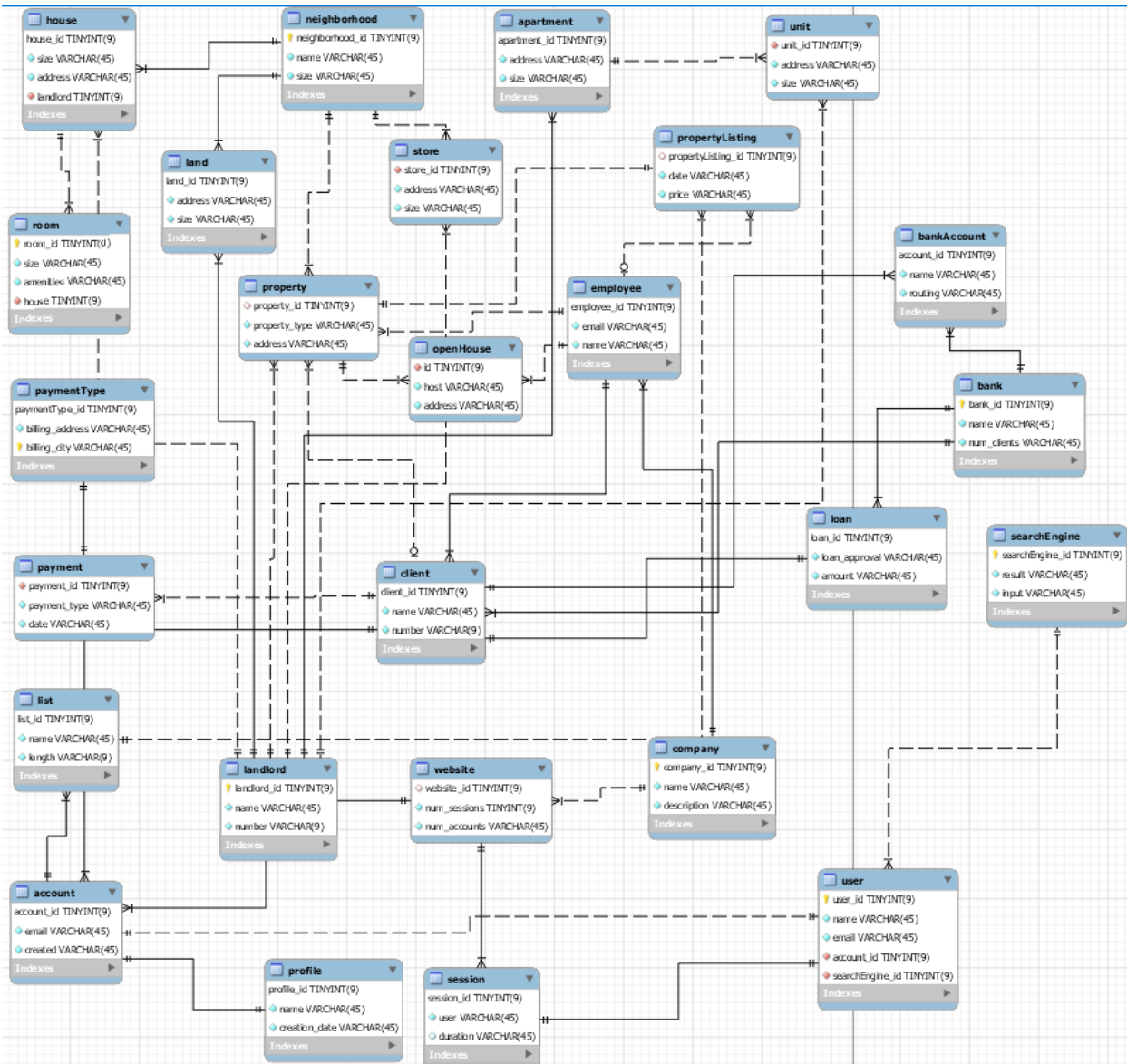


Table	Key	DELETE/UPDATE	Constraint	Reason
Profile	FK	Delete	Cascade	If the account is deleted, the profile associated with it, is also deleted.
List	FK	Delete	Cascade	If the account is deleted, the lists associated with the account is also deleted.
Employee	Fk	Delete	Cascade	If the company is deleted, then the employee information associated with the company is also deleted.
BankAccount	FK	Delete	Cascade	If the bank is deleted, then the bank account information associated with the bank is also deleted.
PropertListing	Fk	Delete	Cascade	If the property is deleted, then the property listing information associated with the property is also deleted.
Room	Fk	Delete	Cascade	1.If the house is deleted, then the room information associated with the house is also deleted. 2.If the landlord is deleted then the room information associated with it will also be deleted.
Unit	Fk	Delete	Cascade	1.If the apartment is deleted, then the unit information associated with the apartment is also deleted. 2.If the landlord is deleted, then the unit information associated with it will also be deleted.
Land	Fk	Delete	Cascade	1.If the neighborhood is deleted, the land information associated with it will also be deleted. 2.If the landlord is deleted, the land information associated with it will also be deleted.
Apartment	Fk	Delete	Cascade	1.If the neighborhood is deleted, the apartment information associated with it will also be deleted. 2.If the landlord is deleted, the apartment information associated with it will also be deleted.
House	Fk	Delete	Cascade	1.If the neighborhood is deleted, the house information associated with it will also be deleted. 2.If the landlord is deleted, the house information associated with it will also be deleted.

Store	Fk	Delete	Cascade	<p>1.If the neighborhood is deleted, the store information associated with it will also be deleted.</p> <p>2.If the landlord is deleted, the store information associated with it will also be deleted.</p>
OpenHouse	Fk	Delete	Cascade	<p>If the property is deleted, the open house information associated with it will also be deleted.</p> <p>If the employee is deleted, the open house information associated with it will also be deleted.</p>
PaymentType	Fk	Delete	Cascade	<p>If the payment is deleted, the payment type information associated with it will also be deleted.</p>
Loan	Fk	Delete	Cascade	<p>If the bank is deleted, the loan information associated with it will also be deleted.</p> <p>If the client is deleted, then the loan information associated with it will also be deleted.</p>
Session	Fk	Delete	Cascade	<p>If the website is deleted, the session information associated with it will also be deleted.</p> <p>If the user is deleted, the session information associated with it will also be deleted.</p>

## Section VIII: Forward Engineering

MySQL Script generated by MySQL Workbench  
 Thu Jul 16 11:04:47 2020  
 Model: New Model Version: 1.0  
 MySQL Workbench Forward Engineering

```

1  -- MySQL Script generated by MySQL Workbench
2  -- Thu Jul 16 11:04:47 2020
3  -- Model: New Model Version: 1.0
4  -- MySQL Workbench Forward Engineering
5
6  • SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;
7  • SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0;
8  • SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,NO_ZERO_DATE,ERROR_FOR_DIVISION_BY_ZERO,NO_ENGINE_SUBSTITUTION';
9
10 -----
11 -- Schema RealEstateDB
12 -----
13 • DROP DATABASE RealEstateDB;
14 • CREATE DATABASE IF NOT EXISTS RealEstateDB;
15 • USE RealEstateDB;
16 -----
17 -- Table `landlord`
18 -----
19 • DROP TABLE IF EXISTS `landlord` ;
20
21 • CREATE TABLE IF NOT EXISTS `landlord` (
22     `landlord_id` TINYINT(9) NOT NULL,
23     `name` VARCHAR(45) NOT NULL,
24     `number` VARCHAR(9) NOT NULL,
25     PRIMARY KEY (`landlord_id`))
26 ENGINE = InnoDB;
27
28 -----
29 -- Table `neighborhood`
30 -----
31 -----

```

Output

#	Time	Action	Message	Duration / Fetch
393	11:09:15	DROP TABLE IF EXISTS 'session'	0 row(s) affected, 1 warning(s): 1051 Unknown table 'realestatedb.session'	0.000 sec
394	11:09:15	CREATE TABLE IF NOT EXISTS 'session' ( 'session_id' TINYINT(9) NOT NULL, 'user' VARCHA...	0 row(s) affected, 1 warning(s): 1681 Integer display width is deprecated and will be removed in a futur...	0.078 sec
395	11:09:15	SET SQL_MODE=@OLD_SQL_MODE	0 row(s) affected, 1 warning(s): 3135 'NO_ZERO_DATE', 'NO_ZERO_IN_DATE' and 'ERROR_FOR_...	0.000 sec
396	11:09:15	SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS	0 row(s) affected	0.000 sec
397	11:09:15	SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS	0 row(s) affected	0.000 sec

## Section IX: Inserting Data

inserts x databasemodel

Limit to 1000 rows

```

51
52
53 • IES (1, 43, 161), (2, 17, 81), (3, 35, 56);
54
55
56 • ida@capital.com', '20 May 1973'), (2, 'fredster66@aol.com', '11 January 2010'), (3, 'kovaka@crs.com', '30 October 2006');
57
58
59 • .id) VALUES (1, 'Freddie Smith', 'fredster66@aol.com', 1, 1), (2, 'Aaron Duarte', 'videogam3lover@gamil.com', 2, 2), (3, 'Justin Wong', 'j_wong5@gmail.com', 3, 3);
60
61
62 • ldie S.', '28 min'), (2, 'Aaron D.', '36 min'), (3, 'Justin W.', '06 min');
63
64
65 • 1), (2, 'Aaron D.', 3), (3, 'Justin W.', 2);
66
67
68 • IES (1, '07 June 2014', '$435,000'), (2, '23 December 2014', '$715,000'), (3, '11 April 2014', '$475,000');
69
70
71 • i (1, 'land', '11 Farm Street'), (2, 'house', '951 Fruitvale Avenue'), (3, 'store', '87 Monterey Drive');
72
73
74 • nr', '148 Diamond Drive'), (2, 'Jordan Stahr', '61 Holly Lane'), (3, 'Allison Kovak', '58 Chumara Drive');

```

Result Grid

landlord_id	name	number
1	John	5556601
2	April	5556602
3	Timmy	5556603

Landlord 5 x

Output

#	Time	Action	Message	Duration / Fetch
421	11:09:39	INSERT INTO Loan (loan_id, loan_approval, amount) VALUES (1, 'yes', '\$250,000'), (2, 'yes', '\$515,000'), (3, 'yes', '\$475,000');	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0	0.000 sec
422	11:09:39	INSERT INTO BankAccount (account_id, name, routing) VALUES (1, 'Freddie Smith', '916538840'), (2, 'Aaron Duarte', '916538840'), (3, 'Justin Wong', '916538840');	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0	0.000 sec
423	11:09:39	INSERT INTO Profile (profile_id, name, creation_date) VALUES (1, 'Linda Collins', '20 May 1973'), (2, 'Aaron D.', '36 min'), (3, 'Justin W.', '06 min');	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0	0.000 sec
424	11:09:39	INSERT INTO Payment (payment_id, payment_type, date) VALUES (1, 'cash', '13 June 2015'), (2, 'cash', '13 June 2015'), (3, 'cash', '13 June 2015');	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0	0.000 sec
425	11:09:39	INSERT INTO PaymentType (paymentType_id, billing_address, billing_city) VALUES (1, '431 Notbrook', 'San Francisco'), (2, '148 Diamond Drive', 'San Francisco'), (3, '61 Holly Lane', 'San Francisco');	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0	0.000 sec

## Section X: Testing

inserts   database model   tests

Limit to 1000 rows

```

109      -- Error: unknown column 'proerty_id; MISSPELLED
110
111      -- 21. Testing Open House table
112      • DELETE FROM OpenHouse WHERE address = '50 Chumasero Drive';
113      • UPDATE OpenHouse SET address = '673 Ashbury Street' WHERE id = 2;
114
115      -- 22. Testing Loan table
116      • DELETE FROM Loan WHERE amount = '$515,000';
117      • UPDATE Loan SET loan_approval = 'no' WHERE loan_id = 2;
118
119      -- 23. Testing Bank Account table
120      -- query: DELETE FROM BankAccount WHERE routing = '916548840';
121      -- Error: unknown column routing; 91638840 not 916546640
122      • UPDATE BankAccount SET routing = 110011000 WHERE name = 'Justin Wong';
123
124      -- 24. Testing Profile table
125      • DELETE FROM Profile WHERE name = 'Freddie Smith';
126      -- query: UPDATE Profile SET profile_id = 11 WHERE name = 'Linda Collins';
127      -- Error: cant delete/update child row; fk constraints fails
128
129      -- 25. Testing Payment table
130      • DELETE FROM Payment WHERE date = '05 September 2015';
131      -- query: UPDATE Payment SET payment_id = 17 WHERE payment_id = 3;
132      -- Error: cant delete/update parent row; fk constraints fails
133
134      -- 26. Testing Payment Type table
135      • DELETE FROM PaymentType WHERE billing_city = 'San Francisco';
136      -- query: UPDATE PaymentType SET paymentType_id = 5 WHERE billing_city = 'Sonoma';
137      -- Error: cant delete/update child row; fk constraints fails
138

```

Output

Action Output

#	Time	Action	Message	Duration / Fetch
✓ 4731	13:08:58	UPDATE Loan SET loan_approval = 'no' WHERE loan_id = 2	0 row(s) affected Rows matched: 0 Changed: 0 Warnings: 0	0.000 sec
✓ 4732	13:08:58	UPDATE BankAccount SET routing = 110011000 WHERE name ...	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.000 sec
✓ 4733	13:08:58	DELETE FROM Profile WHERE name = 'Freddie Smith'	1 row(s) affected	0.000 sec
✓ 4734	13:08:58	DELETE FROM Payment WHERE date = '05 September 2015'	1 row(s) affected	0.000 sec
✓ 4735	13:08:59	DELETE FROM PaymentType WHERE billing_city = 'San Francisco'	0 row(s) affected	0.000 sec

## Section XI: Testing Table

Entity	SQLQuery	Ok/Failed	Error Description	Possible Solution
Neighborhood	Delete	Failed	Can't delete parent row	Maybe updating the row directly
Neighborhood	Update	Ok	None	None
Landlord	Delete	Ok	None	None
Landlord	Update	Ok	None	None
House	Delete	Ok	None	None
House	Update	Ok	None	None
Room	Delete	Ok	None	None
Room	Update	Ok	None	None
Apartment	Delete	Ok	None	None
Apartment	Update	Ok	None	None
Unit	Delete	Ok	None	None
Unit	Update	Ok	None	None
Land	Delete	Ok	None	None
Land	Update	Ok	None	None
Store	Delete	Ok	None	None
Store	Update	Ok	None	None
Company	Delete	Ok	None	None
Company	Update	Failed	Can't update parent row	Delete the parent row, then insert a new row
Employee	Delete	Failed	Can't delete parent row	Update the row directly
Employee	Update	Failed	Can't update parent row	Delete the parent row then insert a new row
Bank	Delete	Failed	Can't delete parent row	Updating the row directly
Bank	Update	Ok	None	None
Client	Delete	Failed	Can't delete parent row	Updating the row directly
Client	Update	Failed	Can't update parent row	Delete the parent row then insert a new row
SearchEngine	Delete	Failed	Can't delete parent row	Update the row directly
SearchEngine	Update	Ok	None	None
Website	Delete	Failed	Can't delete parent row	Update the row directly
Website	Update	Failed	Unknown column bc I misspelled 'num_sessions'	Change from 'num_session' to 'num_sessions'
Account	Delete	Failed	Can't delete parent row	Update the row directly
Account	Update	Ok	None	None
User	Delete	Ok	None	None
User	Update	Ok	None	None

Session	Delete	Ok	None	None
Session	Update	Ok	None	None
List	Delete	Failed	Can't delete parent row	Update the row directly
List	Update	Ok	None	None
PropertyListing	Delete	Failed	Can't delete parent row	Update the row directly
PropertyListing	Update	Ok	None	None
Property	Delete	Ok	None	none
Property	Update	Failed	Unknown column, misspelled 'property_id'	Change property_id' to 'property_id'
OpenHouse	Delete	Ok	None	None
OpenHouse	Update	Ok	None	None
Loan	Delete	Ok	None	None
Loan	Update	Ok	None	none
BankAccount	Delete	Failed	Unknown column, mistyped routing value	Change routing value to 916538840
BankAccount	Update	Ok	None	None
Profile	Delete	Ok	None	None
Profile	Update	Failed	Can't delete/update child row	Updating the parent row with update the child row
Payment	Delete	Ok	None	None
Payment	Update	Failed	Can't delete/update parent row	Delete parent row then insert a new row
PaymentType	Delete	Ok	None	None
PaymentType	Update	Failed	Can't delete/update child row	Updating the parent row will update the child row