

NAME:SURYAWANSHI RANDHEER AJIT

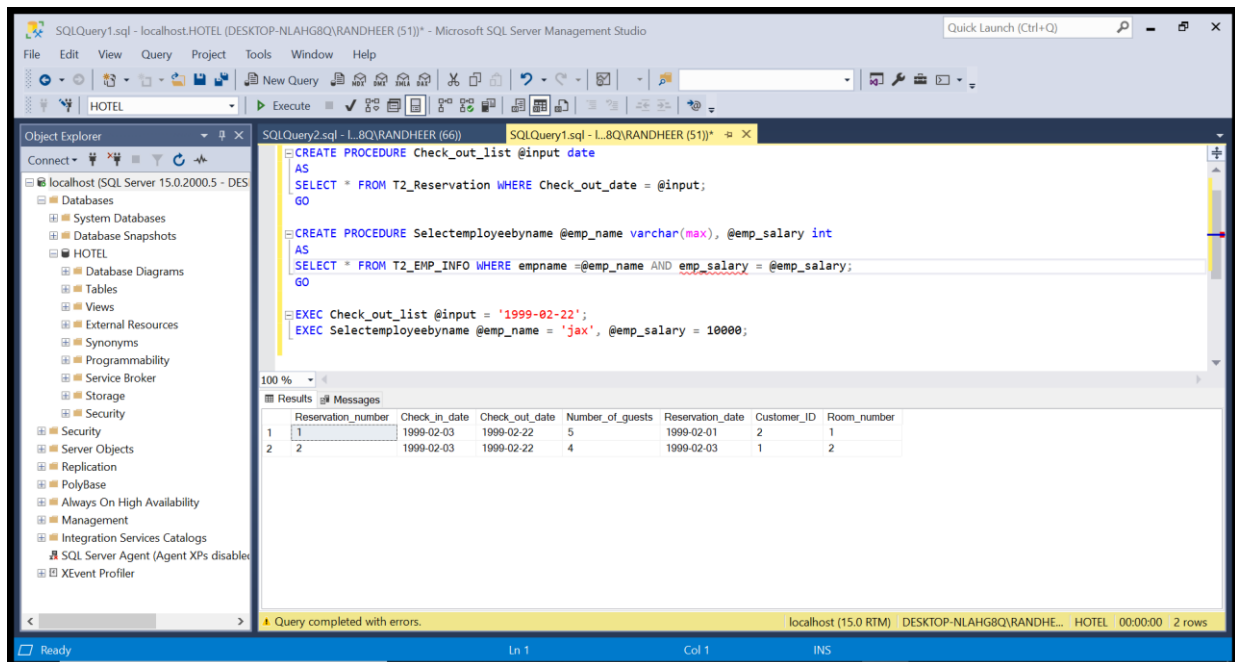
REG NO:19BCS104

DBMS LAB ASSGMENT 7

GROUP NUMBER: 2(HOTEL DATABASE)

Q1) Write two stored Procedures relevant to your database.

OUTPUT:



```
CREATE PROCEDURE Check_out_list @input date
AS
SELECT * FROM T2_Reservation WHERE Check_out_date = @input;
GO

CREATE PROCEDURE Selectemployeebyname @emp_name varchar(max), @emp_salary int
AS
SELECT * FROM T2_EMP_INFO WHERE empname =@emp_name AND emp_salary = @emp_salary;
GO

EXEC Check_out_list @input = '1999-02-22';
EXEC Selectemployeebyname @emp_name = 'jax', @emp_salary = 10000;
```

Reservation_number	Check_in_date	Check_out_date	Number_of_guests	Reservation_date	Customer_ID	Room_number
1	1999-02-03	1999-02-22	5	1999-02-01	2	1
2	1999-02-03	1999-02-22	4	1999-02-03	1	2

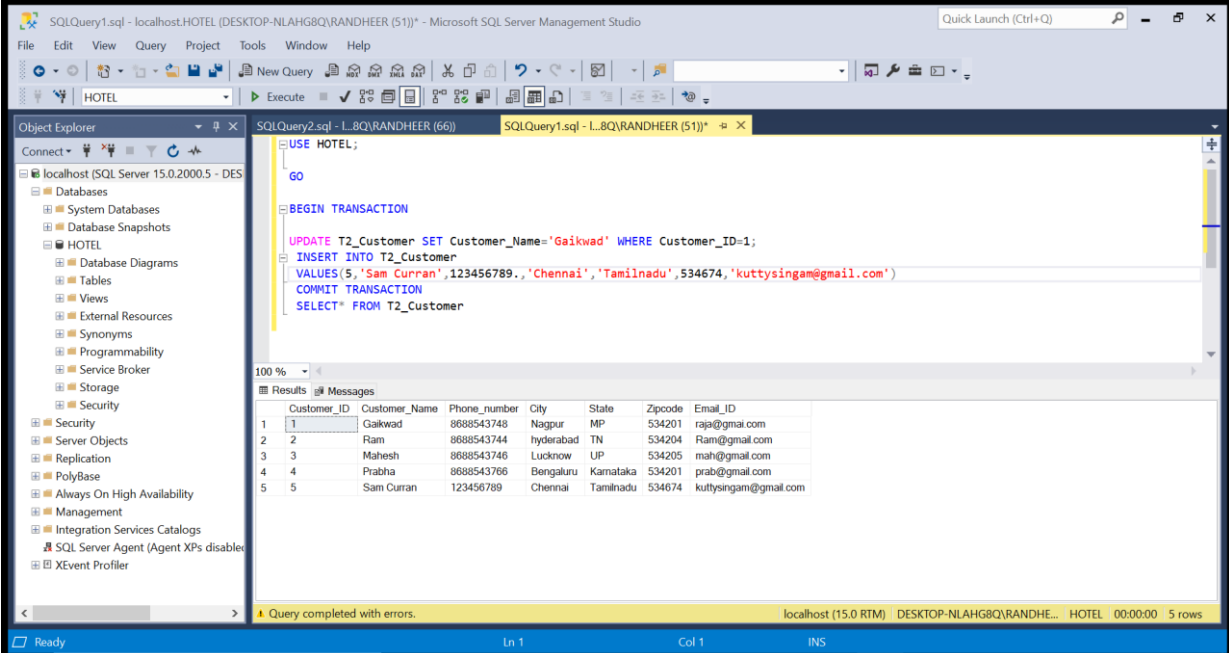
Q2) Write a transaction to illustrate atomicity (related to your database)

T2_Customer table before transaction:

Customer_ID	Customer_Name	Phone_number	City	State	Zipcode	Email_ID
1	Loflin	8688543748	Nagpur	MP	534201	loff@gmail.com
2	Ram	8688543744	Hyderabad	TN	534204	Ram@gmail.com
3	Mahesh	8688543746	Lucknow	UP	534205	mah@gmail.com
4	Prabha	8688543766	Bengaluru	Karnataka	534201	prab@gmail.com

Now we will start the transaction to check the atomicity

OUTPUT AFTER TRANSACTION IS SUCCESSFUL:



SQLQuery1.sql - localhost:HOTEL (DESKTOP-NLAHG8Q\RANDHEER (51)) - Microsoft SQL Server Management Studio

Object Explorer: Connect -> localhost (SQL Server 15.0.2000.5 - DES) -> Databases -> HOTEL -> Tables -> T2_Customer

SQLQuery1.sql - L-8Q\RANDHEER (51) *

```
USE HOTEL;
GO
BEGIN TRANSACTION
UPDATE T2_Customer SET Customer_Name='Gaikwad' WHERE Customer_ID=1;
INSERT INTO T2_Customer
VALUES(5,'Sam Curran',123456789,'Chennai','Tamilnadu',534674,'kuttysingam@gmail.com')
COMMIT TRANSACTION
SELECT* FROM T2_Customer
```

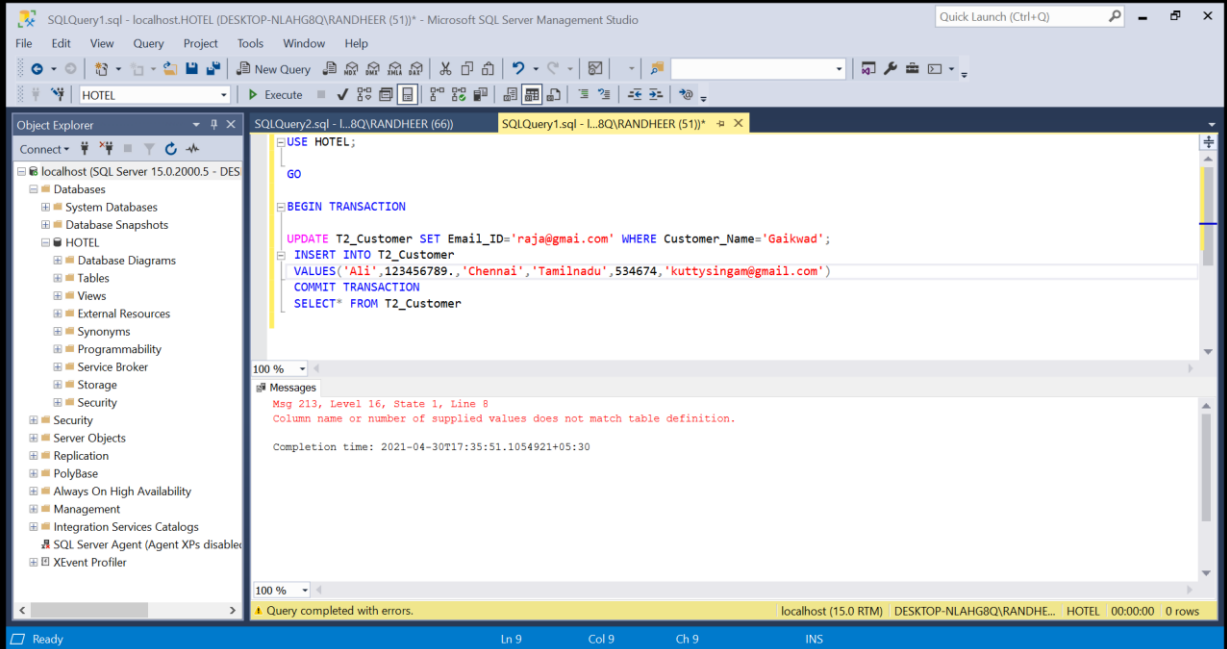
Results

Customer_ID	Customer_Name	Phone_number	City	State	Zipcode	Email_ID
1	Gaikwad	8688543748	Nagpur	MP	534201	raja@gmail.com
2	Ram	8688543744	hyderabad	TN	534204	Ram@gmail.com
3	Maresh	8688543746	Lucknow	UP	534205	mah@gmail.com
4	Prabha	8688543766	Bengaluru	Karnataka	534201	prab@gmail.com
5	Sam Curran	123456789	Chennai	Tamilnadu	534674	kuttysingam@gmail.com

Query completed with errors. localhost (15.0 RTM) DESKTOP-NLAHG8Q\RANDHE... HOTEL 00:00:00 5 rows

This time we will insert wrong information in the Customers table to fail the insertion deliberately.

OUTPUT AFTER TRANSACTION:



SQLQuery1.sql - localhost:HOTEL (DESKTOP-NLAHG8Q\RANDHEER (51)) - Microsoft SQL Server Management Studio

Object Explorer: Connect -> localhost (SQL Server 15.0.2000.5 - DES) -> Databases -> HOTEL -> Tables -> T2_Customer

SQLQuery1.sql - L-8Q\RANDHEER (51) *

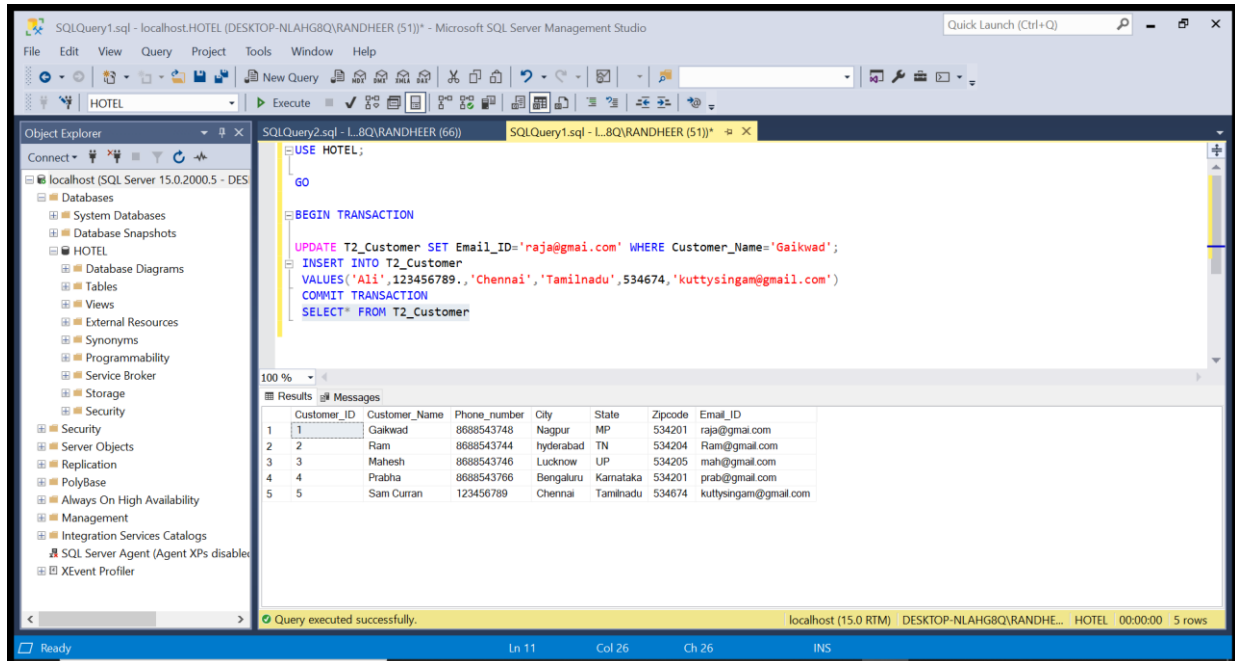
```
USE HOTEL;
GO
BEGIN TRANSACTION
UPDATE T2_Customer SET Email_ID='raja@gmail.com' WHERE Customer_Name='Gaikwad';
INSERT INTO T2_Customer
VALUES('Ali',123456789,'Chennai','Tamilnadu',534674,'kuttysingam@gmail.com')
COMMIT TRANSACTION
SELECT* FROM T2_Customer
```

Messages

Msg 213, Level 16, State 1, Line 8
Column name or number of supplied values does not match table definition.

Completion time: 2021-04-30T17:35:51.1054921+05:30

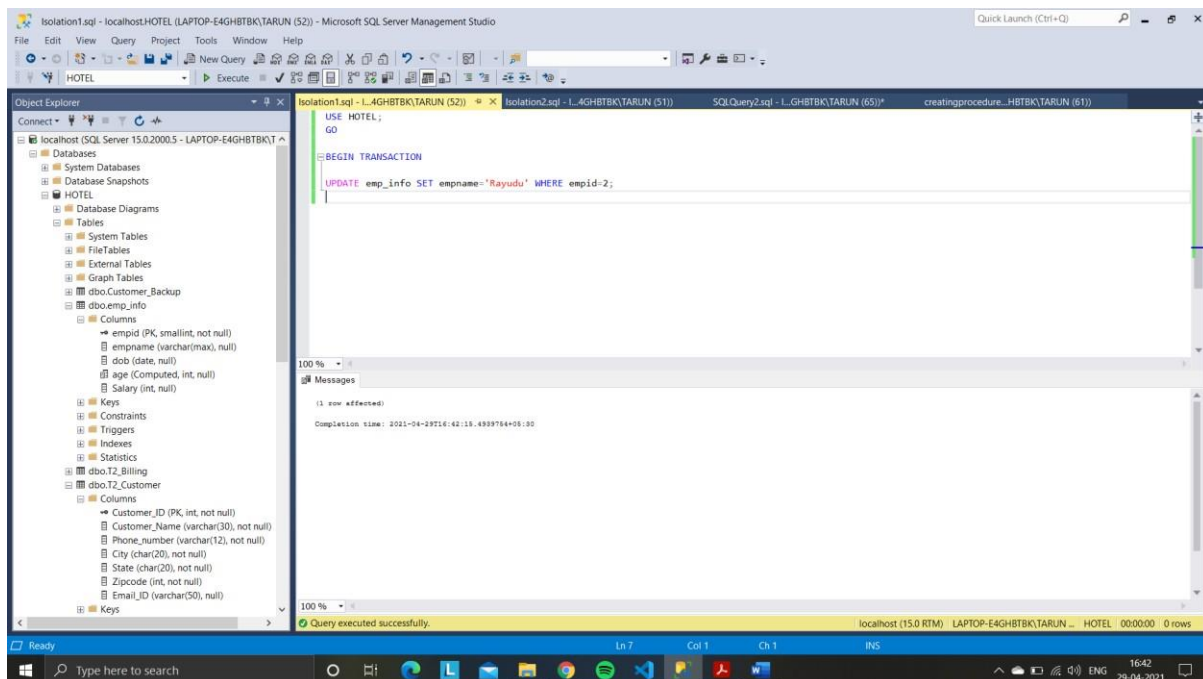
Query completed with errors. localhost (15.0 RTM) DESKTOP-NLAHG8Q\RANDHE... HOTEL 00:00:00 0 rows



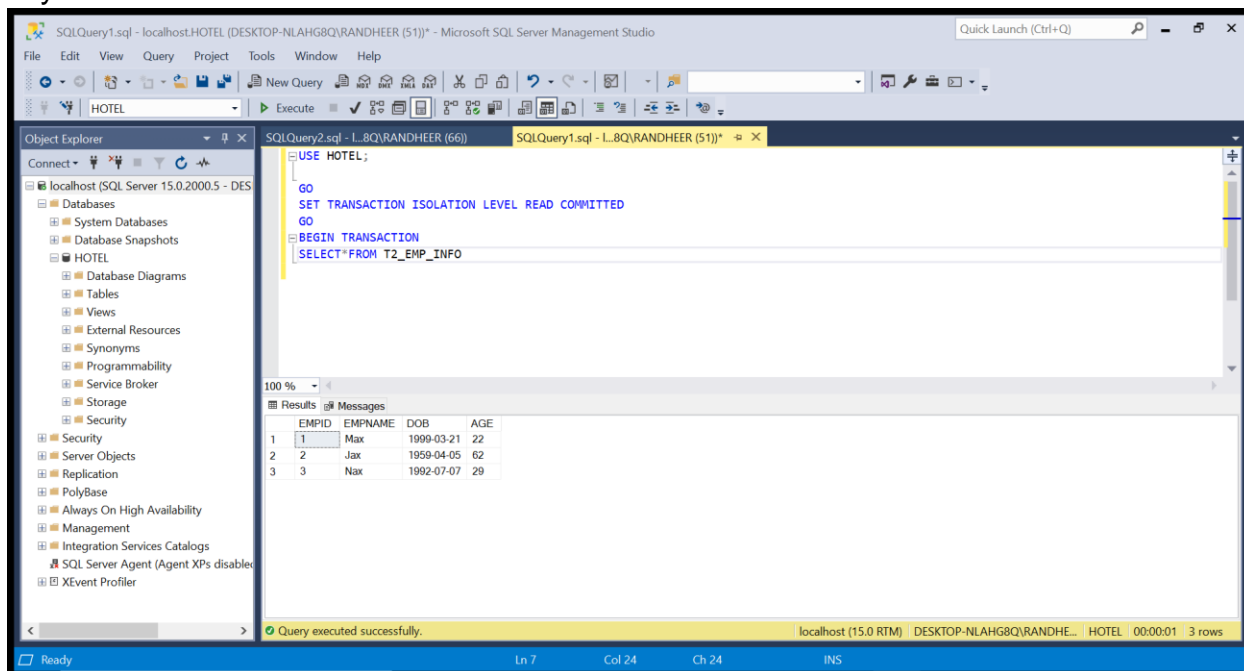
As we could see here when error occurs in the transaction it got rolled back so the table has its previous values and didn't get updated.

It means all the statements inside a transaction should either succeed or fail as a unit. So this is the atomicity property.

Q3) Write a transaction to illustrate isolation level. It can be on commit or uncommitted (related to your database) ?



This transaction in one window is not yet committed so now in other window we will try to access the table



As we can see sql server shows that the query is executing but it is not giving any output.

As you can see from the above acid properties screenshot, the select statement is not returning any information. Because we can't access one transaction result without completing the transaction. This is the isolation property