



**İstanbul
Bilgi University**

CMPE 351:Database Design Report

Abstract

Databases have become something pivotal for an organization to have in the modern era, they allow for the organization a way to store, retrieve and analyse data efficiently. Databases are capable of storing enormous amounts of data and information while also providing the ability to perform calculation far faster than an ordinary human can. System files can also work as databases but a better alternative would be the E-database, and in this area of computing, SQL databases are the staple. SQL databases are what I intend on implementing in this project.

Sadiq Tijjani Umar

119200113

Database Design Report

1) Explain Your Firm And Its Sector

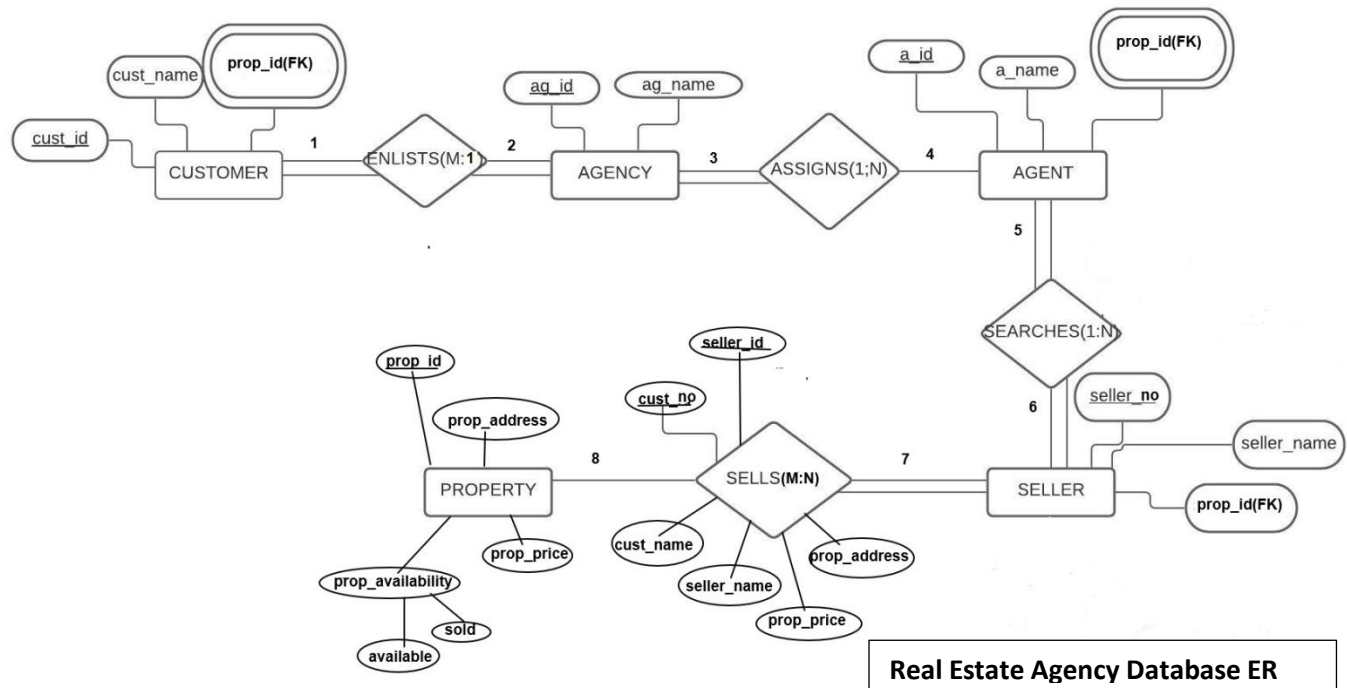
The firm to which I have been assigned is an agency, an agency is an establishment engaged in doing business for either another business(and/or agency) or a customer. The agency that I will design a database for is a real estate agency. This real estate agency does business in the housing sector of the market and not land, meaning the agency only handles the enlisting of agents to purchase ready made homes. The market provides an index of properties and agencies that an interested customer will select from.

2) Explain Your Database Design

An Index of properties of agency will be provided and CUSTOMER will select what he/she wants then CUSTOMER will then ENLIST(more than one CUSTOMER can ENLIST) the AGENCY, the AGENCY will then ASSIGN this project to an AGENT(an AGENT cannot work more than one project at a time, a project can have multiple homes requested by the customer), the AGENT will then SEARCH for a SELLER(SELLER can be able to sell more than one property at a time) in the market, then depending on the availability of the PROPERTY(Sold or Available), SELLER then SELLS the PROPERTY.

- CUSTOMER will have cust_id, cust_name and prop_id(as a foreign key). A CUSTOMER can have more than one wanted address.
- Cust_id , prop_id and ag_id is recorded in ENLISTS relationship
- AGENCY will have ag_id, ag_name. an Agency can ASSIGN work to more than one AGENT.
- Ag_id, a_id and prop_id(FK) is recorded in ASSIGNED relationship
- AGENT will have a_id, a_name, prop_id(inherited from customer request, which he will use to search).
- SEARCH will record prop_id , a_id and seller_no information.
- SELLER will have seller_name, seller_no, prop_id(as foreign key)
- Information of cust_no , cust_name , seller_name , seller_id and prop_address , prop_price are tracked in SELLS relation
- PROPERTY will have prop_id(primary key), prop_address, prop_availability, prop_price.

3) Draw The Entity Relationship Of Your Database And Provide Snapshots Of Your Tables.



PROPERTY Table

<u>prop_id</u>	prop_address	prop_availability	prop_price
110	House 2,128 Nkouchatta street, Abuja.	Available	100,000
111	House 3, 122 Nkouchatta street, Abuja.	Available	150,000
112	House 2, 122 Nkouchatta street, Abuja.	Available	99,000
113	House 5, 125 Nkouchatta street, Abuja.	Booked	87,500
114	House 6, 128 Nkouchatta street, Abuja.	Available	90,000
115	House 1, 128 Nkouchatta street, Abuja.	Available	89,000

CUSTOMER Table

<u>Cust_id</u>	Cust_name	prop_id
1	Micheal Donpo	110, 111
2	Wole Oluwa	114,115
3	Tanko Mangoro	113
4	Cen Emre	112

ENLSISTS Table

<u>cust_id</u>	prop_id	ag_id
1	110,111	20
2	114,115	21
3	113	22
4	112	23

AGENCY Table

<u>ag_id</u>	ag_name
20	Ertap Housing Solutions
21	Brains And Hammers
22	Gwarinpa Housing Management
23	Efab Housing Agency

ASSIGNS Table

<u>agency_id</u>	a_id	Prop_id
20	121	110
20	121	111
21	122	114
21	122	115

22	123	113
23	124	112

AGENT Table

<u>a_id</u>	a_name	AssignedProp_id
121	Fourier Irwin	110,111
122	Cevre Evrim	114,115
123	Roe Jogan	113
124	Maryam Kunama	112

SEARCHES Table

<u>agent_id</u>	prop_id	seller_no
121	110,111	333
122	114,115	331
123	113	332
124	112	334

SELLER Table

<u>Seller_no</u>	Seller_name	Prop_id
331	Kwame Nzuruma	114
331	Kwame Nzuruma	115
332	Sudan Omer	113
333	Micheal Morte	111
333	Micheal Morte	110
334	Ahmad Al-Khwarizmi	112

SELLS Table

<u>seller_id</u>	<u>seller_name</u>	<u>prop_address</u>	<u>prop_price</u>	<u>cust_no</u>	<u>cust_name</u>
331	Kwame Nzurumah	House 6, 128 Nkouchatta street, Abuja	90,000	2	Wole Oluwa
331	Kwame Nzurumah	House 1, 128 Nkouchatta street, Abuja	89,000	2	Wole Oluwa
333	Micheal Morte	House 3, 122 Nkouchatta street, Abuja	150,000	1	Micheal Donpo
333	Micheal Morte	House 2,128 Nkouchatta street, Abuja	100,000	1	Micheal Donpo
334	Ahmad Al-khwarizmi	House 2, 122 Nkouchatta street, Abuja.	99,000	4	Cen Emre

4) Explain Keys And Cardinalities For Each Table

- > In PROPERTY table prop_id is the primary key, it will be used in other tables.
- > In CUSTOMER table cust_id is the primary key, prop_id is foreign key
- > In ENLSITS table cust_id is primary key referencing the CUSTOMER table. Prop_id also referencing PROPERTY table
- > In AGENCY table ag_id is the primary key.
- > In ASSIGNS table agency_id(PK) and prop_id are referencing AGENCY and PROPERTY tables respectively.
- > In AGENT table a_id is the primary key and prop_id is referencing PROPERTY table.
- > In SEARCH table agent_id(Pk) and prop_id are referencing AGENT and PROPERTY tables respectively.
- > In SELLER table seller_no is the primary key and prop_id referencing PROPERTY table.
- > In SELLS table seller_id(pk) and cust_no are referencing the SELLER and CUSTOMER tables respectively

Cardinality Constraints

- 1) ENLISTS is M:1 because a customer can make more than one enlistment and one enlistment is made by only one customer.
- 2) ASSIGNS is 1:N because an agency can assign a project to an agent more than once and every assignment is made by the agency only.
- 3) SEARCHES is 1:N because an agent can make more than one search for a seller and a search is made by only one agent.
- 4) SELLS is M:N a seller may sell different properties and a sell may be made by more than one seller.

Participation Constraints

- 1 is total because every customer will have to make an enlistment to the agency about his/her wanted property.
- 2 is total because every agency needs a customer in order to carry out the operations.
- 3 is total because every agency assigns at least one project to an Agent.
- 4 is partial because some agents may not be assigned a project. As an agent can only work on one project at a time, so if he is already working on something, the project will not be assigned.
- 5 is total because every agent makes at least one search for a seller of a property.
- 6 is total because every Seller needs a search to be done in order to be made aware of the interest in his property.
- 7 is total because every seller makes at least one sale of a property.
- 8 is partial because not every property may be sold as they may already be booked.

5) Normalize all tables to 4NF

- To be normalized all tables will be made to have only atomic values, then values have to be same kind or type and each record will be uniquely identified. Then table will then be altered such that all columns are dependent on a candidate key. Then if needed, tables will then be altered such that all columns depend on only ONE candidate key. Below are the solutions:

CUSTOMER Table(4NF)

First Table

<u>cust_id</u>	cust_name	prop_id
1	Micheal Donpo	110

1	Micheal Donpo	111
2	Wole Oluwa	114
2	Wole Oluwa	115
3	Tanko Mangoro	113
4	Cen Emre	112

Second Table

<u>cust_id</u>	<u>prop_id</u>
1	110
1	111
4	112
3	113
2	114
2	115

Final Table

<u>CUST_ID</u>	<u>CUST_NAME</u>
1	Micheal Donwo
2	Wole Oluwa
3	Tanko Mangoro
4	Cen Emre

ENLISTS Table(4NF)

First Table

<u>cust_id</u>	<u>prop_id</u>	<u>ag_id</u>
1	110	20
1	111	20
2	114	21
2	115	21
3	113	22
4	112	23

- Second table will be the same table as the second table in CUSTOMER table normalization.

Final Table

<u>C_ID</u>	AG_ID
1	20
2	21
3	22
4	23

ASSIGNS Table(4NF)

First Table

<u>ag_id</u>	prop_id
20	110
20	111
21	114
21	115
22	113
23	112

Second Table

<u>a_id</u>	prop_id
121	110
121	111
122	114
122	115
123	113
124	112

Final Table

<u>AG_ID</u>	<u>A_ID</u>
20	121
21	122
22	123
23	124

AGENT Table(4NF)

First Table

<u>a_id</u>	<u>a_name</u>	<u>prop_id</u>
121	Fourier Irwin	110
121	Fourier Irwin	111
122	Cevre Evrim	114
122	Cevre Evrim	115
123	Roe Jogan	113
124	Maryam Kunama	112

Second Table

<u>a_id</u>	<u>prop_id</u>
121	110
121	111
124	112
123	113
122	114
122	115

Final Table

<u>A_ID</u>	<u>A_NAME</u>
121	Fourier Irwin
122	Cevre Evrim
123	Roe Jogan
124	Maryam

	Kunama
--	--------

SEARCHES Table(4NF)

First Table

<u>agent_id</u>	<u>prop_id</u>	<u>seller_no</u>
121	110	333
121	111	333
122	114	331
122	115	331
123	113	332
124	112	334

Second Table

<u>agent_id</u>	<u>prop_id</u>
121	110
121	111
122	114
122	115
123	113
124	112

Third Table

Seller_no	prop_id
333	110
333	111
331	114
331	115
332	113
334	112

Final Table

<u>AGENT_ID</u>	S_NO
121	333
122	331
123	332
124	334

SELLER Table(4NF)

First Table

<u>Seller_no</u>	<u>Prop_id</u>
331	114
331	115
332	113
333	111
333	110
334	112

Final Table

<u>SELLER_NO</u>	<u>SELLER_NAME</u>
331	Kwame Nzurumah
332	Sudan Omer
333	Micheal Morte
334	Ahmad Al-Khwarizmi

SELLS Table(4NF)

First Table

<u>prop_id</u>	<u>prop_address</u>	<u>prop_price</u>
114	House 6, 128 Nkouchatta street, Abuja	90,000
115	House 1, 128 Nkouchatta street, Abuja	89,000
111	House 3, 122 Nkouchatta street, Abuja	150,000
110	House 2,128 Nkouchatta street, Abuja	100,000
112	House 2, 122 Nkouchatta street, Abuja.	99,000

Second Table

<u>seller_id</u>	seller_name	<u>prop_id</u>	cust_no	cust_name
331	Kwame Nzurumah	114	2	Wole Oluwa
331	Kwame Nzurumah	115	2	Wole Oluwa
333	Micheal Morte	111	1	Micheal Donpo
333	Micheal Morte	110	1	Micheal Donpo
334	Ahmad Al- khwarizmi	112	4	Cen Emre

Third Table

<u>seller_id</u>	<u>prop_id</u>
331	114
331	115
333	111
333	110
334	112

Fourth Table

<u>cust_no</u>	<u>prop_id</u>
2	114
2	115
1	111
1	110
4	112

Fifth Table

<u>cust_id</u>	<u>cust_name</u>
1	Micheal Donpo
2	Wole Oluwa
4	Cen Emre

Final table

<u>SELLER_ID</u>	<u>SELLER_NAME</u>	<u>CUST_NO</u>
331	Kwame Nzurumah	1
333	Micheal Morte	2
334	Ahmad Al-Khwarizmi	4

Appendices

1)Code Implementation

In the implementation of this project, the jupyter IDE was used, python native libraries were used to create functions such as *execute_query*, *read_query*, *create_server_connection* and *create_database* whose function was to assist in making and connecting to the database, and then also to execute and get results from queries. After our database was made, we then created the necessary tables and populated them. The simple User Interface App is in short a series of if else statements that are conditioned to give unique responses and results depending on the inputs of the user. We also had to import SQL and Pandas libraries into the environment.

2)Improvements

Improvements can be made by maybe adding more functions to the app but most queries can be executed using the Advanced Command feature.