Real Estate Residential Database

Group Project Final Report

Group #9

Bhavin Darji bhavin.darji@baruchmail.cuny.edu
Mohammed Rahman mohammed.rahman8@baruchmail.cuny.edu
Marina Lukyanchenko marina.lukyanchenko@baruchmail.cuny.edu
Sandy Ramos sandy.ramos@baruchmail.cuny.edu
Soohun Han soohun.han@baruchmail.cuny.edu

CIS3400 – EMWA

Introduction

We are a small residential real estate company that focuses on leasing out apartment buildings. With this database, we will be able to identify which tenant lives in which building as well as their lease information. Excel sheets are not meant to be a database. With our database, it will eliminate the need of keeping track of multiple excel sheets and keep the information in one centralized location. It will also make it easier for the employees to know how much rent is paid for what kind of apartment and how many apartments are still vacant.

To run a successful business you can't have information in multiple places. With a database, this information can be tracked and organized in one central location. To accurately identify which tenant rents out which apartment, how much rent they pay and their lease information. Keep track of asset manager information such as what building they are in charge of and contact information.

List of Entities (Tables)

- Apartment (# of rooms, floor, apartment size, etc.)
- Appointment (Agent name, appointment date, time, etc.)
- AssetManager (Employees in charge of building management)
- Building (Address, # of units, vacancy/rental status, etc.)
- Lease (Lease start/end dates, amount of rent, rent due dates, etc.)
- Tenant (Tenant name, phone number, etc.)
- ZipCode (City, state, zip code)

Distribution of Duties

- System Analyst: Bhavin Darji
- Logical/Physical Modeler: Mohammed Rahman, Sandy Ramos, Marina Lukyanchenko
- Database Implementer: *Team Effort*
- Application Developer: *Team Effort*
- Documentation Writer: Soohun Han

Entity-Relationship Model Diagram

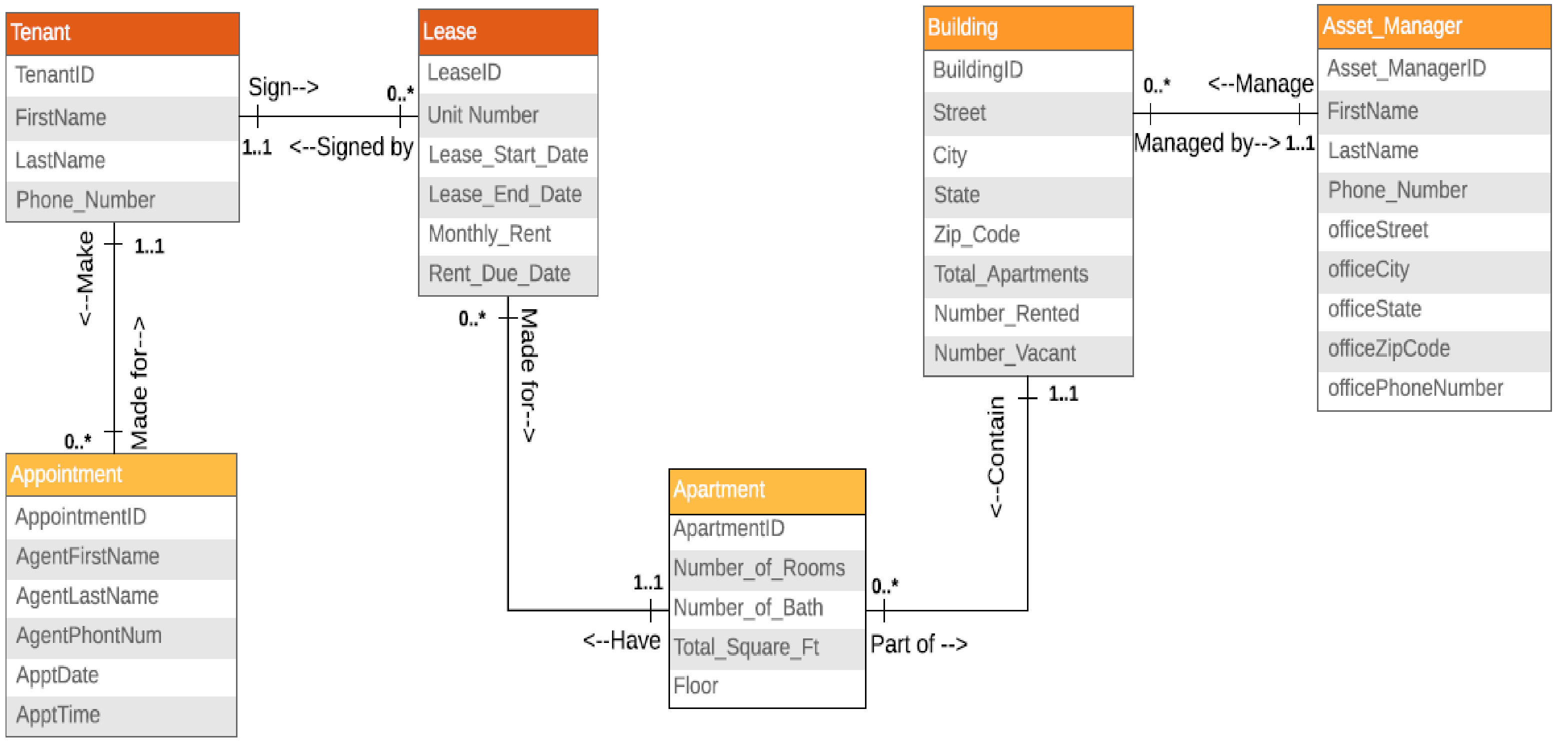
Model: Real Estate Residential

Created by: Bhavin Darji, Marina Lukyanchenko, Mohammed Rahman,Sandy Ramos, Soohun Han

Due Date: Oct 24,2018

CIS 3400-EMWA **Group #9**

CIS3400 Group Project E-R Diagram



Set of Relations (Normalized)

1. **AssetManager** (AssetManagerID <Key>, FirstName, LastName, Phone_Number, Street, City, State, ZipCode, OfficePhoneNumber)

   Key: AssetManagerID
   FD1: AssetManagerID → FirstName, LastName, Phone_Number, Street, City, State, ZipCode, OfficePhoneNumber
   FD2: ZipCode → City, State

   1. It is in 1NF from because it's a relation
   2. In 2NF because key is the only one attribute so, no partial key dependency
   3. Not in 3NF because of transitive dependency

   Normalization: Split the relation AssetManager into 2 new relations A and B. Copy ZipCode and remove City and State

   A (ZipCode <Key>, City, State)
   Key: ZipCode
   FD1: ZipCode → City, State

   B (AssetManagerID <Key>, FirstName, LastName, Phone_Number, Street, ZipCode <FK>, OfficePhoneNumber)
   Key: AssetManagerID
   FD1: AssetManagerID → FirstName, LastName, Phone_Number, Street, ZipCode, OfficePhoneNumber)

   1. It is in 1NF from because it's a relation
   2. In 2NF because key is the only one attribute so, no partial key dependency
   3. In 3NF because there is no transitive dependency

2. **Building** (BuildingID <Key>, Street, City, State, ZipCode, TotalApartments, NumberRented, NumberVacant, AssetManagerID <FK>)

   Key: BuildingID
   FD1: BuildingID → Street, City, State, ZipCode, TotalApartments, NumberRented, NumberVacant, AssetManagerID
   FD2: ZipCode → City, State

   1. It is in 1NF from because it's a relation
   2. In 2NF because key is the only one attribute so, no partial key dependency
   3. Not in 3NF because of transitive dependency

   Normalization: Split the relation Building into two new relation A and B. Copy Zip_Code and remove city and state

A (ZipCode <Key>, City, State)
Key: ZipCode
FD1: ZipCode → City, State

B (BuildingID <Key>, Street, ZipCode <FK>, TotalApartments, NumberRented, NumberVacant, AssetManagerID <FK>)
Key: BuildingID
FD1: BuildingID → Street, ZipCode, TotalApartments, NumberRented, NumberVacant, AssetManagerID

1. It is in 1NF from because it's a relation
2. In 2NF because key is the only one attribute so, no partial key dependency
3. In 3NF because there is no transitive dependency

3. **Apartment** (ApartmentID <Key>, NumberOfRooms, NumberOfBath, TotalSquareFt, Floor, BuildingID <FK>)

Key: ApartmentID
FD1: ApartmentID → NumberOfRooms, NumberOfBath, TotalSquareFt, Floor, BuildingID

1. It is in 1NF from because it's a relation
2. In 2NF because key is the only one attribute so, no partial key dependency
3. IN 3NF because there is no transitive dependency

4. **Tenant** (TenantID <Key>, FirstName, LastName, PhoneNumber, AppointmentID <FK>)

Key: TenantID
FD1: TenantID → FirstName, LastName, PhoneNumber, AppointmentID

1. It is in 1NF from because it's a relation
2. In 2NF because key is the only one attribute so, no partial key dependency
3. IN 3NF because there is no transitive dependency

5. **Lease** (LeaseID <Key>, UnitNumber, LeaseStartDate, LeaseEndDate, MonthlyRent, RentDueDate, ApartmentID <FK>)

Key: LeaseID
FD1: LeaseID → UnitNumber, LeaseStartDate, LeaseEndDate, MonthlyRent, RentDueDate, ApartmentID

1. It is in 1NF from because it's a relation
2. In 2NF because key is the only one attribute so, no partial key dependency
3. IN 3NF because there is no transitive dependency

6. **Appointment** (AppointmentID <Key>, AgentFirstName, AgentLastName, AgentPhoneNum, ApptDate, ApptTime, TenantID <FK>)

Key: AppointmentID
FD1: AppointmentID → AgentFirstName, AgentLastName, AgentPhoneNum, ApptDate, ApptTime, TenantID

1. It is in 1NF from because it’s a relation
2. In 2NF because key is the only one attribute so, no partial key dependency
3. IN 3NF because there is no transitive dependency

SQL Statements

```
=======================================
       CREATE TABLE SQL Commands
=======================================

#Create AssetManager Table
CREATE TABLE AssetManager
(
AssetManagerID      VARCHAR(10) NOT NULL,
FirstName           VARCHAR(35),
LastName            VARCHAR(35),
Phone_Number        VARCHAR(15),
OfficeStreet        VARCHAR(35),
OfficePhoneNumber   VARCHAR(15),
ZipCode             VARCHAR(8) NOT NULL,
CONSTRAINT pk_AssetManagerID
     PRIMARY KEY (AssetManagerID)
)

#Create Building Table
CREATE TABLE Building
(
BuildingID               VARCHAR(10) NOT NULL,
Street              VARCHAR(35),
TotalApartments     NUMBER,
NumberRented        NUMBER,
NumberVacant        NUMBER,
AssetManagerID      VARCHAR(10) NOT NULL,
ZipCode             VARCHAR(8) NOT NULL,
CONSTRAINT pk_BuildingID
     PRIMARY KEY (BuildingID)
)
```

```
#Create Apartment Table
CREATE TABLE Apartment
(
ApartmentID         VARCHAR(10) NOT NULL,
NumberOfRooms       NUMBER,
NumberOfBath        NUMBER,
TotalSquareFt       NUMBER,
Floor               NUMBER,
BuildingID          VARCHAR(10) NOT NULL,
CONSTRAINT pk_ApartmentID
     PRIMARY KEY (ApartmentID)
)

#Create Tenant Table
CREATE TABLE Tenant
(
TenantID       VARCHAR(10) NOT NULL,
FirstName      VARCHAR(35),
LastName       VARCHAR(35),
PhoneNumber    VARCHAR(15),
AppointmentID  VARCHAR(10) NOT NULL,
CONSTRAINT pk_TenantID
     PRIMARY KEY (TenantID)
)

#Create Lease Table
CREATE TABLE Lease
(
LeaseID             VARCHAR(10) NOT NULL,
UnitNumber          NUMBER,
LeaseStartDate      DATE,
LeaseEndDate        DATE,
Monthly_Rent        NUMBER,
RentDueDate         VARCHAR,
ApartmentID         VARCHAR(10) NOT NULL,
TenantID            VARCHAR(10) NOT NULL,
CONSTRAINT pk_LeaseID
     PRIMARY KEY (LeaseID)
)

#Create Appointment Table
CREATE TABLE Appointment
(
AppointmentID       VARCHAR(10) NOT NULL,
AgentFirstName      VARCHAR(35),
AgentLastName       VARCHAR(35),
AgentPhoneNum       VARCHAR(15),
ApptDate            DATE,
ApptTime            DATE,
TenantID            VARCHAR(10) NOT NULL,
CONSTRAINT pk_AppointmentID
     PRIMARY KEY(AppointmentID)
)

#Create ZipCode Table
CREATE TABLE ZipCode
(
ZipCode   VARCHAR NOT NULL,
City      VARCHAR,
State     VARCHAR,
CONSTRAINT pk_zipcode
     PRIMARY KEY (ZipCode)
)

=======================================
     ALTER TABLE & Add Foreign Key
=======================================

#Set ZipCode in AssetManager as a Foreign Key
ALTER TABLE AssetManager
ADD CONSTRAINT fk_zipcodeassetmanager FOREIGN KEY (ZipCode)
REFERENCES ZipCode

#Set ZipCode in Building as a Foreign Key
ALTER TABLE Building
ADD CONSTRAINT fk_zipcodebuilding FOREIGN KEY (ZipCode)
REFERENCES ZipCode

#Set TenantID in Lease as a Foreign Key
ALTER TABLE Lease
ADD CONSTRAINT fk_tenantid FOREIGN KEY (TenantID)
REFERENCES Tenant

#Set AssetManagerID in Building as a Foreign Key
ALTER TABLE Building
ADD CONSTRAINT fk_assetmanagerid FOREIGN KEY
(AssetManagerID)
REFERENCES AssetManager

#Set BuildingID in Apartment as a Foreign Key
ALTER TABLE Apartment
```

```
     ADD CONSTRAINT fk_buildingid FOREIGN KEY (BuildingID)
     REFERENCES Building

     #Set TenantID in Appointment as a Foreign Key
     ALTER TABLE Appointment
     ADD CONSTRAINT fk_tenantidappointment FOREIGN KEY
     (TenantID)
     REFERENCES Tenant

     #Set ApartmentID in Lease as a Foreign Key
     ALTER TABLE Lease
     ADD CONSTRAINT fk_apartmentid FOREIGN KEY (ApartmentID)
     REFERENCES Apartment

     =======================================
           INSERT Data Into the Table
     =======================================

     #Insert data into Apartment Table
     INSERT INTO Apartment
     VALUES ('APT101', '1', '1', '550', '3', 'B101');

     INSERT INTO Apartment
     VALUES ('APT102', '2', '1', '750', '2', 'B102');

     INSERT INTO Apartment
     VALUES ('APT103', '3', '2', '1050', '5', 'B103');

     INSERT INTO Apartment
     VALUES ('APT104', '1', '1', '550', '6', 'B104');

     INSERT INTO Apartment
     VALUES ('APT105', '2', '1', '750', '9', 'B105');

     #Insert data into Appointment Table
     INSERT INTO Appointment
     VALUES ('A101','Thomas','Snyder','718-521
     2581','10/31/2018','10:00:00 AM','T101');

     INSERT INTO Appointment
     VALUES('A102','David','Blossem','212-485-1241','11/07/2018',
     '11:30:00 AM','T102');

     INSERT INTO Appointment
     VALUES('A103','Rick','Glover','347-182-
5161','11/19/2018','0
```

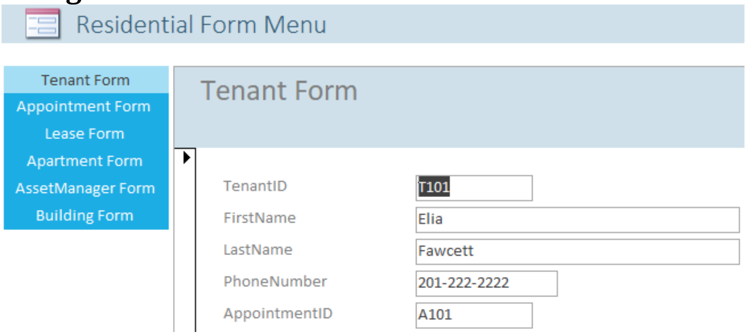
```
     9:00:00 AM','T103');

     INSERT INTO Appointment
     VALUES('A104','Samantha','Nyce','202-518-1050','11/22/2018',
     '01:00:00 PM','T104');

     INSERT INTO Appointment
     VALUES('A105','Robert','Smith','314-586-
1825','12/05/2018','
     03:30:00 PM','T105');

     #Insert data into AssetManager Table
     INSERT INTO AssetManager
     VALUES ('M101','John','Smith','902-933-3563','401 Pyle
     Drive','212-207-1111','07026');

     INSERT INTO AssetManager
     VALUES ('M102','Dylan','Doyle','917-462-1491','20 Smith
     Town','212-207-2222','07010');

     INSERT INTO AssetManager
     VALUES ('M103', 'Julia', 'Kelly','347-182-5610', '67
     Woodhaven Blvd','212-207-3333','07631');

     INSERT INTO AssetManager
     VALUES ('M104','Tommie','Wilkins','646-710-9311','33 3rd
     Ave','212-207-4444','07665');

     INSERT INTO AssetManager
     VALUES ('M105','Sharon','Neal','917-462-1491','20 Smith
     Town','212-207-5555','07010');

     #Insert data into Building Table
     INSERT INTO Building
     VALUES ('B101', '8989 Smith Rd', 30, 17, 13, 'M101',
     '07026');

     INSERT INTO Building
     VALUES ('B102', '514 Mamaroneck Rd', 50, 42, 8, 'M102',
     '07010');

     INSERT INTO Building
     VALUES ('B103', '333 Grand Ave', 66, 50, 16, 'M103',
     '07631');

     INSERT INTO Building
```

```
VALUES ('B104', '111 Bergen Blvd.', 100, 67, 33, 'M104',
'07665');

INSERT INTO Building
VALUES ('B105', '512 Maroon Rd', 20, 18, 2, 'M105',
'07010');


#Insert data into Lease Table
INSERT INTO Lease
VALUES ('L101', '1', '04/2/2017', '04/2/2018', '1400',
'5th', "APT101", 'T101');

INSERT INTO Lease
VALUES ('L102', '2', '06/10/2017', '06/10/2018', '1000',
'5th', "APT101", 'T102');

INSERT INTO Lease
VALUES ('L103', '3', '07/15/2017', '07/15/2018', '1400',
'5th', "APT105", 'T103');

INSERT INTO Lease
VALUES ('L104', '4', '10/2/2017', '10/2/2018', '1800',
'3rd', "APT103", 'T104');

INSERT INTO Lease
VALUES ('L105', '5', '01/2/2017', '01/2/2018', '1400',
'1st', "APT104", 'T105');

#Insert data into Tenant Table
INSERT INTO Tenant
VALUES ('T101', 'Elia', 'Fawcett', '201-222-2222', 'A101');

INSERT INTO Tenant
VALUES ('T102', 'Ishwarya', 'Roberts', '201-333-3333',
'A102');

INSERT INTO Tenant
VALUES ('T103', 'Frederic', 'Smith', '201-444-4444',
'A103');

INSERT INTO Tenant
VALUES ('T104', 'Goldie', 'Montand', '201-555-5555',
'A104');

INSERT INTO Tenant
```

```
VALUES ('T105', 'Dheeraj', 'Mamamoo', '201-666-6666',
'A105');

#Insert data into ZipCode Table
INSERT INTO ZipCode
VALUES ('07026', 'Garfield', 'NJ');

INSERT INTO ZipCode
VALUES ('07621', 'Bergenfield', 'NJ');

INSERT INTO ZipCode
VALUES ('07010', 'Cliffside Park', 'NJ');

INSERT INTO ZipCode
VALUES ('07631', 'Englewood', 'NJ');

INSERT INTO ZipCode
VALUES ('07665', 'Teaneck', 'NJ');
```

Forms

Navigation Form

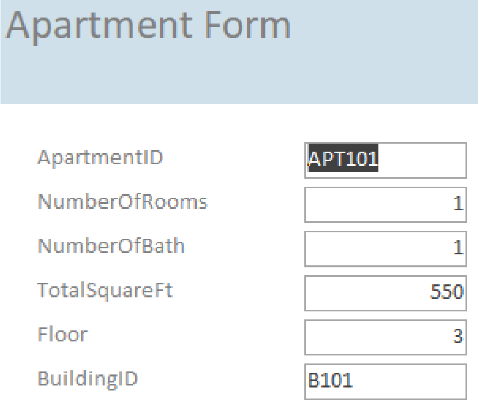


Tenant Form

TenantID	T101
FirstName	Elia
LastName	Fawcett
PhoneNumber	201-222-2222
AppointmentID	A101

-This is the main navigation form where you can access different forms from the menu

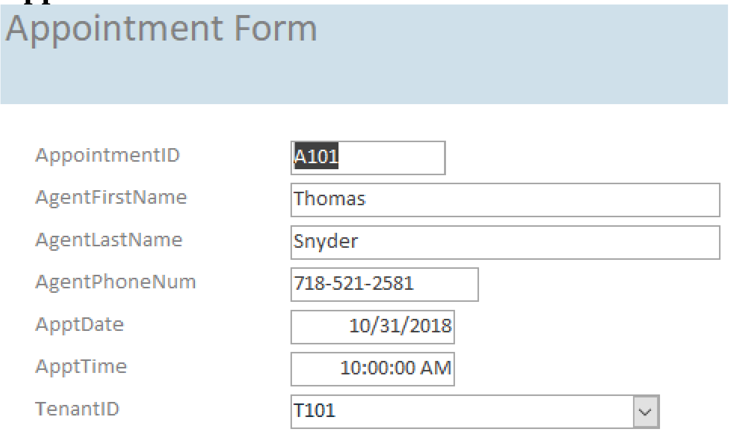
Apartment Form



Apartment Form

ApartmentID	APT101
NumberOfRooms	1
NumberOfBath	1
TotalSquareFt	550
Floor	3
BuildingID	B101

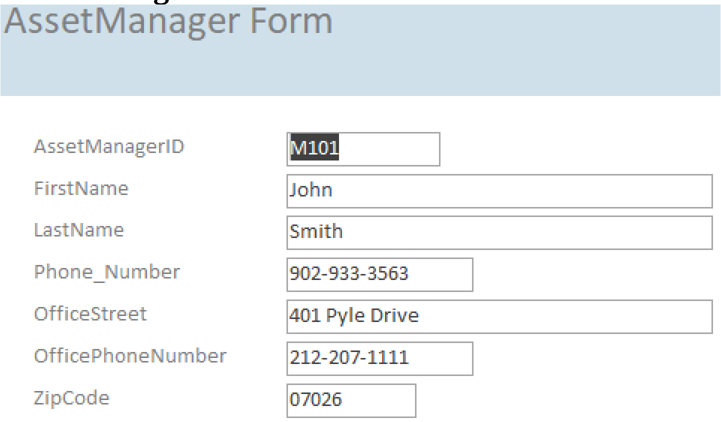
Appointment Form



Appointment Form

AppointmentID	A101
AgentFirstName	Thomas
AgentLastName	Snyder
AgentPhoneNum	718-521-2581
ApptDate	10/31/2018
ApptTime	10:00:00 AM
TenantID	T101

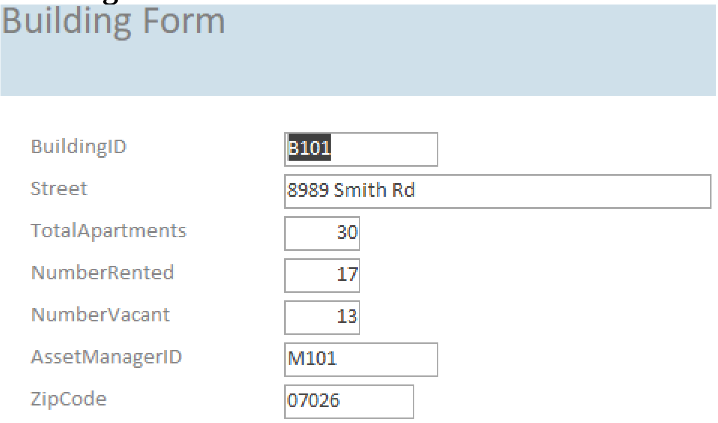
AssetManager Form



AssetManager Form

AssetManagerID	M101
FirstName	John
LastName	Smith
Phone_Number	902-933-3563
OfficeStreet	401 Pyle Drive
OfficePhoneNumber	212-207-1111
ZipCode	07026

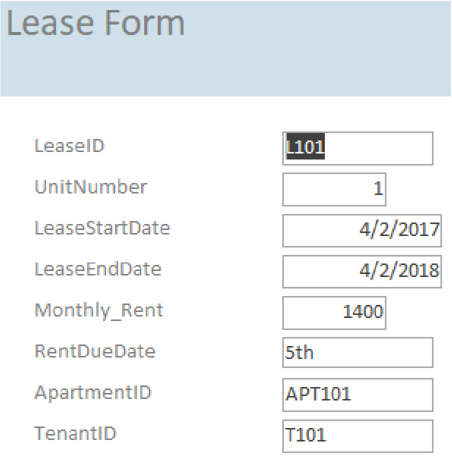
Building Form



Building Form

BuildingID	B101
Street	8989 Smith Rd
TotalApartments	30
NumberRented	17
NumberVacant	13
AssetManagerID	M101
ZipCode	07026

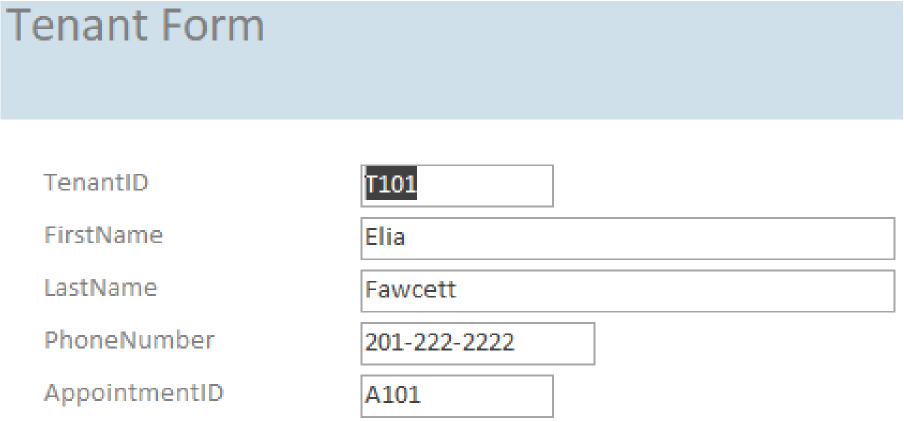
Lease Form



Lease Form

LeaseID	L101
UnitNumber	1
LeaseStartDate	4/2/2017
LeaseEndDate	4/2/2018
Monthly_Rent	1400
RentDueDate	5th
ApartmentID	APT101
TenantID	T101

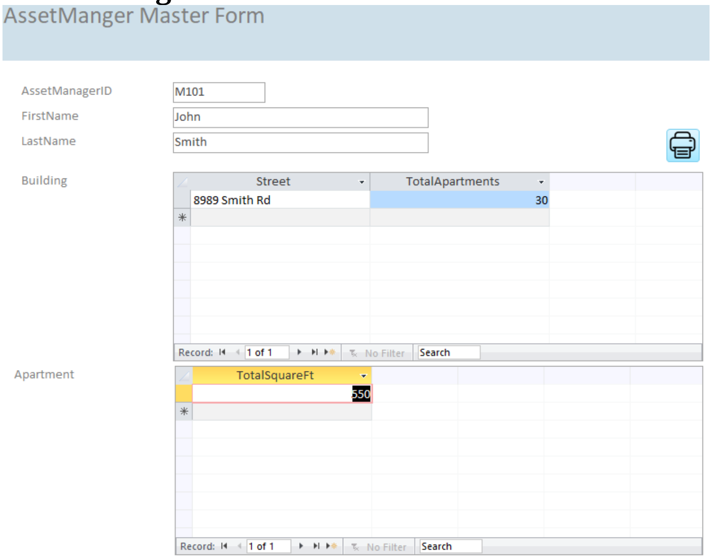
Tenant Form



Tenant Form

TenantID	T101
FirstName	Elia
LastName	Fawcett
PhoneNumber	201-222-2222
AppointmentID	A101

AssetManager Master Form



AssetManger Master Form

AssetManagerID	M101
FirstName	John
LastName	Smith

Building

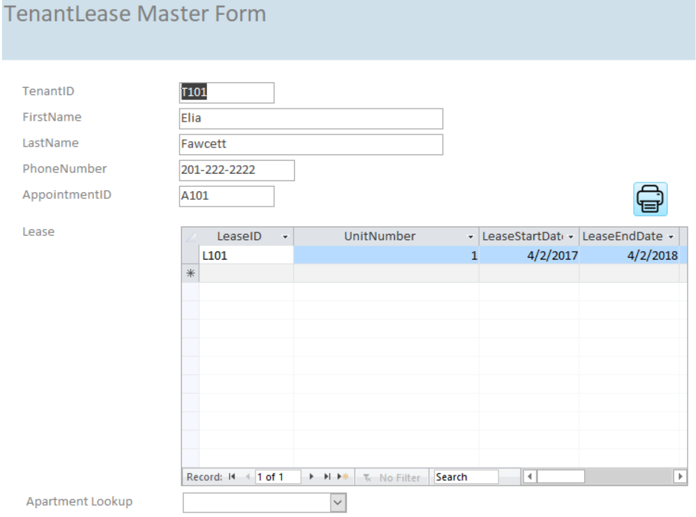
Street	TotalApartments
8989 Smith Rd	30

Apartment

TotalSquareFt
550

-AssetManager Master Form allows you to view AssetManager, Building, and Apartment tables at the same time. It has a print button that prints the current form.

TenantLease Master Form



TenantLease Master Form

TenantID	T101
FirstName	Elia
LastName	Fawcett
PhoneNumber	201-222-2222
AppointmentID	A101

Lease

LeaseID	UnitNumber	LeaseStartDat	LeaseEndDate
L101	1	4/2/2017	4/2/2018

Record: 1 of 1

Apartment Lookup

-TenantLease Master Form allows you to view the tenant and lease tables simultaneously. It has the print button to print the current form as well as apartment lookup feature on the bottom.

For all the forms containing names, we applied visual basic StrConv(n, vbProperCase) function to capitalize the first letter of names in the following way:

```
Private Sub LastName_AfterUpdate()
     LastName = StrConv(LastName, vbProperCase)
End Sub
```

Conclusion

a) the group's experience with the project (which steps were the most difficult? Which were the easiest? what did you learn that you did not imagine you would have? if you had to do it all over again, what would you have done differently?)

The steps for each phase were outlined clearly so we felt it was more manageable to handle than expected. However, for each step, we had to refine our outputs due to some mismatch in relationships among tables.

Using the SQL to create and connect tables went smoothly, except for inserting data, we encountered trouble with some of the records due to minor syntax errors (like the use of single or double quotation marks).

In retrospect, we feel like outlining and identifying the proper tables, fields, and relationships was the hardest part of the project. Applying SQL statements gave us some unexpected troubles as well which was not very obvious to notice.

If we were to do it again, we would want to take more time inspecting the relationships; identify all possible reasons for not-working SQL commands; and also have a chance to thoroughly study the VBA code to create more practical and advanced forms.

b) if the proposed benefits can be realized by the new system

We believe that our database will satisfy the needs introduced in the proposal - replacing the use of multiple excel sheets, keep information in one location, and identify employees, rent, and vacancy info with ease.