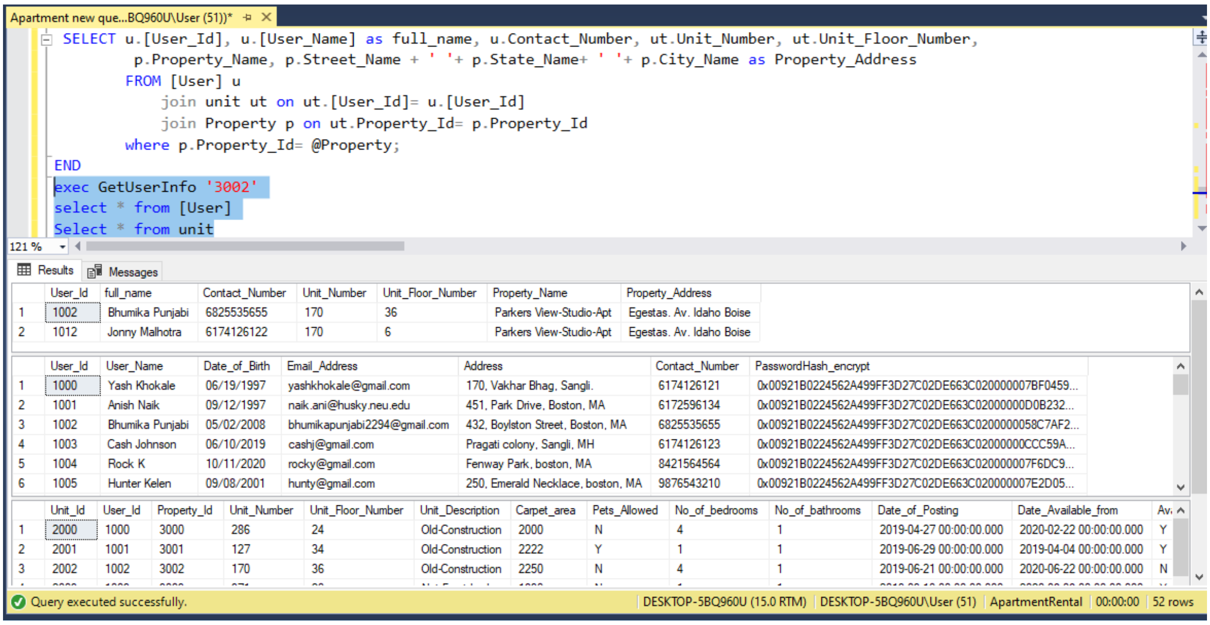
STORED PROCEDURES:

1) Stored Procedure to get user info who has rented the property



```
SELECT u.[User_Id], u.[User_Name] as full_name, u.Contact_Number, ut.Unit_Number, ut.Unit_Floor_Number,
         p.Property_Name, p.Street_Name + ' '+ p.State_Name+ ' '+ p.City_Name as Property_Address
        FROM [User] u
            join unit ut on ut.[User_Id]= u.[User_Id]
            join Property p on ut.Property_Id= p.Property_Id
        where p.Property_Id= @Property;
END
exec GetUserInfo '3002'
select * from [User]
Select * from unit
```

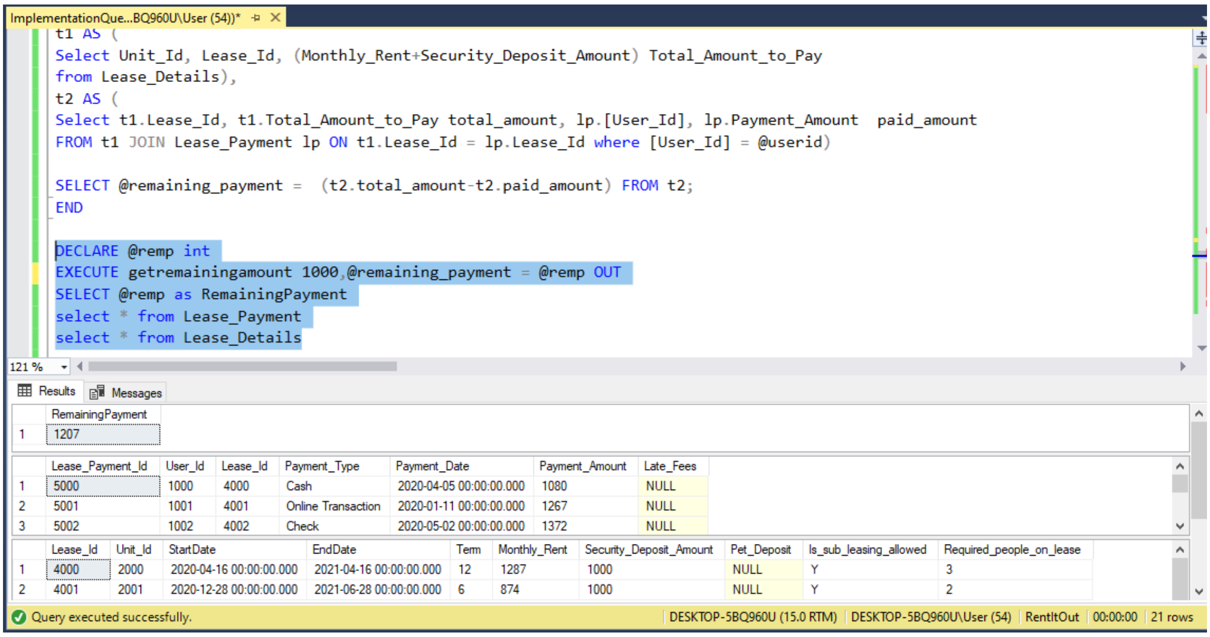
	User_Id	full_name	Contact_Number	Unit_Number	Unit_Floor_Number	Property_Name	Property_Address
1	1002	Bhumika Punjabi	6825535655	170	36	Parkers View-Studio-Apt	Egestas. Av. Idaho Boise
2	1012	Jonny Malhotra	6174126122	170	6	Parkers View-Studio-Apt	Egestas. Av. Idaho Boise

	User_Id	User_Name	Date_of_Birth	Email_Address	Address	Contact_Number	PasswordHash_encrypt
1	1000	Yash Khokale	06/19/1997	yashkhokale@gmail.com	170, Vakhar Bhag, Sangli.	6174126121	0x00921B0224562A499FF3D27C02DE663C020000007BF0459...
2	1001	Anish Naik	09/12/1997	naik.ani@husky.neu.edu	451, Park Drive, Boston, MA	6172596134	0x00921B0224562A499FF3D27C02DE663C02000000D0B232...
3	1002	Bhumika Punjabi	05/02/2008	bhumikapunjabi2294@gmail.com	432, Boylston Street, Boston, MA	6825535655	0x00921B0224562A499FF3D27C02DE663C0200000058C7AF2...
4	1003	Cash Johnson	06/10/2019	cashj@gmail.com	Pragati colony, Sangli, MH	6174126123	0x00921B0224562A499FF3D27C02DE663C02000000CCC59A...
5	1004	Rock K	10/11/2020	rocky@gmail.com	Fenway Park, boston, MA	8421564564	0x00921B0224562A499FF3D27C02DE663C020000007F6DC9...
6	1005	Hunter Kelen	09/08/2001	hunty@gmail.com	250, Emerald Necklace, boston, MA	9876543210	0x00921B0224562A499FF3D27C02DE663C020000007E2D05...

	Unit_Id	User_Id	Property_Id	Unit_Number	Unit_Floor_Number	Unit_Description	Carpet_area	Pets_Allowed	No_of_bedrooms	No_of_bathrooms	Date_of_Posting	Date_Available_from	Av.
1	2000	1000	3000	286	24	Old-Construction	2000	N	4	1	2019-04-27 00:00:00.000	2020-02-22 00:00:00.000	Y
2	2001	1001	3001	127	34	Old-Construction	2222	Y	1	1	2019-06-29 00:00:00.000	2019-04-04 00:00:00.000	Y
3	2002	1002	3002	170	36	Old-Construction	2250	N	4	1	2019-06-21 00:00:00.000	2020-06-22 00:00:00.000	N

```
Query executed successfully.    DESKTOP-5BQ960U (15.0 RTM) | DESKTOP-5BQ960U\User (51) | ApartmentRental | 00:00:00 | 52 rows
```

2) Stored Procedure to Check Pending Payment of user(rentier)



```
t1 AS (
Select Unit_Id, Lease_Id, (Monthly_Rent+Security_Deposit_Amount) Total_Amount_to_Pay
from Lease_Details),
t2 AS (
Select t1.Lease_Id, t1.Total_Amount_to_Pay total_amount, lp.[User_Id], lp.Payment_Amount  paid_amount
FROM t1 JOIN Lease_Payment lp ON t1.Lease_Id = lp.Lease_Id where [User_Id] = @userid)

SELECT @remaining_payment =  (t2.total_amount-t2.paid_amount) FROM t2;
END

DECLARE @remp int
EXECUTE getremainingamount 1000,@remaining_payment = @remp OUT
SELECT @remp as RemainingPayment
select * from Lease_Payment
select * from Lease_Details
```

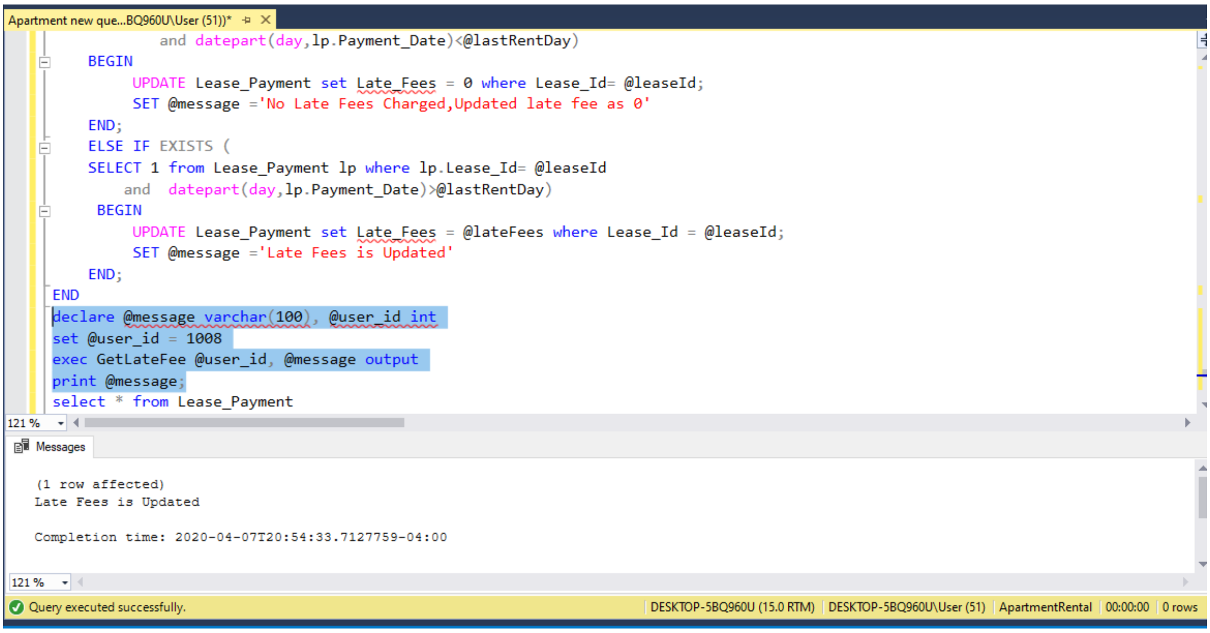
	RemainingPayment
1	1207

	Lease_Payment_Id	User_Id	Lease_Id	Payment_Type	Payment_Date	Payment_Amount	Late_Fees
1	5000	1000	4000	Cash	2020-04-05 00:00:00.000	1080	NULL
2	5001	1001	4001	Online Transaction	2020-01-11 00:00:00.000	1267	NULL
3	5002	1002	4002	Check	2020-05-02 00:00:00.000	1372	NULL

	Lease_Id	Unit_Id	StartDate	EndDate	Term	Monthly_Rent	Security_Deposit_Amount	Pet_Deposit	Is_sub_leasing_allowed	Required_people_on_lease
1	4000	2000	2020-04-16 00:00:00.000	2021-04-16 00:00:00.000	12	1287	1000	NULL	Y	3
2	4001	2001	2020-12-28 00:00:00.000	2021-06-28 00:00:00.000	6	874	1000	NULL	Y	2

```
Query executed successfully.    DESKTOP-5BQ960U (15.0 RTM) | DESKTOP-5BQ960U\User (54) | RentItOut | 00:00:00 | 21 rows
```

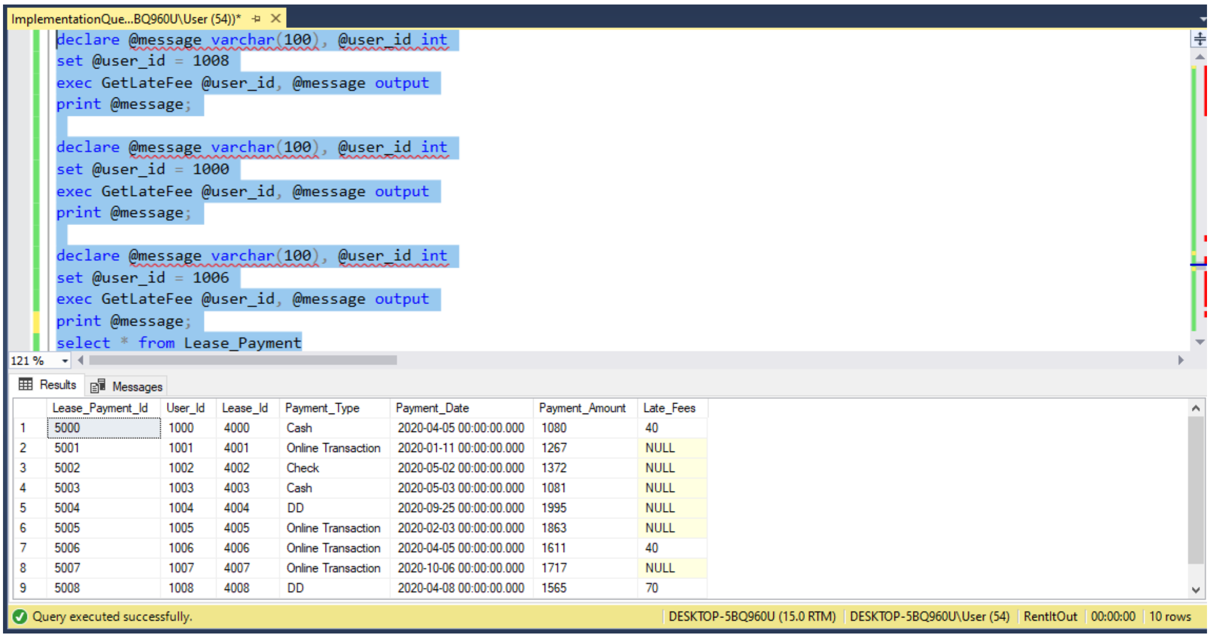
3) Stored Procedure to retrieve lateFee of user if any


Apartment new que...BQ960U\User (51))*
121 %
Messages
121 %
Query executed successfully.
DESKTOP-5BQ960U (15.0 RTM)
DESKTOP-5BQ960U\User (51)
ApartmentRental
00:00:00
0 rows

```sql
                and datepart(day,lp.Payment_Date)<@lastRentDay)
        BEGIN
            UPDATE Lease_Payment set Late_Fees = 0 where Lease_Id= @leaseId;
            SET @message ='No Late Fees Charged,Updated late fee as 0'
        END;
        ELSE IF EXISTS (
        SELECT 1 from Lease_Payment lp where lp.Lease_Id= @leaseId
            and  datepart(day,lp.Payment_Date)>@lastRentDay)
         BEGIN
            UPDATE Lease_Payment set Late_Fees = @lateFees where Lease_Id = @leaseId;
            SET @message ='Late Fees is Updated'
        END;
END
declare @message varchar(100), @user_id int
set @user_id = 1008
exec GetLateFee @user_id, @message output
print @message;
select * from Lease_Payment
```

```
(1 row affected)
Late Fees is Updated

Completion time: 2020-04-07T20:54:33.7127759-04:00
```

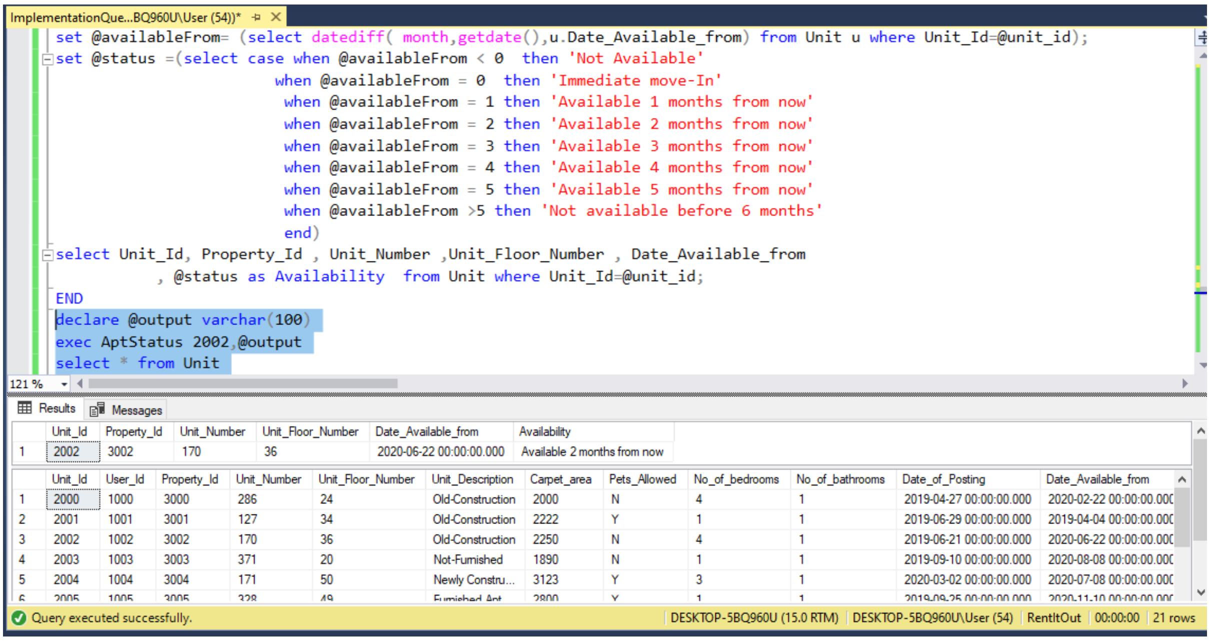
Output-late Fee value updated in the Table


ImplementationQue...BQ960U\User (54))*
121 %
Results
Messages
Query executed successfully.
DESKTOP-5BQ960U (15.0 RTM)
DESKTOP-5BQ960U\User (54)
RentItOut
00:00:00
10 rows

```sql
declare @message varchar(100), @user_id int
set @user_id = 1008
exec GetLateFee @user_id, @message output
print @message;

declare @message varchar(100), @user_id int
set @user_id = 1000
exec GetLateFee @user_id, @message output
print @message;

declare @message varchar(100), @user_id int
set @user_id = 1006
exec GetLateFee @user_id, @message output
print @message;
select * from Lease_Payment
```

	Lease_Payment_Id	User_Id	Lease_Id	Payment_Type	Payment_Date	Payment_Amount	Late_Fees
1	5000	1000	4000	Cash	2020-04-05 00:00:00.000	1080	40
2	5001	1001	4001	Online Transaction	2020-01-11 00:00:00.000	1267	NULL
3	5002	1002	4002	Check	2020-05-02 00:00:00.000	1372	NULL
4	5003	1003	4003	Cash	2020-05-03 00:00:00.000	1081	NULL
5	5004	1004	4004	DD	2020-09-25 00:00:00.000	1995	NULL
6	5005	1005	4005	Online Transaction	2020-02-03 00:00:00.000	1863	NULL
7	5006	1006	4006	Online Transaction	2020-04-05 00:00:00.000	1611	40
8	5007	1007	4007	Online Transaction	2020-10-06 00:00:00.000	1717	NULL
9	5008	1008	4008	DD	2020-04-08 00:00:00.000	1565	70

4) Stored Procedure to Check Availability Status of a Property Unit



ImplementationQue...BQ960U\User (54))*

```sql
set @availableFrom= (select datediff( month,getdate(),u.Date_Available_from) from Unit u where Unit_Id=@unit_id);
set @status =(select case when @availableFrom < 0  then 'Not Available'
                         when @availableFrom = 0  then 'Immediate move-In'
                          when @availableFrom = 1 then 'Available 1 months from now'
                          when @availableFrom = 2 then 'Available 2 months from now'
                          when @availableFrom = 3 then 'Available 3 months from now'
                          when @availableFrom = 4 then 'Available 4 months from now'
                          when @availableFrom = 5 then 'Available 5 months from now'
                          when @availableFrom >5 then 'Not available before 6 months'
                          end)
select Unit_Id, Property_Id , Unit_Number ,Unit_Floor_Number , Date_Available_from
             , @status as Availability  from Unit where Unit_Id=@unit_id;
END
declare @output varchar(100)
exec AptStatus 2002,@output
select * from Unit
```

121 %

Results | Messages

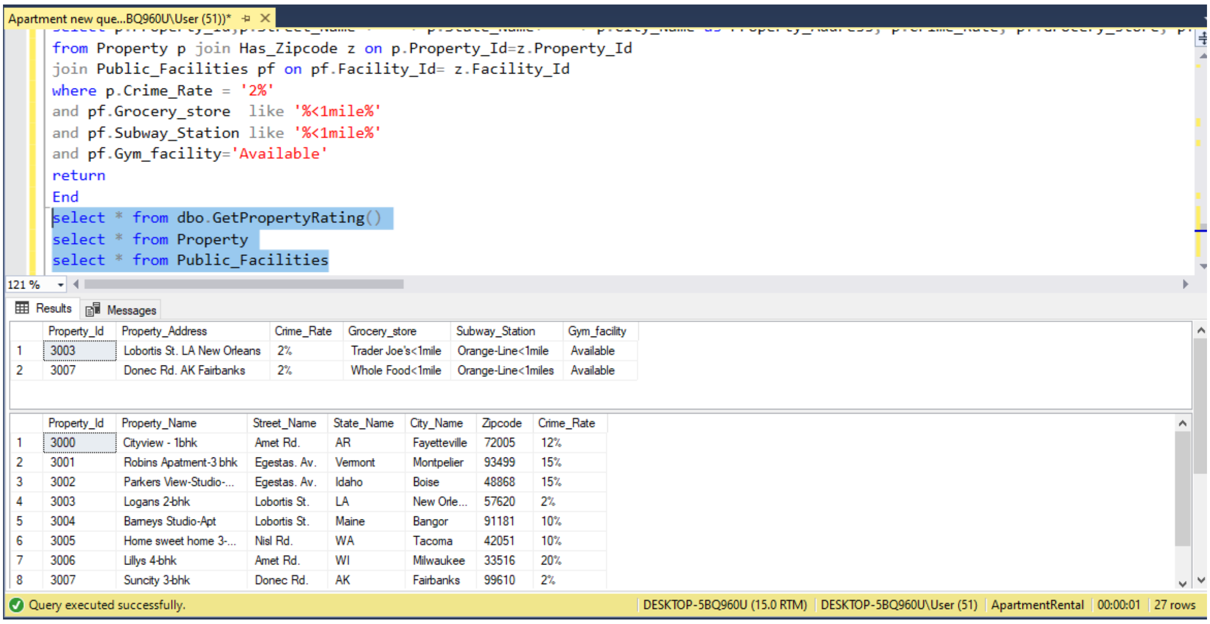
	Unit_Id	Property_Id	Unit_Number	Unit_Floor_Number	Date_Available_from	Availability
1	2002	3002	170	36	2020-06-22 00:00:00.000	Available 2 months from now

	Unit_Id	User_Id	Property_Id	Unit_Number	Unit_Floor_Number	Unit_Description	Carpet_area	Pets_Allowed	No_of_bedrooms	No_of_bathrooms	Date_of_Posting	Date_Available_from
1	2000	1000	3000	286	24	Old-Construction	2000	N	4	1	2019-04-27 00:00:00.000	2020-02-22 00:00:00.000
2	2001	1001	3001	127	34	Old-Construction	2222	Y	1	1	2019-06-29 00:00:00.000	2019-04-04 00:00:00.000
3	2002	1002	3002	170	36	Old-Construction	2250	N	4	1	2019-06-21 00:00:00.000	2020-06-22 00:00:00.000
4	2003	1003	3003	371	20	Not-Furnished	1890	N	1	1	2019-09-10 00:00:00.000	2020-08-08 00:00:00.000
5	2004	1004	3004	171	50	Newly Constru...	3123	Y	3	1	2020-03-02 00:00:00.000	2020-07-08 00:00:00.000
6	2005	1005	3005	328	49	Furnished Apt	2800	Y	1	1	2019-09-25 00:00:00.000	2020-11-10 00:00:00.000

Query executed successfully. | DESKTOP-5BQ960U (15.0 RTM) | DESKTOP-5BQ960U\User (54) | RentItOut | 00:00:00 | 21 rows

FUNCTIONS:

1) Function to retrieve best property in terms of crime rate and facilities available



Apartment new que...BQ960U\User (51))*

```sql
select p.Property_Id,p.Street_Name + ' ' + p.State_Name+ ' ' + p.City_Name as Property_Address, p.Crime_Rate, p.Grocery_store, p.
from Property p join Has_Zipcode z on p.Property_Id=z.Property_Id
join Public_Facilities pf on pf.Facility_Id= z.Facility_Id
where p.Crime_Rate = '2%'
and pf.Grocery_store  like '%<1mile%'
and pf.Subway_Station like '%<1mile%'
and pf.Gym_facility='Available'
return
End
select * from dbo.GetPropertyRating()
select * from Property
select * from Public_Facilities
```

121 %

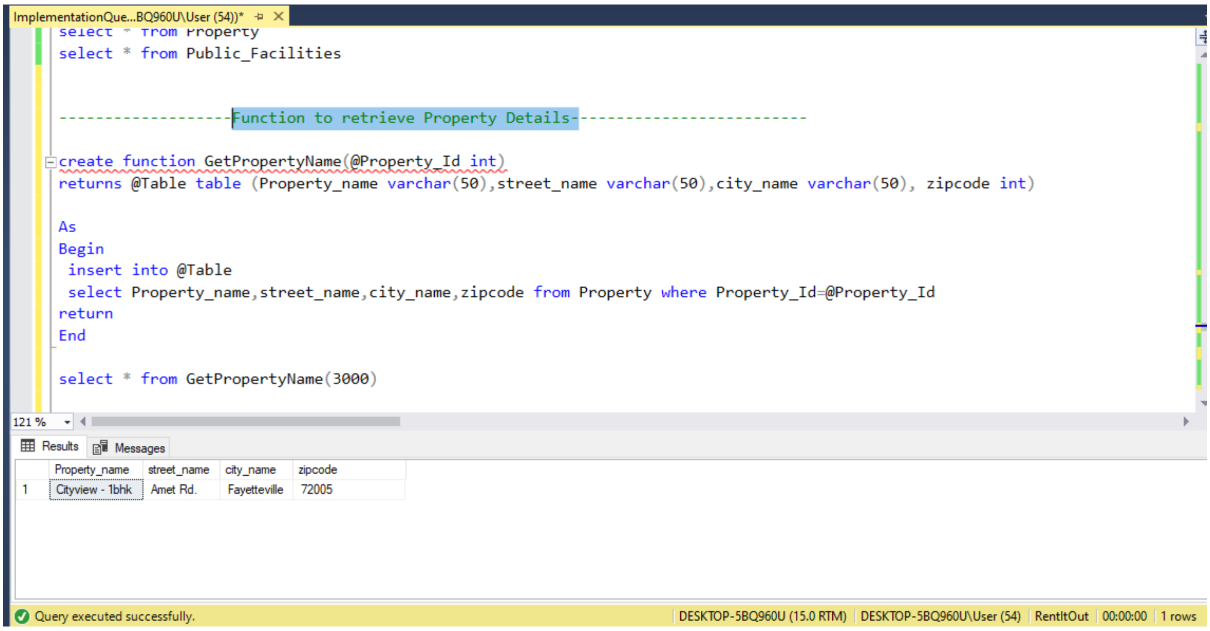
Results | Messages

	Property_Id	Property_Address	Crime_Rate	Grocery_store	Subway_Station	Gym_facility
1	3003	Lobortis St. LA New Orleans	2%	Trader Joe's<1mile	Orange-Line<1mile	Available
2	3007	Donec Rd. AK Fairbanks	2%	Whole Food<1mile	Orange-Line<1miles	Available

	Property_Id	Property_Name	Street_Name	State_Name	City_Name	Zipcode	Crime_Rate
1	3000	Cityview - 1bhk	Amet Rd.	AR	Fayetteville	72005	12%
2	3001	Robins Apatment-3 bhk	Egestas. Av.	Vermont	Montpelier	93499	15%
3	3002	Parkers View-Studio-...	Egestas. Av.	Idaho	Boise	48868	15%
4	3003	Logans 2-bhk	Lobortis St.	LA	New Orle...	57620	2%
5	3004	Barneys Studio-Apt	Lobortis St.	Maine	Bangor	91181	10%
6	3005	Home sweet home 3-...	Nisl Rd.	WA	Tacoma	42051	10%
7	3006	Lillys 4-bhk	Amet Rd.	WI	Milwaukee	33516	20%
8	3007	Suncity 3-bhk	Donec Rd.	AK	Fairbanks	99610	2%

Query executed successfully. | DESKTOP-5BQ960U (15.0 RTM) | DESKTOP-5BQ960U\User (51) | ApartmentRental | 00:00:01 | 27 rows

2) Function to retrieve Property Details-



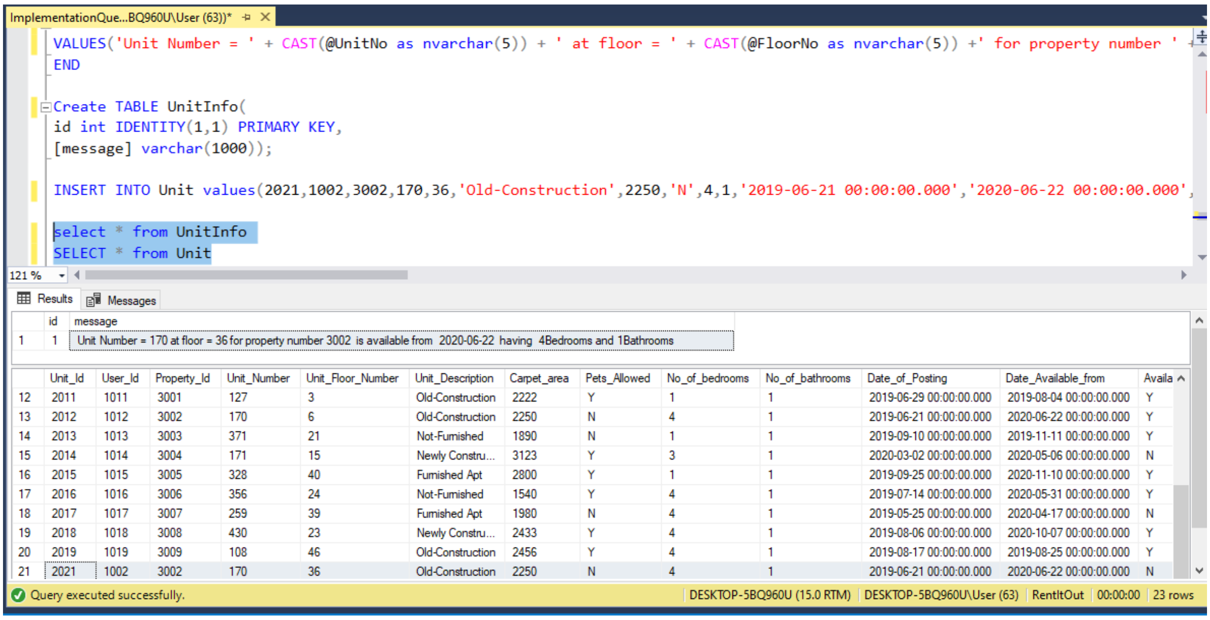
```
select * from Property
select * from Public_Facilities


------------------Function to retrieve Property Details-------------------------

create function GetPropertyName(@Property_Id int)
returns @Table table (Property_name varchar(50),street_name varchar(50),city_name varchar(50), zipcode int)

As
Begin
 insert into @Table
 select Property_name,street_name,city_name,zipcode from Property where Property_Id=@Property_Id
return
End

select * from GetPropertyName(3000)
```

	Property_name	street_name	city_name	zipcode
1	Cityview - 1bhk	Amet Rd.	Fayetteville	72005

Query executed successfully. | DESKTOP-5BQ960U (15.0 RTM) | DESKTOP-5BQ960U\User (54) | RentItOut | 00:00:00 | 1 rows

Triggers

1) Trigger to track availability of a Unit when posted



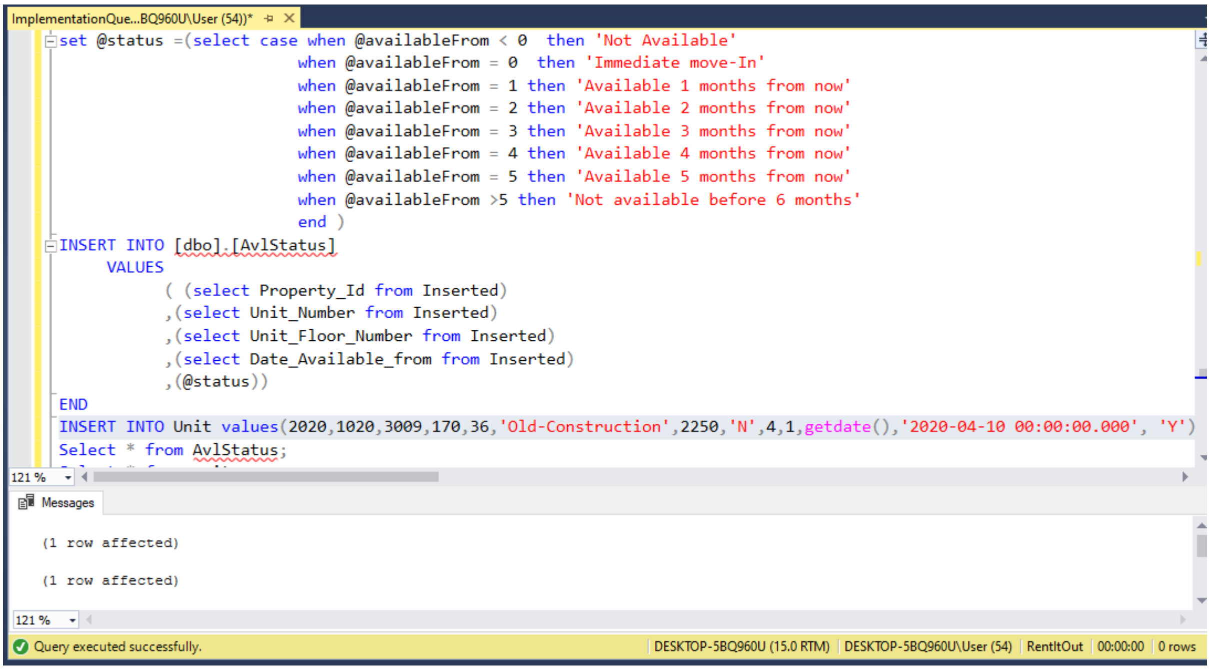
```
VALUES('Unit Number = ' + CAST(@UnitNo as nvarchar(5)) + ' at floor = ' + CAST(@FloorNo as nvarchar(5)) +' for property number ' +
END

Create TABLE UnitInfo(
id int IDENTITY(1,1) PRIMARY KEY,
[message] varchar(1000));

INSERT INTO Unit values(2021,1002,3002,170,36,'Old-Construction',2250,'N',4,1,'2019-06-21 00:00:00.000','2020-06-22 00:00:00.000',

select * from UnitInfo
SELECT * from Unit
```

	id	message
1	1	Unit Number = 170 at floor = 36 for property number 3002 is available from 2020-06-22 having 4Bedrooms and 1Bathrooms

	Unit_Id	User_Id	Property_Id	Unit_Number	Unit_Floor_Number	Unit_Description	Carpet_area	Pets_Allowed	No_of_bedrooms	No_of_bathrooms	Date_of_Posting	Date_Available_from	Availa
12	2011	1011	3001	127	3	Old-Construction	2222	Y	1	1	2019-06-29 00:00:00.000	2019-08-04 00:00:00.000	Y
13	2012	1012	3002	170	6	Old-Construction	2250	N	4	1	2019-06-21 00:00:00.000	2020-06-22 00:00:00.000	Y
14	2013	1013	3003	371	21	Not-Furnished	1890	N	1	1	2019-09-10 00:00:00.000	2019-11-11 00:00:00.000	Y
15	2014	1014	3004	171	15	Newly Constru...	3123	Y	3	1	2020-03-02 00:00:00.000	2020-05-06 00:00:00.000	N
16	2015	1015	3005	328	40	Furnished Apt	2800	Y	1	1	2019-09-25 00:00:00.000	2020-11-10 00:00:00.000	Y
17	2016	1016	3006	356	24	Not-Furnished	1540	Y	4	1	2019-07-14 00:00:00.000	2020-05-31 00:00:00.000	Y
18	2017	1017	3007	259	39	Furnished Apt	1980	N	4	1	2019-05-25 00:00:00.000	2020-04-17 00:00:00.000	N
19	2018	1018	3008	430	23	Newly Constru...	2433	Y	4	1	2019-08-06 00:00:00.000	2020-10-07 00:00:00.000	Y
20	2019	1019	3009	108	46	Old-Construction	2456	Y	4	1	2019-08-17 00:00:00.000	2019-08-25 00:00:00.000	Y
21	2021	1002	3002	170	36	Old-Construction	2250	N	4	1	2019-06-21 00:00:00.000	2020-06-22 00:00:00.000	N

Query executed successfully. | DESKTOP-5BQ960U (15.0 RTM) | DESKTOP-5BQ960U\User (63) | RentItOut | 00:00:00 | 23 rows

2) Trigger to track Availability status of Apartments

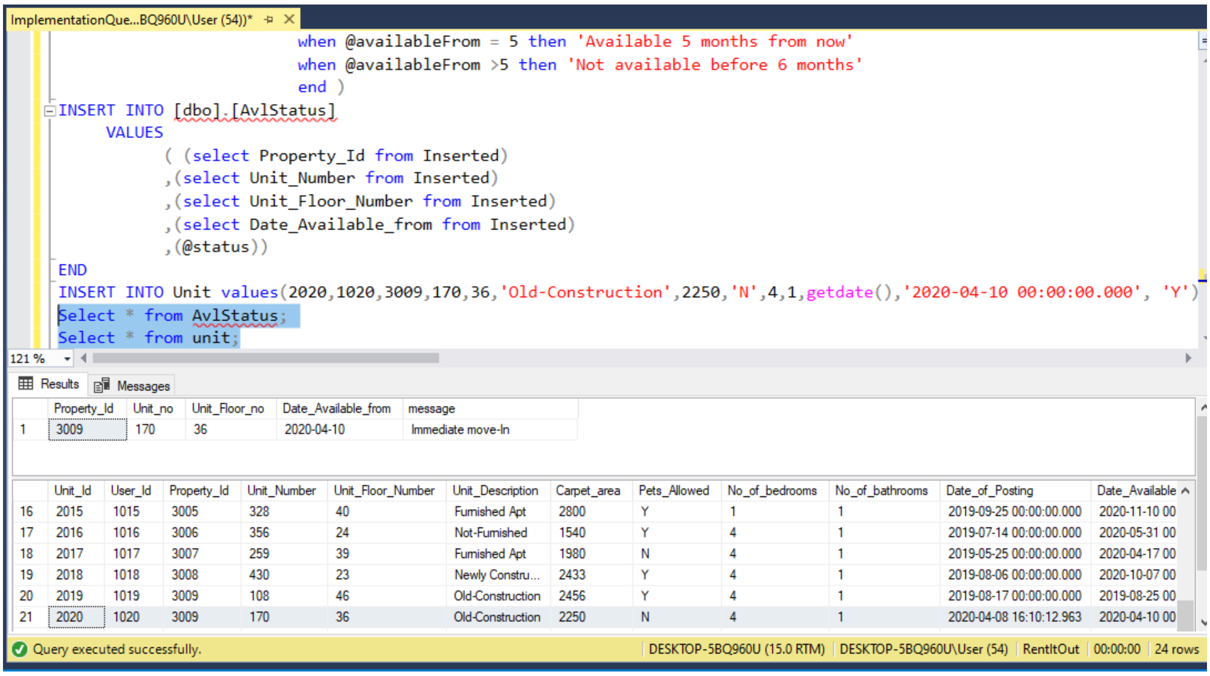


```
set @status =(select case when @availableFrom < 0  then 'Not Available'
                          when @availableFrom = 0  then 'Immediate move-In'
                          when @availableFrom = 1 then 'Available 1 months from now'
                          when @availableFrom = 2 then 'Available 2 months from now'
                          when @availableFrom = 3 then 'Available 3 months from now'
                          when @availableFrom = 4 then 'Available 4 months from now'
                          when @availableFrom = 5 then 'Available 5 months from now'
                          when @availableFrom >5 then 'Not available before 6 months'
                          end )
INSERT INTO [dbo].[AvlStatus]
     VALUES
           ( (select Property_Id from Inserted)
           ,(select Unit_Number from Inserted)
           ,(select Unit_Floor_Number from Inserted)
           ,(select Date_Available_from from Inserted)
           ,(@status))
END
INSERT INTO Unit values(2020,1020,3009,170,36,'Old-Construction',2250,'N',4,1,getdate(),'2020-04-10 00:00:00.000', 'Y')
Select * from AvlStatus;
```

```
(1 row affected)

(1 row affected)
```

```
Query executed successfully.    DESKTOP-5BQ960U (15.0 RTM) | DESKTOP-5BQ960U\User (54) | RentItOut | 00:00:00 | 0 rows
```

Output-



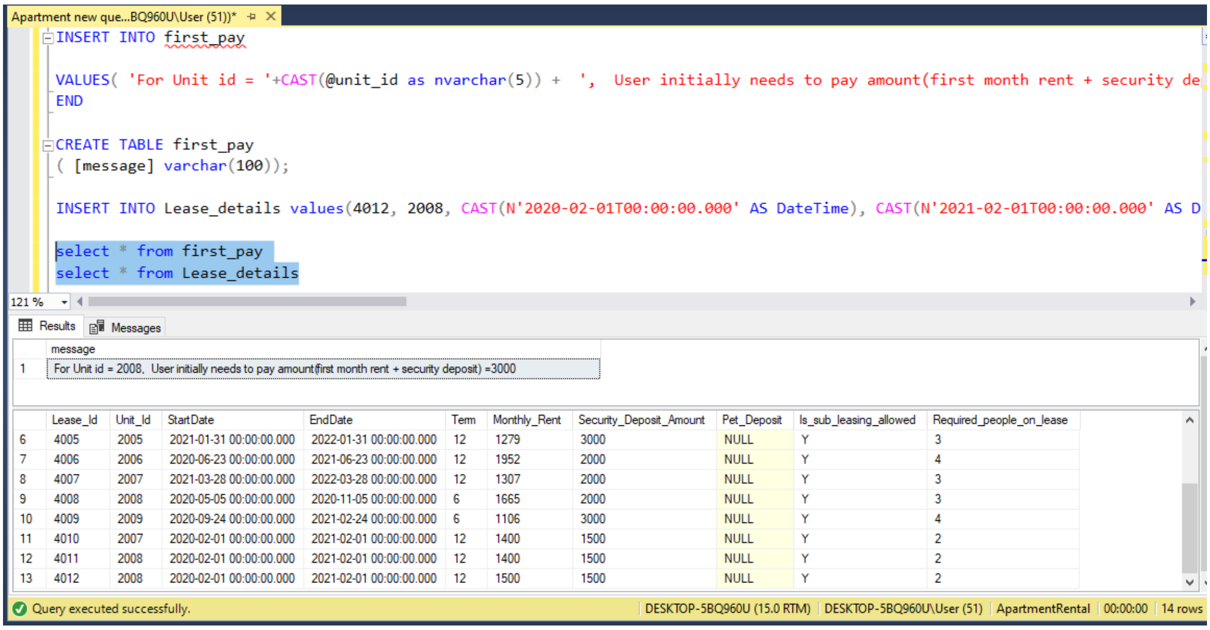
```
                          when @availableFrom = 5 then 'Available 5 months from now'
                          when @availableFrom >5 then 'Not available before 6 months'
                          end )
INSERT INTO [dbo].[AvlStatus]
     VALUES
           ( (select Property_Id from Inserted)
           ,(select Unit_Number from Inserted)
           ,(select Unit_Floor_Number from Inserted)
           ,(select Date_Available_from from Inserted)
           ,(@status))
END
INSERT INTO Unit values(2020,1020,3009,170,36,'Old-Construction',2250,'N',4,1,getdate(),'2020-04-10 00:00:00.000', 'Y')
Select * from AvlStatus;
Select * from unit;
```

	Property_Id	Unit_no	Unit_Floor_no	Date_Available_from	message
1	3009	170	36	2020-04-10	Immediate move-In

	Unit_Id	User_Id	Property_Id	Unit_Number	Unit_Floor_Number	Unit_Description	Carpet_area	Pets_Allowed	No_of_bedrooms	No_of_bathrooms	Date_of_Posting	Date_Available
16	2015	1015	3005	328	40	Furnished Apt	2800	Y	1	1	2019-09-25 00:00:00.000	2020-11-10 00
17	2016	1016	3006	356	24	Not-Furnished	1540	Y	4	1	2019-07-14 00:00:00.000	2020-05-31 00
18	2017	1017	3007	259	39	Furnished Apt	1980	N	4	1	2019-05-25 00:00:00.000	2020-04-17 00
19	2018	1018	3008	430	23	Newly Constru...	2433	Y	4	1	2019-08-06 00:00:00.000	2020-10-07 00
20	2019	1019	3009	108	46	Old-Construction	2456	Y	4	1	2019-08-17 00:00:00.000	2019-08-25 00
21	2020	1020	3009	170	36	Old-Construction	2250	N	4	1	2020-04-08 16:10:12.963	2020-04-10 00

```
Query executed successfully.    DESKTOP-5BQ960U (15.0 RTM) | DESKTOP-5BQ960U\User (54) | RentItOut | 00:00:00 | 24 rows
```

3) Trigger to generate first payment(monthlyrent +securitydeposit) required by the user for each unit



Apartment new que...BQ960U\User (51))*

```
INSERT INTO first_pay

VALUES( 'For Unit id = '+CAST(@unit_id as nvarchar(5)) +  ',  User initially needs to pay amount(first month rent + security de
END

CREATE TABLE first_pay
( [message] varchar(100));

INSERT INTO Lease_details values(4012, 2008, CAST(N'2020-02-01T00:00:00.000' AS DateTime), CAST(N'2021-02-01T00:00:00.000' AS D

select * from first_pay
select * from Lease_details
```

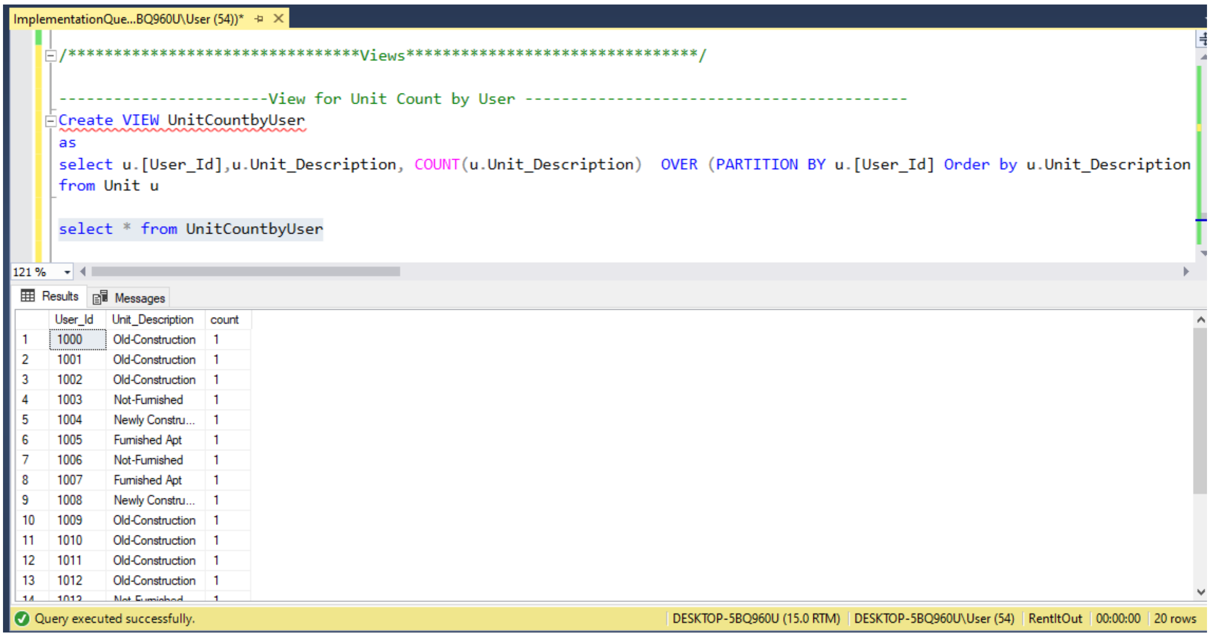
	message
1	For Unit id = 2008, User initially needs to pay amount(first month rent + security deposit) =3000

	Lease_Id	Unit_Id	StartDate	EndDate	Term	Monthly_Rent	Security_Deposit_Amount	Pet_Deposit	Is_sub_leasing_allowed	Required_people_on_lease
6	4005	2005	2021-01-31 00:00:00.000	2022-01-31 00:00:00.000	12	1279	3000	NULL	Y	3
7	4006	2006	2020-06-23 00:00:00.000	2021-06-23 00:00:00.000	12	1952	2000	NULL	Y	4
8	4007	2007	2021-03-28 00:00:00.000	2022-03-28 00:00:00.000	12	1307	2000	NULL	Y	3
9	4008	2008	2020-05-05 00:00:00.000	2020-11-05 00:00:00.000	6	1665	2000	NULL	Y	3
10	4009	2009	2020-09-24 00:00:00.000	2021-02-24 00:00:00.000	6	1106	3000	NULL	Y	4
11	4010	2007	2020-02-01 00:00:00.000	2021-02-01 00:00:00.000	12	1400	1500	NULL	Y	2
12	4011	2008	2020-02-01 00:00:00.000	2021-02-01 00:00:00.000	12	1400	1500	NULL	Y	2
13	4012	2008	2020-02-01 00:00:00.000	2021-02-01 00:00:00.000	12	1500	1500	NULL	Y	2

Query executed successfully. | DESKTOP-5BQ960U (15.0 RTM) | DESKTOP-5BQ960U\User (51) | ApartmentRental | 00:00:00 | 14 rows

Views -

1) View for Unit Count by User



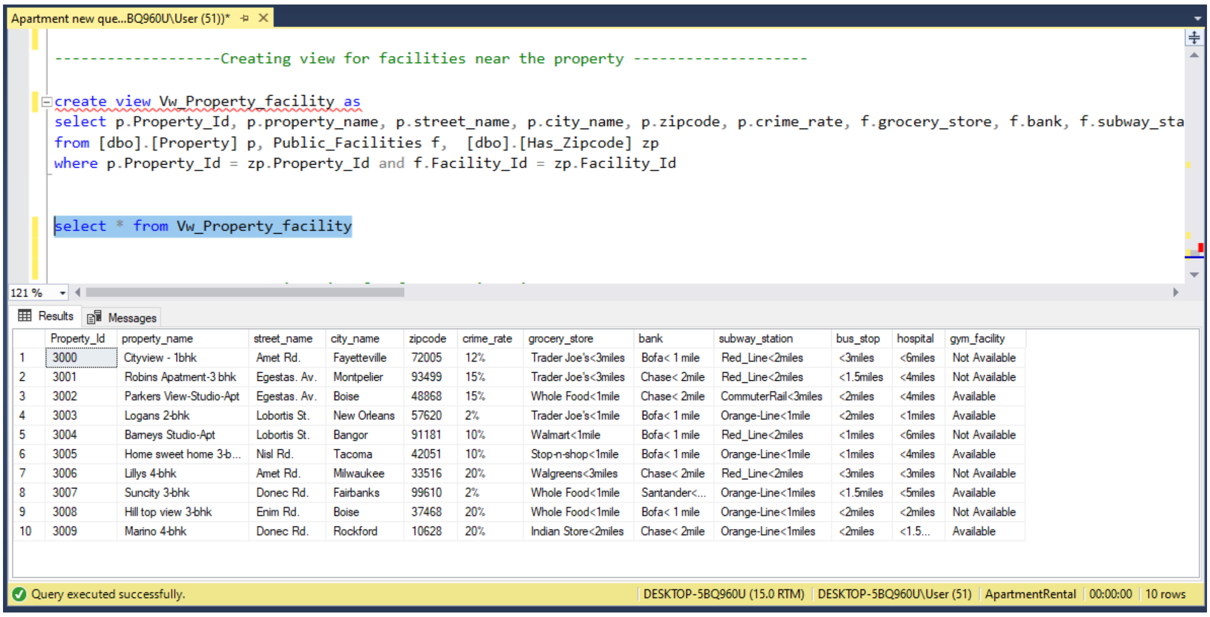
ImplementationQue...BQ960U\User (54))*

```
/*******************************Views*******************************/

----------------------View for Unit Count by User -----------------------------------------
Create VIEW UnitCountbyUser
as
select u.[User_Id],u.Unit_Description, COUNT(u.Unit_Description)  OVER (PARTITION BY u.[User_Id] Order by u.Unit_Description
from Unit u

select * from UnitCountbyUser
```

	User_Id	Unit_Description	count
1	1000	Old-Construction	1
2	1001	Old-Construction	1
3	1002	Old-Construction	1
4	1003	Not-Fumished	1
5	1004	Newly Constru...	1
6	1005	Fumished Apt	1
7	1006	Not-Fumished	1
8	1007	Fumished Apt	1
9	1008	Newly Constru...	1
10	1009	Old-Construction	1
11	1010	Old-Construction	1
12	1011	Old-Construction	1
13	1012	Old-Construction	1
14	1013	Not-Fumished	1

Query executed successfully. | DESKTOP-5BQ960U (15.0 RTM) | DESKTOP-5BQ960U\User (54) | RentItOut | 00:00:00 | 20 rows

2) View showing Facilities Available near the Property

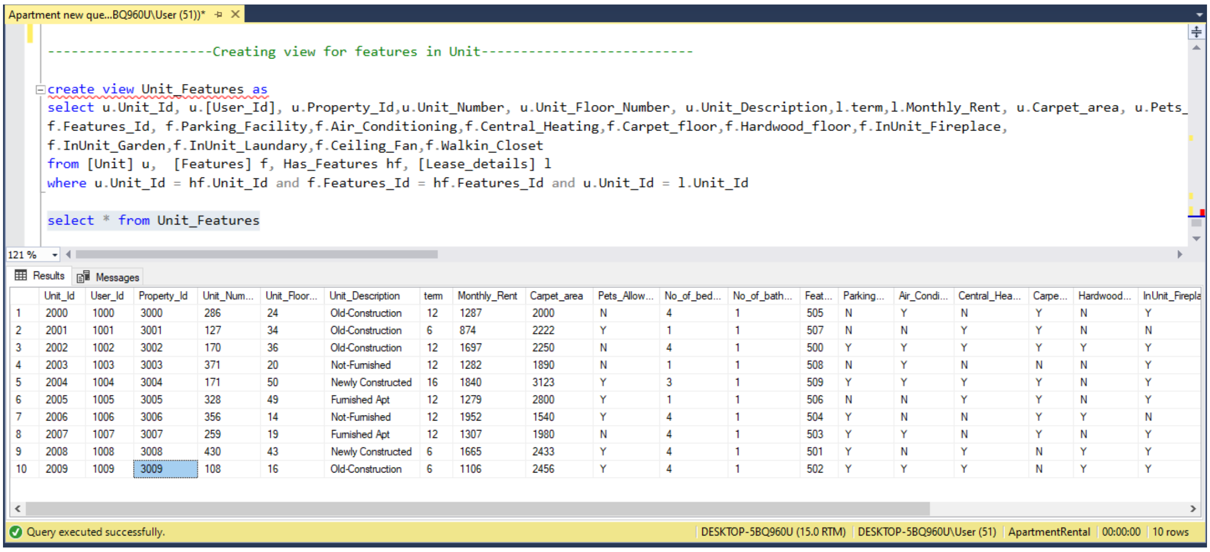


```
-------------------Creating view for facilities near the property --------------------

create view Vw_Property_facility as
select p.Property_Id, p.property_name, p.street_name, p.city_name, p.zipcode, p.crime_rate, f.grocery_store, f.bank, f.subway_sta
from [dbo].[Property] p, Public_Facilities f,  [dbo].[Has_Zipcode] zp
where p.Property_Id = zp.Property_Id and f.Facility_Id = zp.Facility_Id


select * from Vw_Property_facility
```

	Property_Id	property_name	street_name	city_name	zipcode	crime_rate	grocery_store	bank	subway_station	bus_stop	hospital	gym_facility
1	3000	Cityview - 1bhk	Amet Rd.	Fayetteville	72005	12%	Trader Joe's<3miles	Bofa< 1 mile	Red_Line<2miles	<3miles	<6miles	Not Available
2	3001	Robins Apatment-3 bhk	Egestas. Av.	Montpelier	93499	15%	Trader Joe's<3miles	Chase< 2mile	Red_Line<2miles	<1.5miles	<4miles	Not Available
3	3002	Parkers View-Studio-Apt	Egestas. Av.	Boise	48868	15%	Whole Food<1mile	Chase< 2mile	CommuterRail<3miles	<2miles	<4miles	Available
4	3003	Logans 2-bhk	Lobortis St.	New Orleans	57620	2%	Trader Joe's<1mile	Bofa< 1 mile	Orange-Line<1mile	<2miles	<1miles	Available
5	3004	Bameys Studio-Apt	Lobortis St.	Bangor	91181	10%	Walmart<1mile	Bofa< 1 mile	Red_Line<2miles	<1miles	<6miles	Not Available
6	3005	Home sweet home 3-b...	Nisl Rd.	Tacoma	42051	10%	Stop-n-shop<1mile	Bofa< 1 mile	Orange-Line<1mile	<1miles	<4miles	Available
7	3006	Lillys 4-bhk	Amet Rd.	Milwaukee	33516	20%	Walgreens<3miles	Chase< 2mile	Red_Line<2miles	<3miles	<3miles	Not Available
8	3007	Suncity 3-bhk	Donec Rd.	Fairbanks	99610	2%	Whole Food<1mile	Santander<...	Orange-Line<1miles	<1.5miles	<5miles	Available
9	3008	Hill top view 3-bhk	Enim Rd.	Boise	37468	20%	Whole Food<1mile	Bofa< 1 mile	Orange-Line<1miles	<2miles	<2miles	Not Available
10	3009	Marino 4-bhk	Donec Rd.	Rockford	10628	20%	Indian Store<2miles	Chase< 2mile	Orange-Line<1miles	<2miles	<1.5...	Available

Query executed successfully. | DESKTOP-5BQ960U (15.0 RTM) | DESKTOP-5BQ960U\User (51) | ApartmentRental | 00:00:00 | 10 rows

3) View showing Unit Features



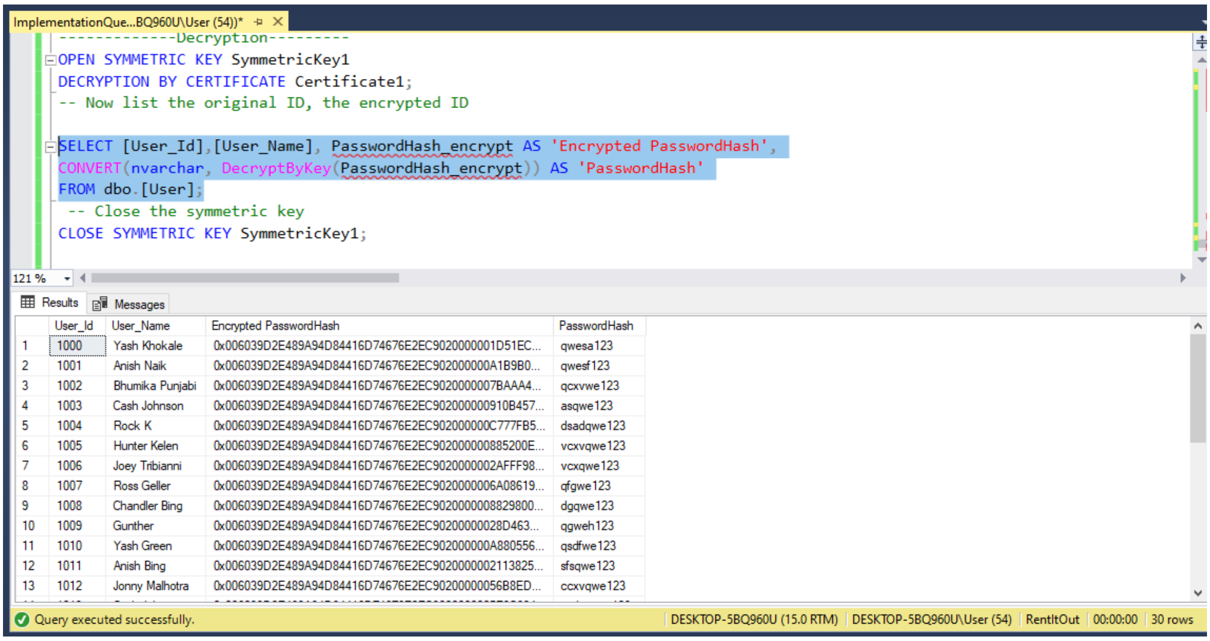
```
--------------------Creating view for features in Unit--------------------------

create view Unit_Features as
select u.Unit_Id, u.[User_Id], u.Property_Id,u.Unit_Number, u.Unit_Floor_Number, u.Unit_Description,l.term,l.Monthly_Rent, u.Carpet_area, u.Pets_
f.Features_Id, f.Parking_Facility,f.Air_Conditioning,f.Central_Heating,f.Carpet_floor,f.Hardwood_floor,f.InUnit_Fireplace,
f.InUnit_Garden,f.InUnit_Laundary,f.Ceiling_Fan,f.Walkin_Closet
from [Unit] u,  [Features] f, Has_Features hf, [Lease_details] l
where u.Unit_Id = hf.Unit_Id and f.Features_Id = hf.Features_Id and u.Unit_Id = l.Unit_Id

select * from Unit_Features
```

	Unit_Id	User_Id	Property_Id	Unit_Num...	Unit_Floor...	Unit_Description	term	Monthly_Rent	Carpet_area	Pets_Allow...	No_of_bed...	No_of_bath...	Feat...	Parking...	Air_Condi...	Central_Hea...	Carpe...	Hardwood...	InUnit_Firepla
1	2000	1000	3000	286	24	Old-Construction	12	1287	2000	N	4	1	505	N	Y	N	Y	N	Y
2	2001	1001	3001	127	34	Old-Construction	6	874	2222	Y	1	1	507	N	N	Y	Y	N	N
3	2002	1002	3002	170	36	Old-Construction	12	1697	2250	N	4	1	500	Y	Y	Y	Y	Y	Y
4	2003	1003	3003	371	20	Not-Fumished	12	1282	1890	N	1	1	508	N	Y	N	N	N	Y
5	2004	1004	3004	171	50	Newly Constructed	16	1840	3123	Y	3	1	509	Y	Y	Y	Y	N	Y
6	2005	1005	3005	328	49	Fumished Apt	12	1279	2800	Y	1	1	506	N	N	Y	Y	N	Y
7	2006	1006	3006	356	14	Not-Fumished	12	1952	1540	Y	4	1	504	Y	N	N	Y	Y	N
8	2007	1007	3007	259	19	Fumished Apt	12	1307	1980	N	4	1	503	Y	Y	N	Y	N	Y
9	2008	1008	3008	430	43	Newly Constructed	6	1665	2433	Y	4	1	501	Y	N	Y	N	Y	Y
10	2009	1009	3009	108	16	Old-Construction	6	1106	2456	Y	4	1	502	Y	Y	Y	N	Y	Y

Query executed successfully. | DESKTOP-5BQ960U (15.0 RTM) | DESKTOP-5BQ960U\User (51) | ApartmentRental | 00:00:00 | 10 rows

Computed Data Encryption

Column Data Encryption on **User’s Password**



```
-------------Decryption---------
OPEN SYMMETRIC KEY SymmetricKey1
DECRYPTION BY CERTIFICATE Certificate1;
-- Now list the original ID, the encrypted ID

SELECT [User_Id],[User_Name], PasswordHash_encrypt AS 'Encrypted PasswordHash',
CONVERT(nvarchar, DecryptByKey(PasswordHash_encrypt)) AS 'PasswordHash'
FROM dbo.[User];
 -- Close the symmetric key
CLOSE SYMMETRIC KEY SymmetricKey1;
```

	User_Id	User_Name	Encrypted PasswordHash	PasswordHash
1	1000	Yash Khokale	0x006039D2E489A94D84416D74676E2EC9020000001D51EC...	qwesa123
2	1001	Anish Naik	0x006039D2E489A94D84416D74676E2EC902000000A1B9B0...	qwesf123
3	1002	Bhumika Punjabi	0x006039D2E489A94D84416D74676E2EC9020000007BAAA4...	qcxvwe123
4	1003	Cash Johnson	0x006039D2E489A94D84416D74676E2EC90200000910B457...	asqwe123
5	1004	Rock K	0x006039D2E489A94D84416D74676E2EC902000000C777FB5...	dsadqwe123
6	1005	Hunter Kelen	0x006039D2E489A94D84416D74676E2EC90200000885200E...	vcxvqwe123
7	1006	Joey Tribianni	0x006039D2E489A94D84416D74676E2EC9020000002AFFF98...	vcxqwe123
8	1007	Ross Geller	0x006039D2E489A94D84416D74676E2EC9020000006A08619...	qfgwe123
9	1008	Chandler Bing	0x006039D2E489A94D84416D74676E2EC9020000008829800...	dgqwe123
10	1009	Gunther	0x006039D2E489A94D84416D74676E2EC90200000028D463...	qgweh123
11	1010	Yash Green	0x006039D2E489A94D84416D74676E2EC902000000A880556...	qsdfwe123
12	1011	Anish Bing	0x006039D2E489A94D84416D74676E2EC9020000002113825...	sfsqwe123
13	1012	Jonny Malhotra	0x006039D2E489A94D84416D74676E2EC90200000056B8ED...	ccxvqwe123

Query executed successfully. | DESKTOP-5BQ960U (15.0 RTM) | DESKTOP-5BQ960U\User (54) | RentItOut | 00:00:00 | 30 rows