DATABASE 1 PROJECT
(HOUSE OF BALLONS HOTEL)

by: Ahad Abuyabis

&

Team

HOUSE OF BALLONS HOTEL

GROUP WORK

WORK	AHAD				
REFINE PHASE 1	✓	✓			
REFINE PHASE2	✓	✓		✓	✓
PHASE3	✓	✓	✓	✓	✓

BUSINESS RULES

- for Employees:
  - An employee must have a unique employee ID.
  - An employee must be assigned to a department.
  - An employee's contact information must be accurate and up-to-date.
  - An employee must be at least 18 years old to work at the hotel

BUSINESS RULES

- for Rooms:
  - A room must have a unique room number.
  - A room must be assigned to a room type.
  - A room must be in a clean and working condition.
  - A room must be equipped with the necessary amenities, such as a bed, a bathroom, and a television

BUSINESS RULES

- for Guests:
  - A guest must have a unique guest ID.
  - A guest must provide a valid credit card number.
  - A guest must provide accurate contact information.

BUSINESS RULES

- for Guests:
  - A guest must be at least 18 years old to check into the hotel without a parent or guardian.
  - A guest must provide a valid government-issued ID upon check-in.
  - A guest must not have any outstanding warrants or other legal issues.

BUSINESS RULES

- Reservations
  - A reservation must have a unique reservation ID.
  - A reservation must be made by a guest.
  - A reservation must be for a specific room type and room number.
  - A reservation must have a start date and end date.
  - A reservation must be made at least 24 hours in advance.
  - A reservation must be paid for in full at the time of booking.

BUSINESS RULES

- Departments
  - A department must have a unique department ID.
  - A department must have a name.
  - Each department must have a manager.

BUSINESS RULES

- Services
  - A service must have a unique service ID.
  - A service must be provided by a department.
  - A service must have a name, description, and cost.
  - A service must be available for purchase by all guests.
  - A service must be priced fairly.
  - A service must be delivered in a timely and efficient manner.

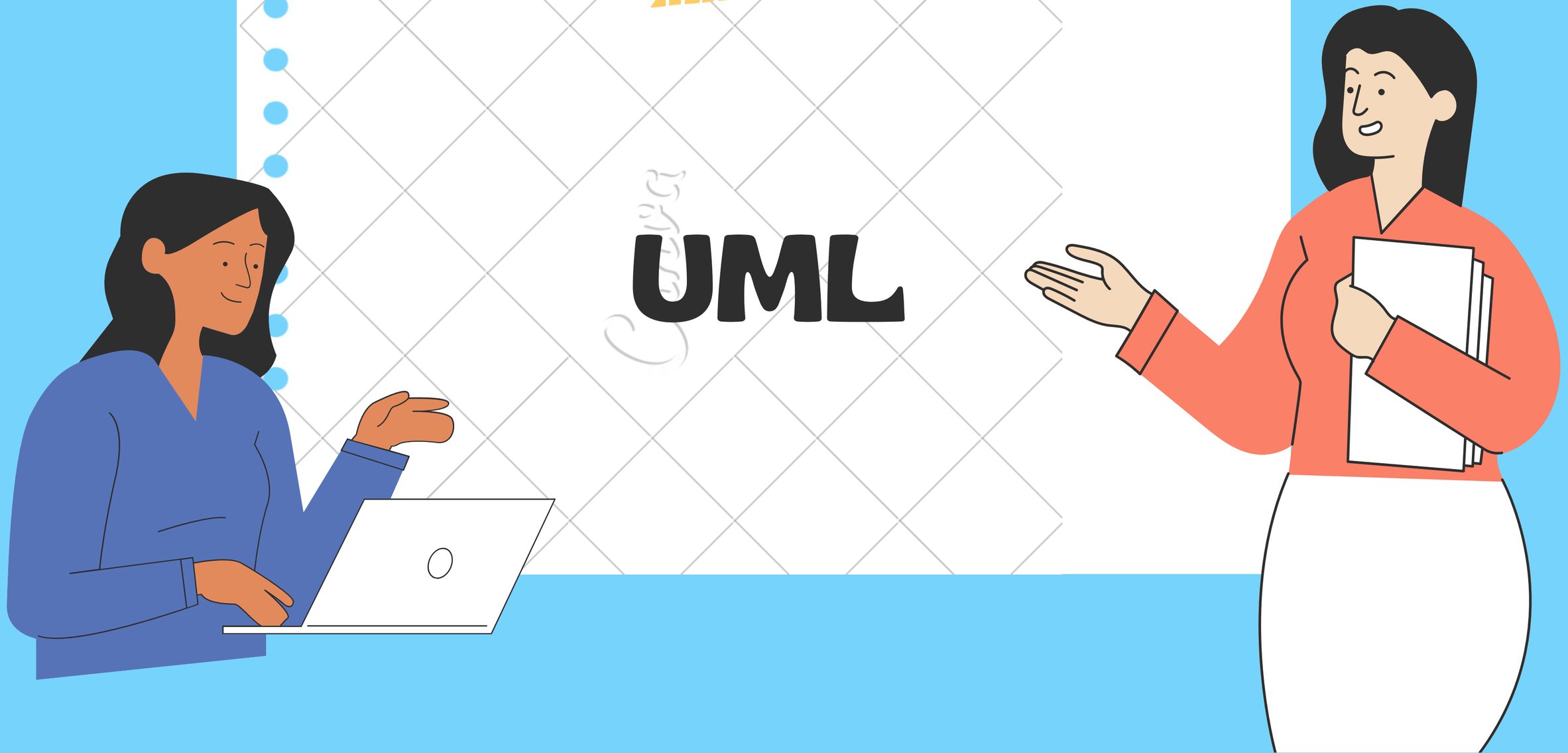
BUSINESS RULES

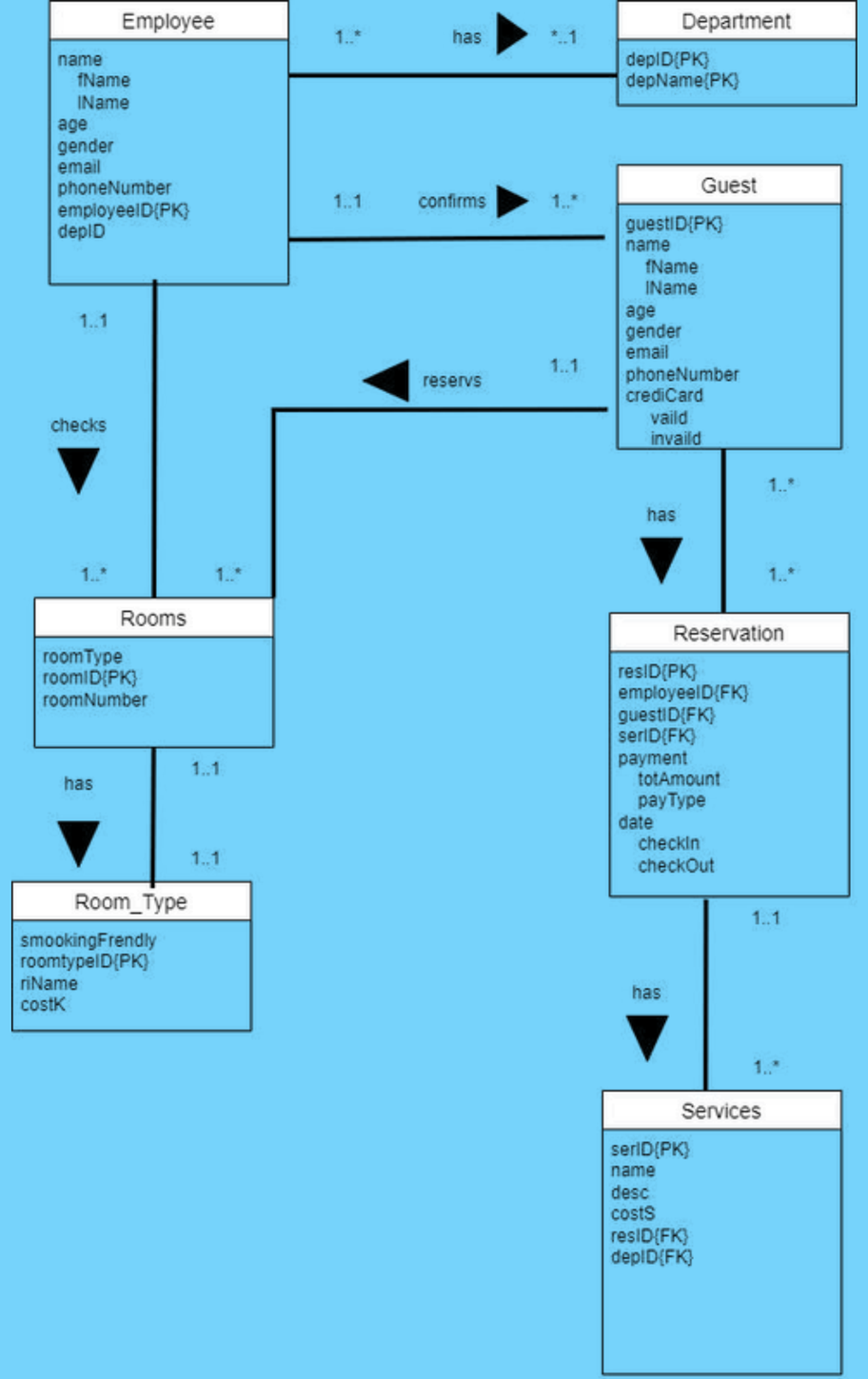
- Room Types
  - A room type must have a unique room type ID.
  - A room type must have a name and cost.
  - A room type must be smoking-friendly or non-smoking.
  - Each room type must be priced according to its amenities and features.

SUMMARY:

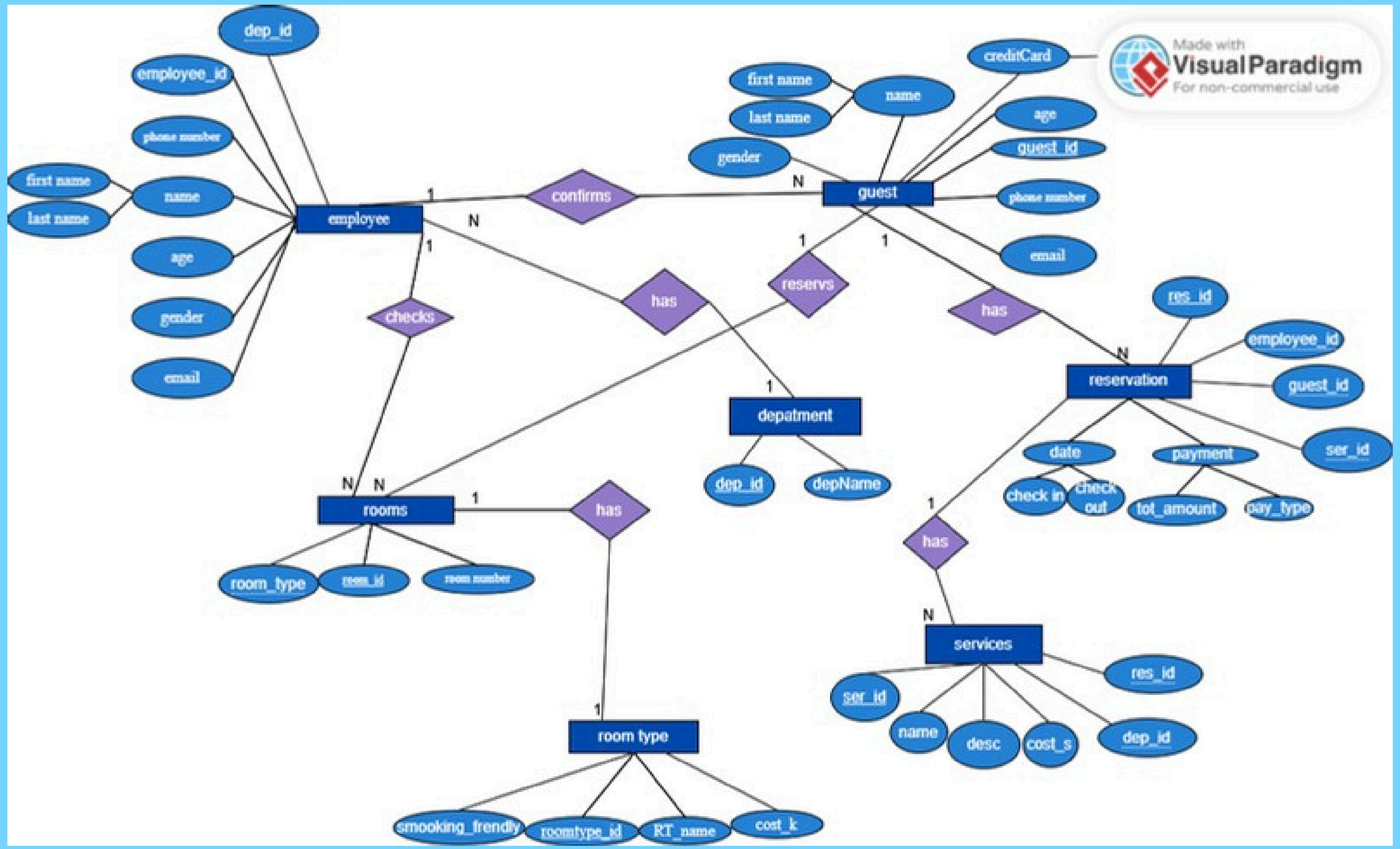
- Employee: An employee works at the hotel and is assigned to a department.
- Room: A room is available for guests to stay in.
- Guest: A guest is someone who stays at the hotel.
- Department: A department is a group of employees who work together on a specific task.
- Reservation: A reservation is a booking for a room by a guest.
- Services: A service is something that can be provided to guests by the hotel.
- RoomType: A room type is a category of rooms that have similar features and amenities.

UML




Employee
name
fName
lName
age
gender
email
phoneNumber
employeeID{PK}
depID
1..*
has
*..1
Department
depID{PK}
depName{PK}
1..1
confirms
1..*
Guest
guestID{PK}
name
fName
lName
age
gender
email
phoneNumber
crediCard
vaild
invaild
1..1
checks
reservs
1..1
1..*
1..*
Rooms
roomType
roomID{PK}
roomNumber
1..1
has
1..1
Room_Type
smookingFrendly
roomtypeID{PK}
riName
costK
has
1..*
1..*
Reservation
resID{PK}
employeeID{FK}
guestID{FK}
serID{FK}
payment
totAmount
payType
date
checkIn
checkOut
1..1
has
1..*
Services
serID{PK}
name
desc
costS
resID{FK}
depID{FK}

CHIN NOTATION


dep_id
employee_id
phone number
first name
name
last name
age
gender
email
employee
confirms
checks
has
reservs
has
has
has
first name
last name
name
gender
creditCard
age
guest_id
phone number
email
guest
depatment
dep_id
depName
rooms
room_type
rees_id
room number
room type
smooking_frendly
roomtype_id
RT_name
cost_k
reservation
res_id
employee_id
guest_id
ser_id
date
payment
check in
check out
tot_amount
pay_type
services
ser_id
name
desc
cost_s
res_id
dep_id
Made with
Visual Paradigm
For non-commercial use

MAPPING

Step1: mapping regular entities

only include simple components of composite attributes (name, date, card statu)

Employee

depID	eID	Phone_number	First_name	Last_name	age	gender	email

Guest

gID	Phone_number	age	valid	invalid	First_name	Last_name	gender	email

Rooms

Room_typeID	Room_ID	Room_number

Room type

Smooking_frendly	Room_typeID	RT_name	Cost_k

Reservation

resID	eID	gID	serID	Tot_amount	Pay_type	Check_in	Check_out

Depatment

depID	depName

Services

serID	name	desc	Cost_s	depID	resID

Step2: mapping of weak entity type; there is no weak entities

Step3: mapping of binary 1:1 relationship types:

Employee

depID	eID	Phone_number	First_name	Last_name	age	gender	email

Guest

gID	Phone_number	age	valid	invalid	First_name	Last_name	gender	email

Room has the foreign key of room type id because each room has only one type

Rooms

Room_typeID	Room_ID	Room_number

Room type

Smooking_frendly	Room_typeID	RT_name	Cost_k

Reservation

resID	eID	gID	serID	Tot_amount	Pay_type	Check_in	Check_out

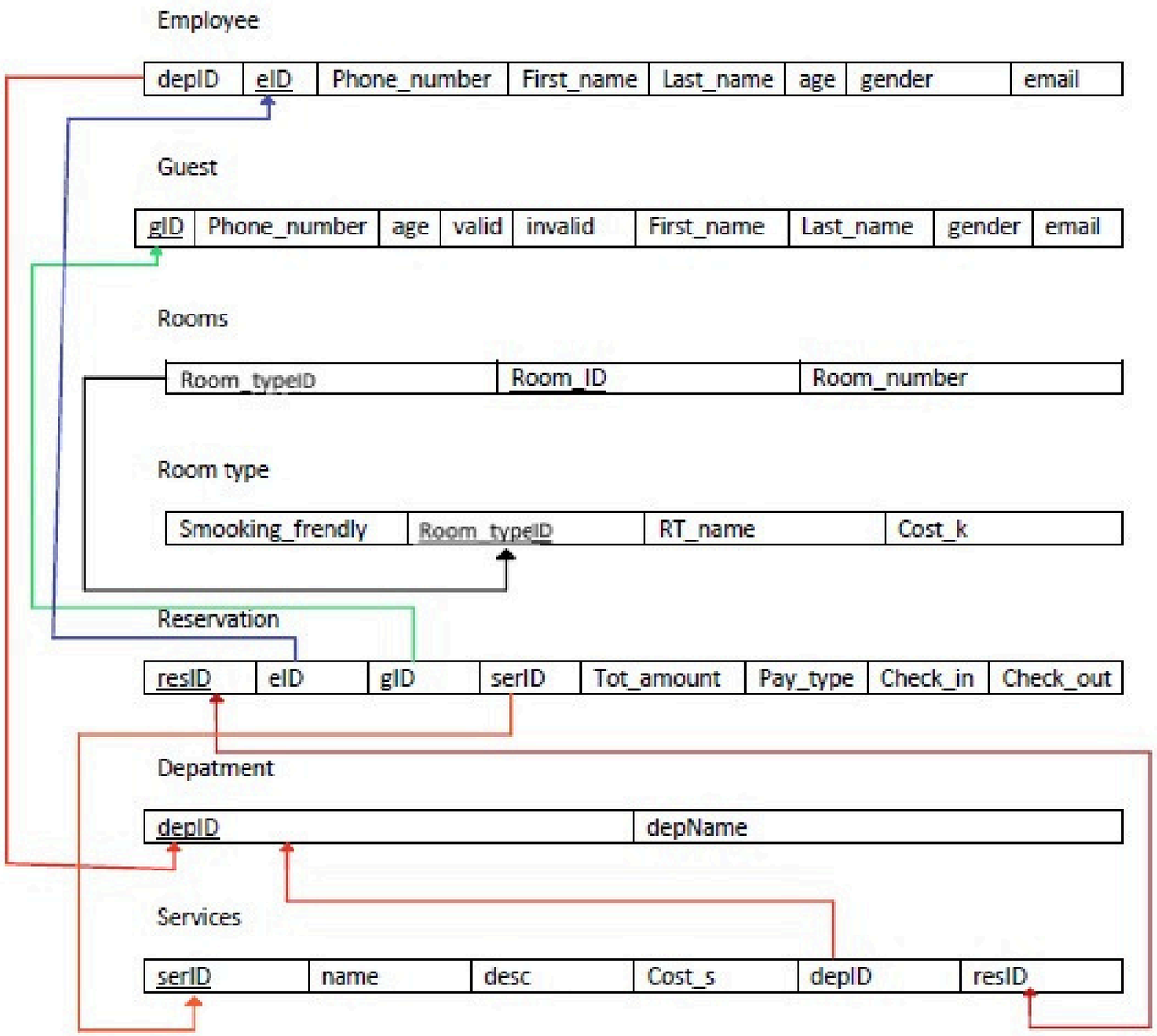
Depatment

depID	depName

Services

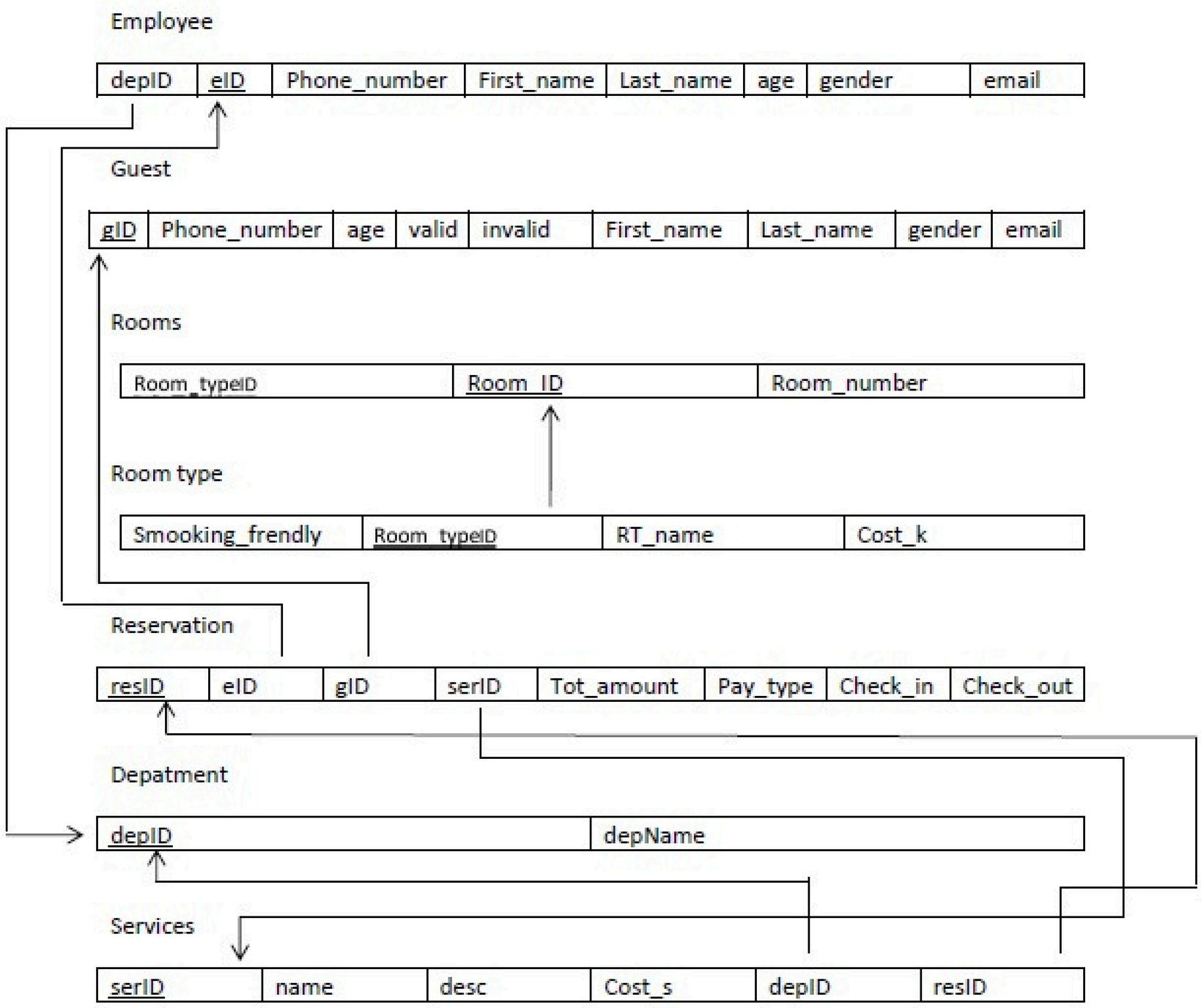
serID	name	desc	Cost_s	depID	resID

Step4: mapping of binary 1:N relationship types:



- STEP5: MAPPING OF BINARY M:N RELATIONSHIP TYPES: THERE IS NO M:N RELATIONSHIP
STEP6: MAPPING OF MULTIVALUED
- ATTRIBUTES: THERE IS NO MULTIVALUED ATTRIBUTES
STEP7: MAPPING OF TERNARY RELATIONSHIP:

- THERE IS NO TERNARY RELATIONSHIP


Employee
depID
eID
Phone_number
First_name
Last_name
age
gender
email
Guest
gID
Phone_number
age
valid
invalid
First_name
Last_name
gender
email
Rooms
Room_typeID
Room_ID
Room_number
Room type
Smooking_frendly
Room_typeID
RT_name
Cost_k
Reservation
resID
eID
gID
serID
Tot_amount
Pay_type
Check_in
Check_out
Depatment
depID
depName
Services
serID
name
desc
Cost_s
depID
resID

NORMALIZATION

NORMALIZATION:

- ACCORDING TO 1NF EACH RECORD IN TABLE CELL SHOULD CONTAIN A SINGLE VALUE AND EACH RECORD SHOULD BE UNIQUE.
- ACCORDING TO ERD (FIGURE 1.) ALL THE ENTITIES (TABLES) HAS UNIQUE ATTRIBUTES (COLUMNS) FOR EVERY RECORD. SO IT ACHIEVES THE 1NF OF NORMALIZATION.
- ACCORDING TO 2NF 1NF SHOULD ACHIEVE AND THEN APPLY DEPENDENCIES OR RELATIONSHIPS BETWEEN TABLES.

FOR THAT WE CREATED PRIMARY KEYS AND FOREIGN KEYS BETWEEN TABLES EACH TABLES HAS JOIN WITH THOSE KEYS.

TO ACHIEVE 2NF WE HAVE CREATED PRIMARY KEYS AND FOREIGN KEY AND THEN CREATED RELATIONSHIPS BETWEEN THOSE PRIMARY AND FOREIGN KEYS.

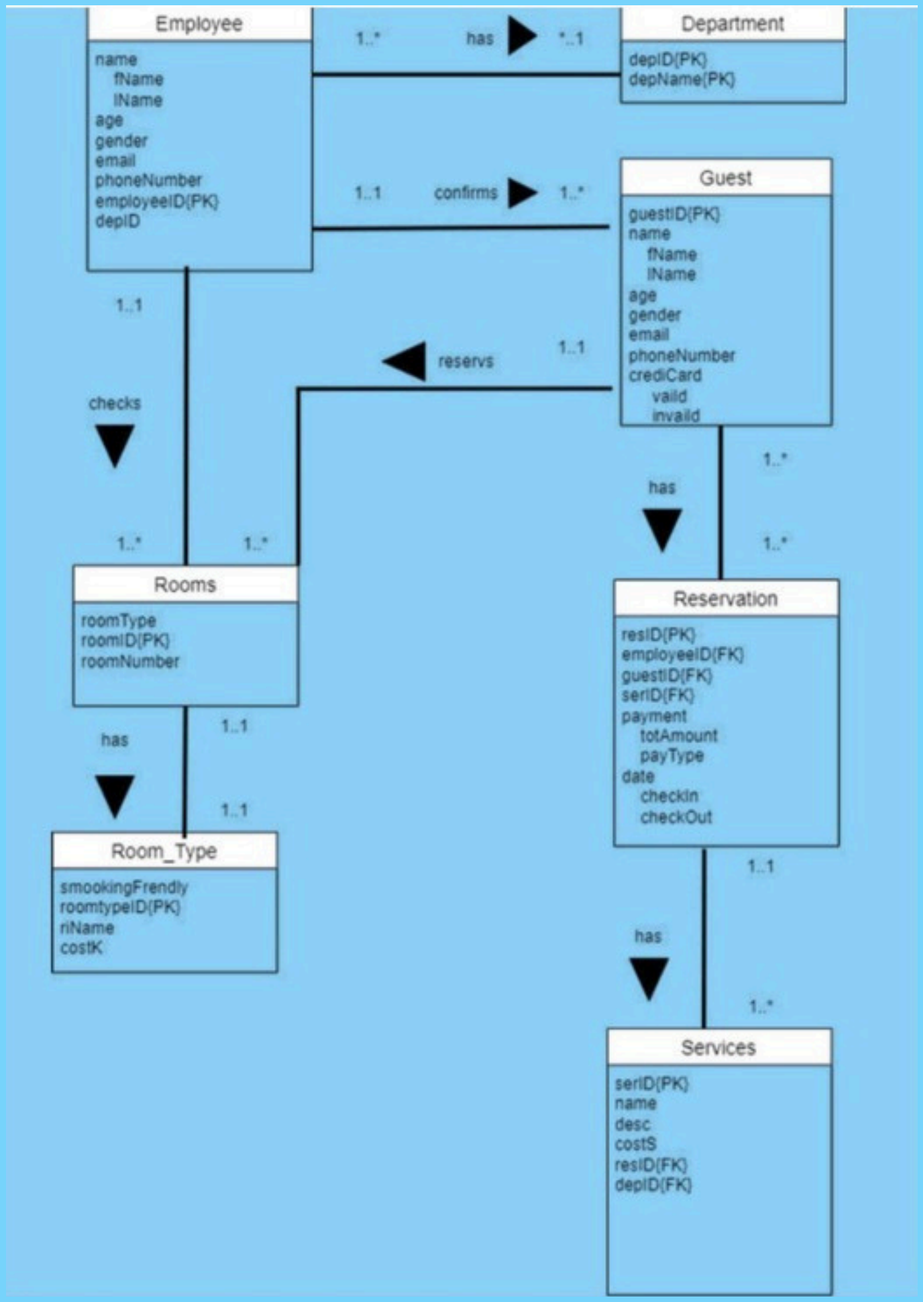


FIGURE1. ENTITY REALTIONSHIP DIAGRAM (ERD)

ACCORDING TO 3NF IT SHOULD BE IN 2NF AND IT HAS NO TRANSITIVE FUNCTIONAL DEPENDENCIES. IN
PREVIOUS ERD RESERVATION TABLE HAS FOREIGN KEY OF SERID (SERVICEID) AND SERVICES TABLE HAS
FOREIGN KEY OF RESID (RESERVATIONID) (FIGURE 2). IT CREATES REDUNDANCY BECAUSE BOTH TABLES CAN
RELATED WITH ONE FOREIGN KEY AND ONE TABLE SHOULD HAVE FOREIGN KEY. HERE BOTH TABLES HAS
FOREIGN KEY OF SAME TABLES. TO REMOVE REDUNDANCY WE HAVE DELETED SERID FROM RESERVATIONS
TABLE (FIGURE 3). SERVICES TABLE CONTAINS THE RECORD THAT WHICH EMPLOYEE HAS RESERVATION

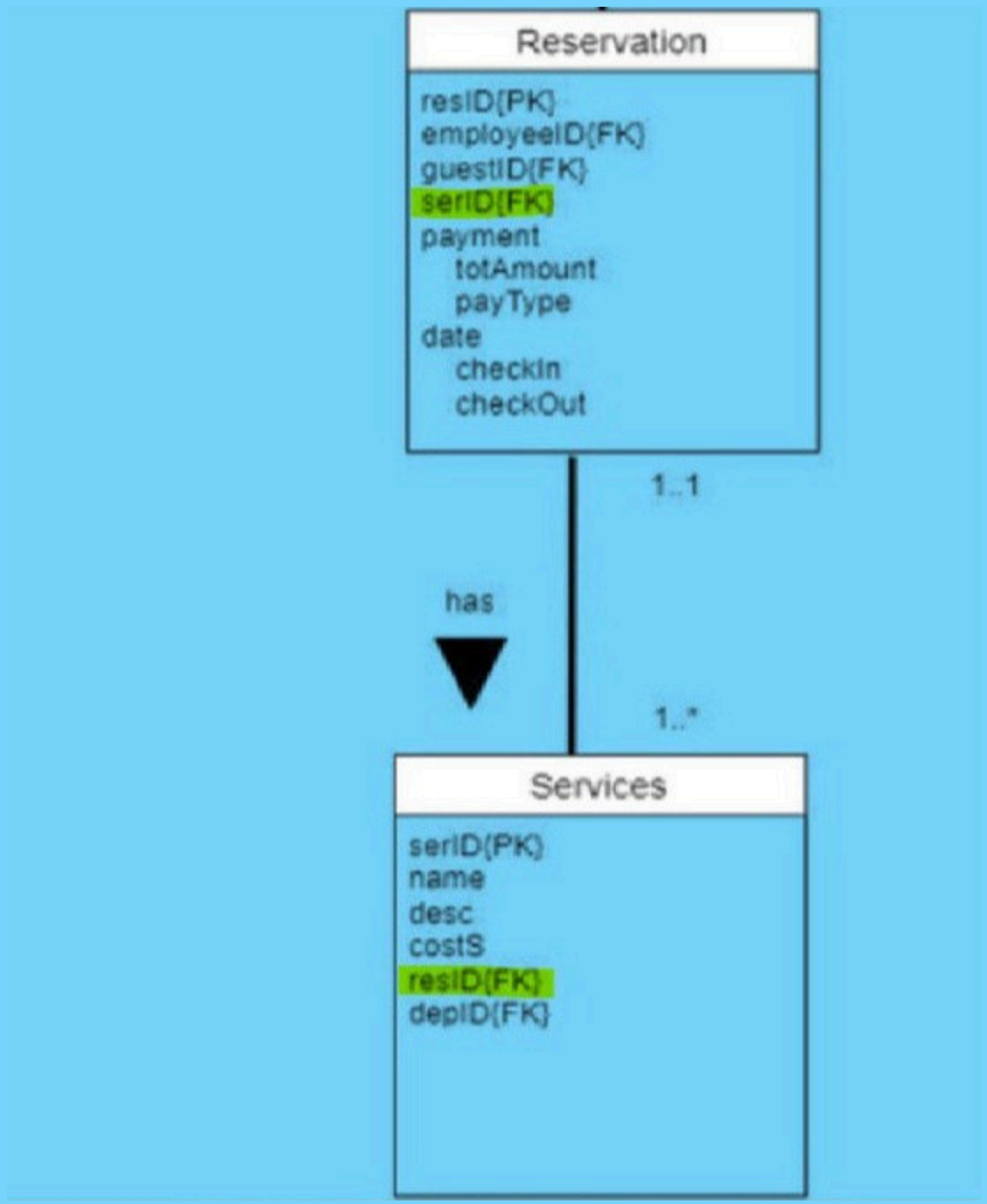


FIGURE 2. REDUNDANCY IN TABLES

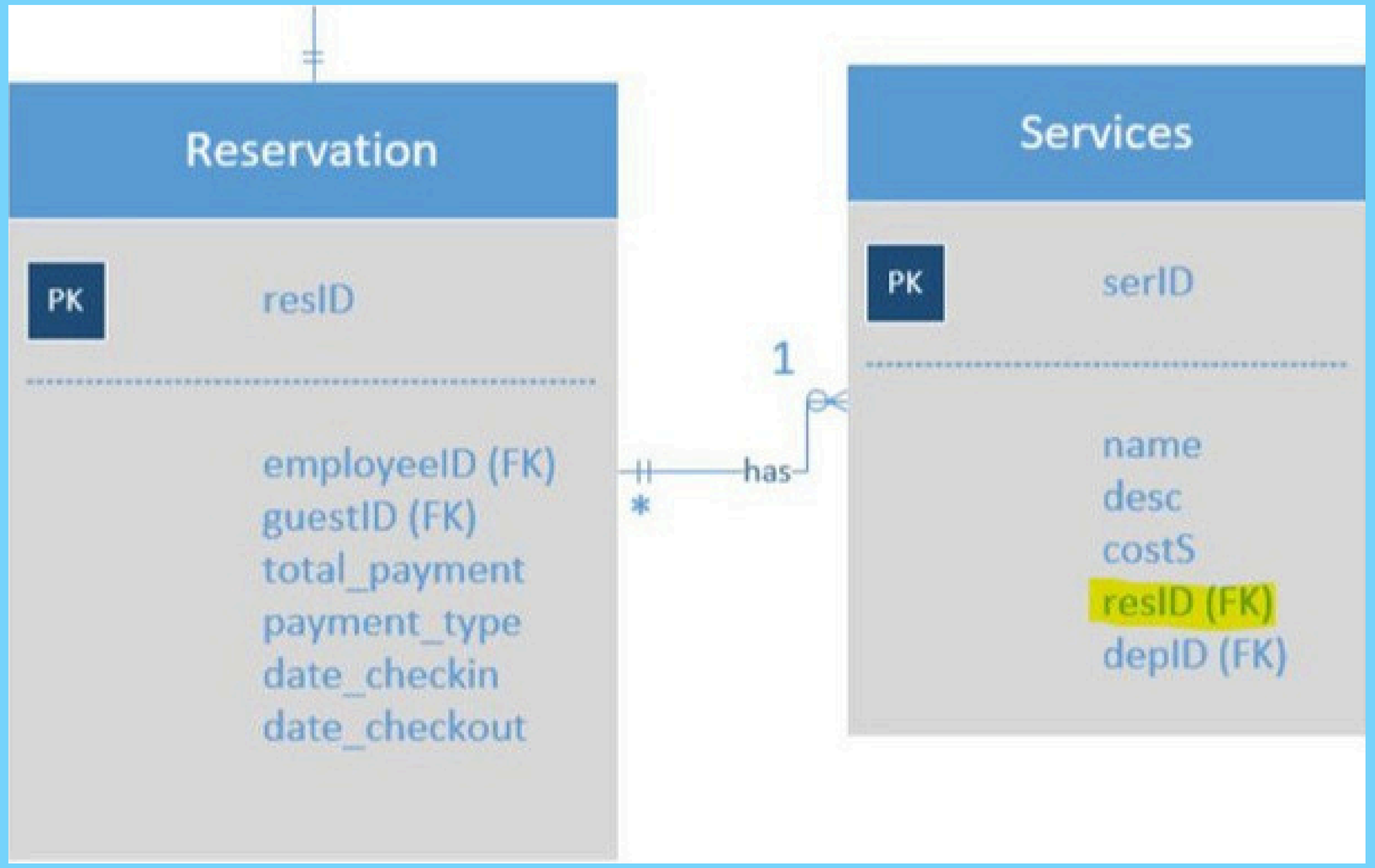
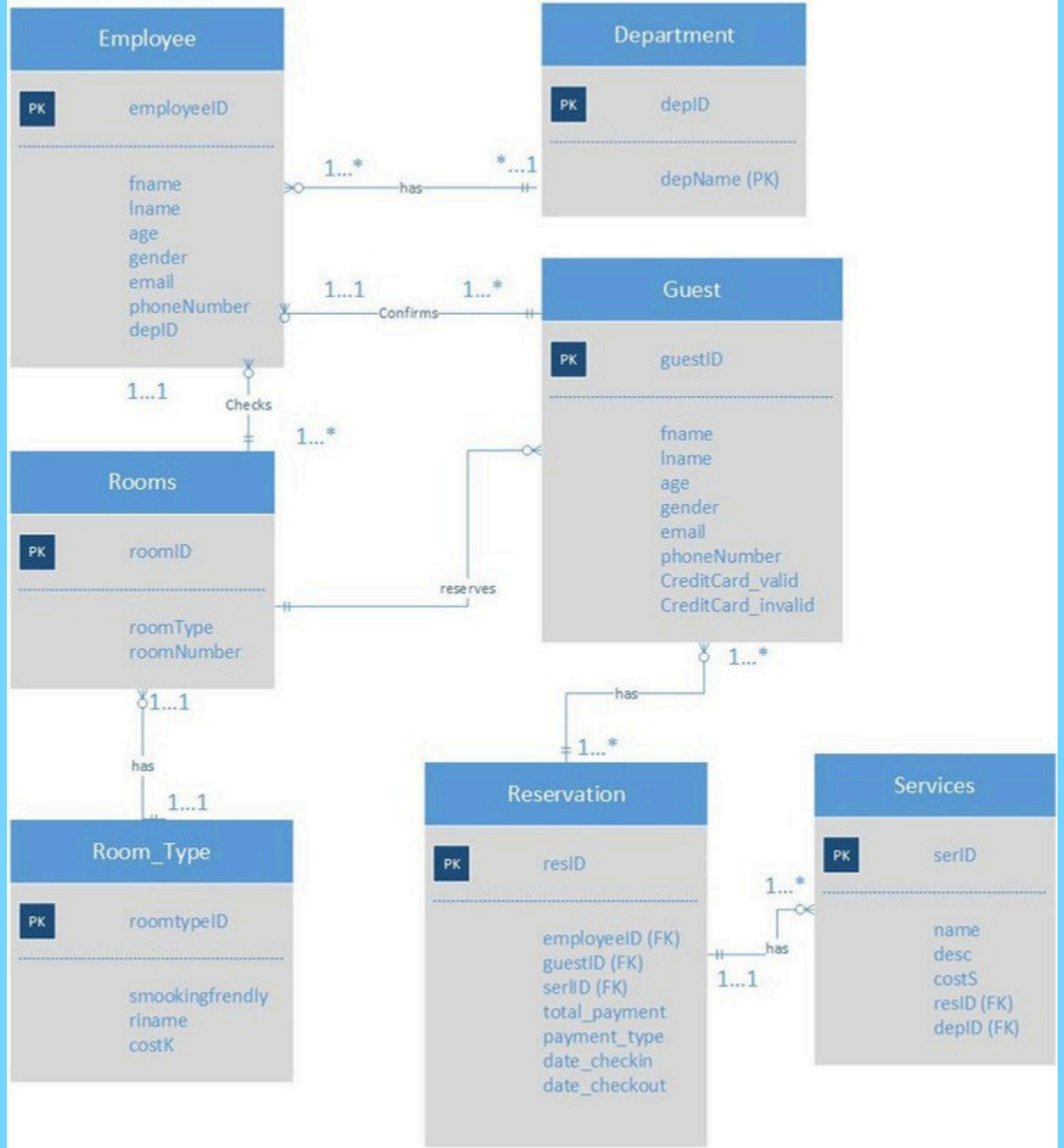


FIGURE 3. REMOVING REDUNDANCY IN TABLES

FINAL NORMALIZATION

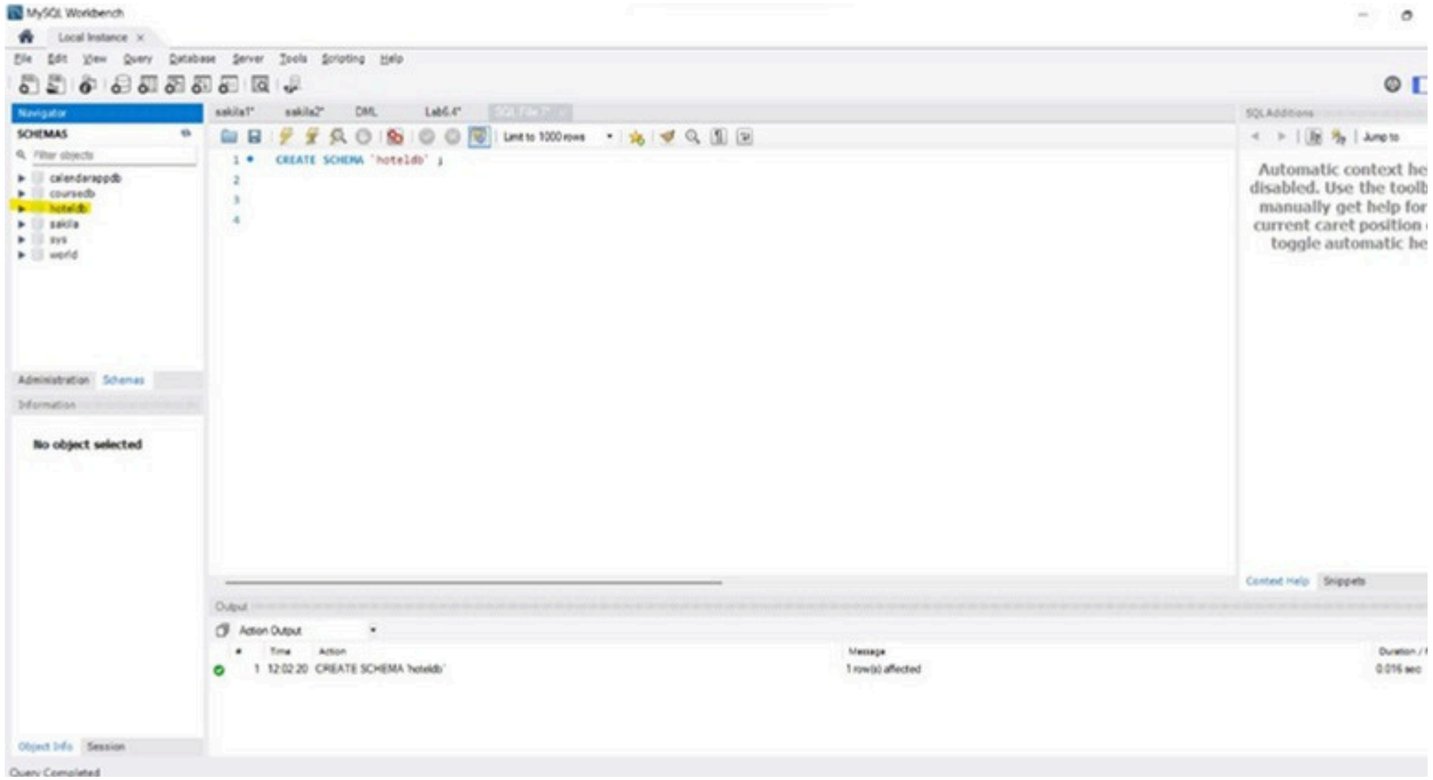
IN ERD


Employee
PK
employeeID
fname
lname
age
gender
email
phoneNumber
depID
Department
PK
depID
depName (PK)
1...*
has
*...1
1...1
Confirms
1...*
Guest
PK
guestID
fname
lname
age
gender
email
phoneNumber
CreditCard_valid
CreditCard_invalid
1...1
Checks
1...*
Rooms
PK
roomID
roomType
roomNumber
reserves
1...*
has
1...*
1...1
has
1...1
Room_Type
PK
roomtypeID
smookingfrendly
riname
costK
Reservation
PK
resID
employeeID (FK)
guestID (FK)
serID (FK)
total_payment
payment_type
date_checkin
date_checkout
1...*
has
1...1
Services
PK
serID
name
desc
costS
resID (FK)
depID (FK)

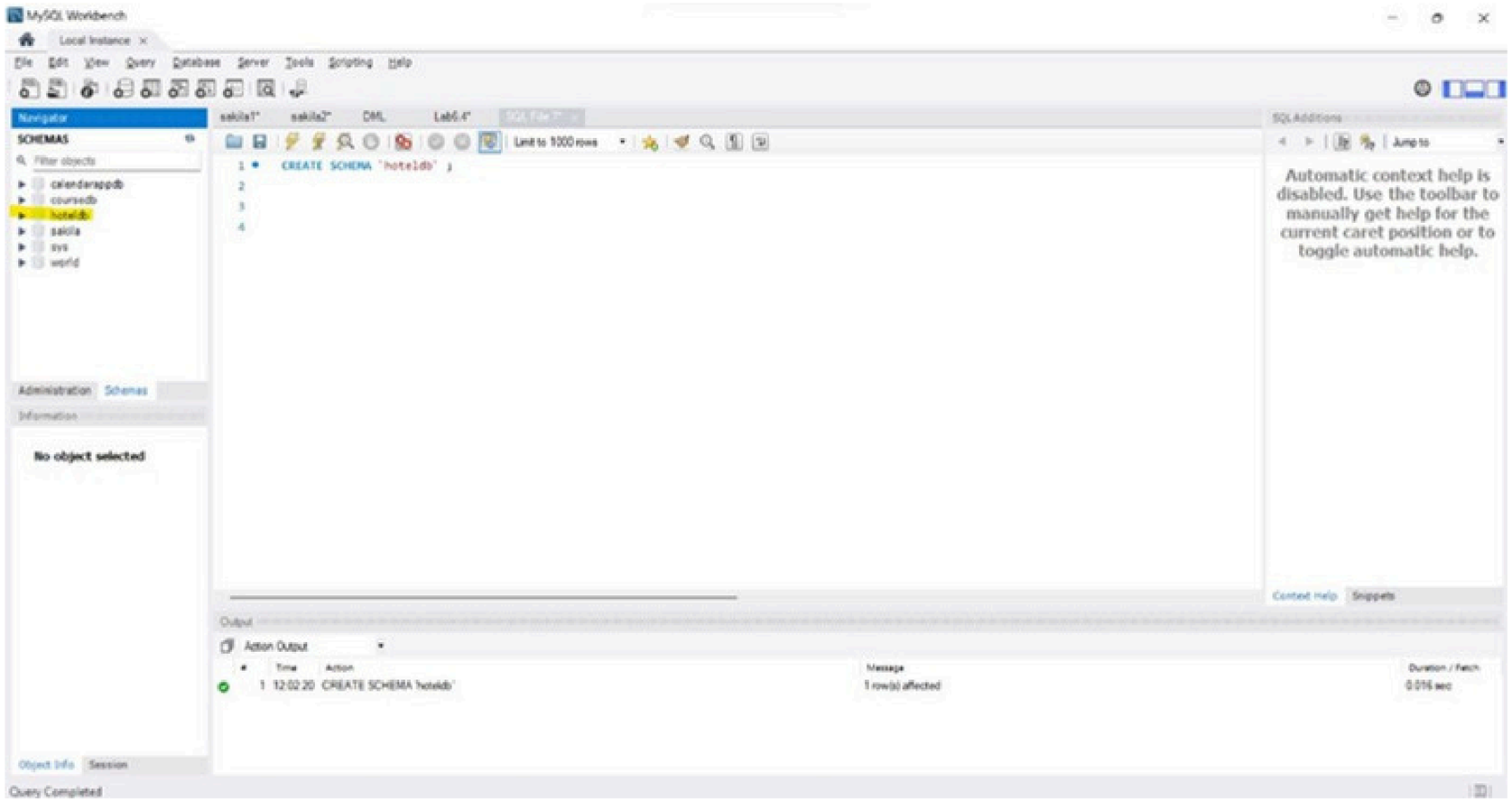
PHASE 3

SQL DEFINITION COMMANDS IN MYSQL WORKBENCH:

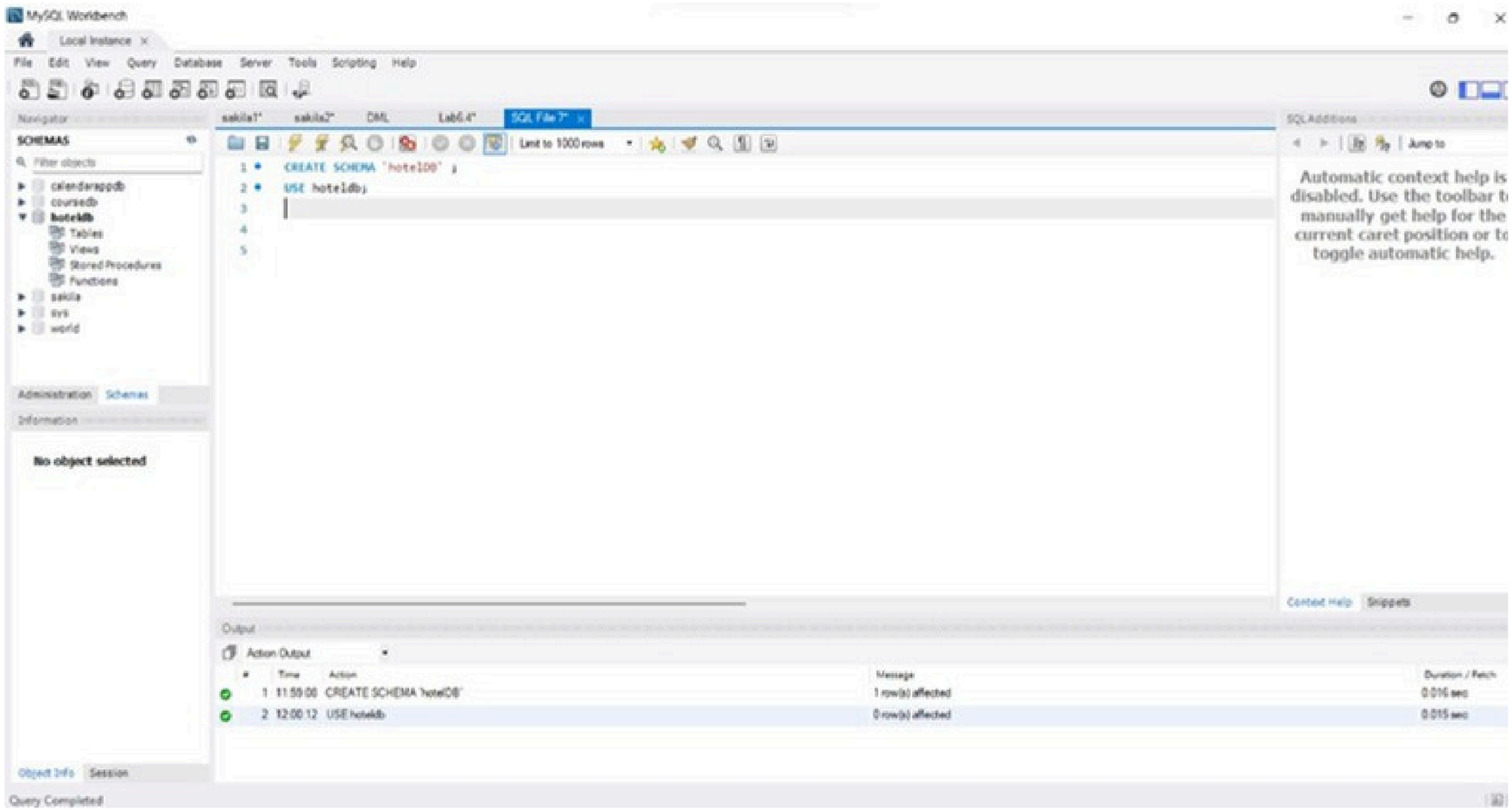
CREATE DATABASE USING CREATE SCHEMA:
DATABASE NAME IS HOTELDB FOLLOWING THE SQL QUERY TO CREATE SCHEMA.
BELOW IS THE SCREEN SHOT OF MYSQL WORKBENCH TO CREATE SCHEMA.



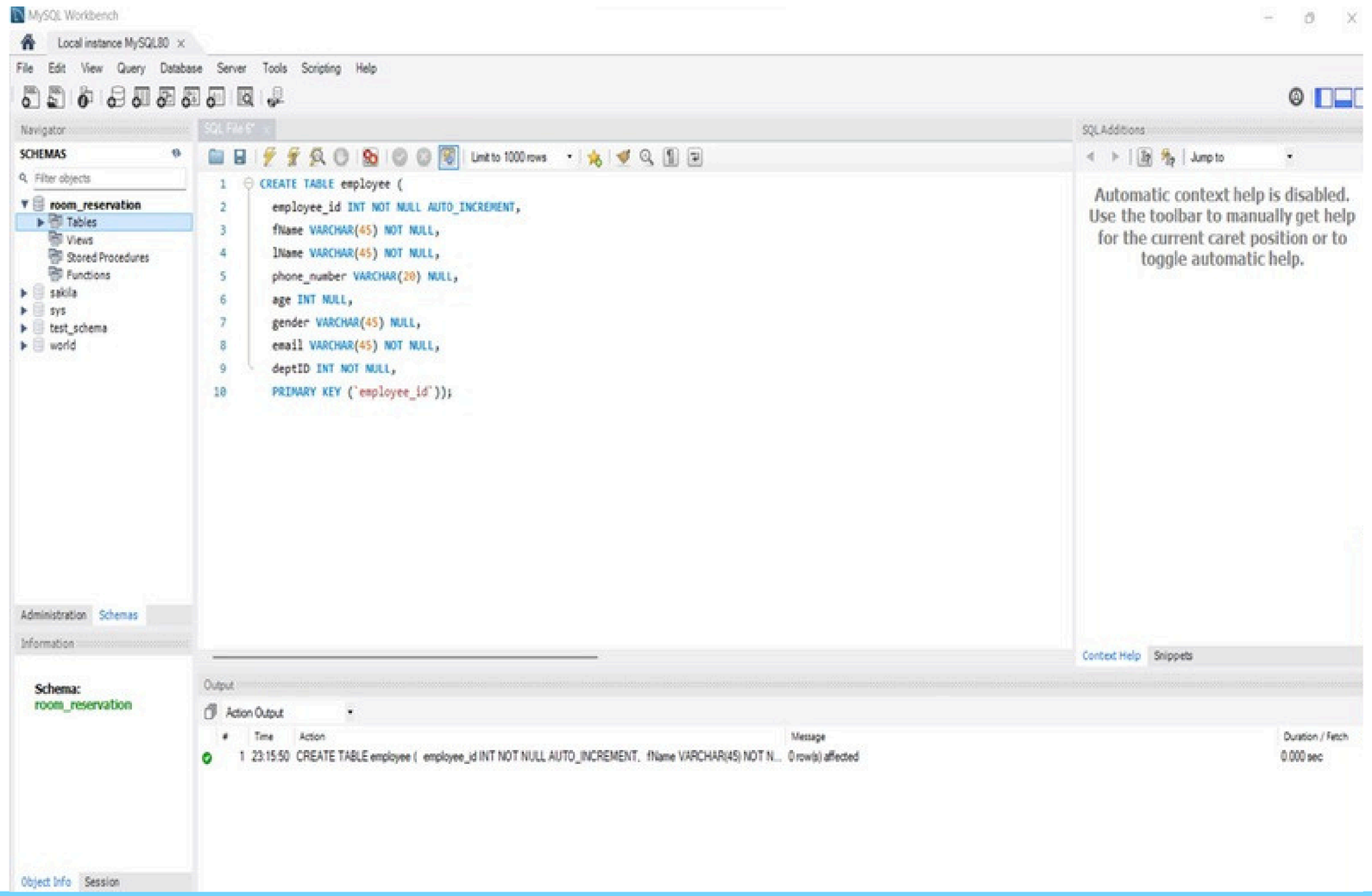
FOLLOWING SCREEN SHOT SHOWS CREATED SCHEMA IN MYSQL WORKBENCH.



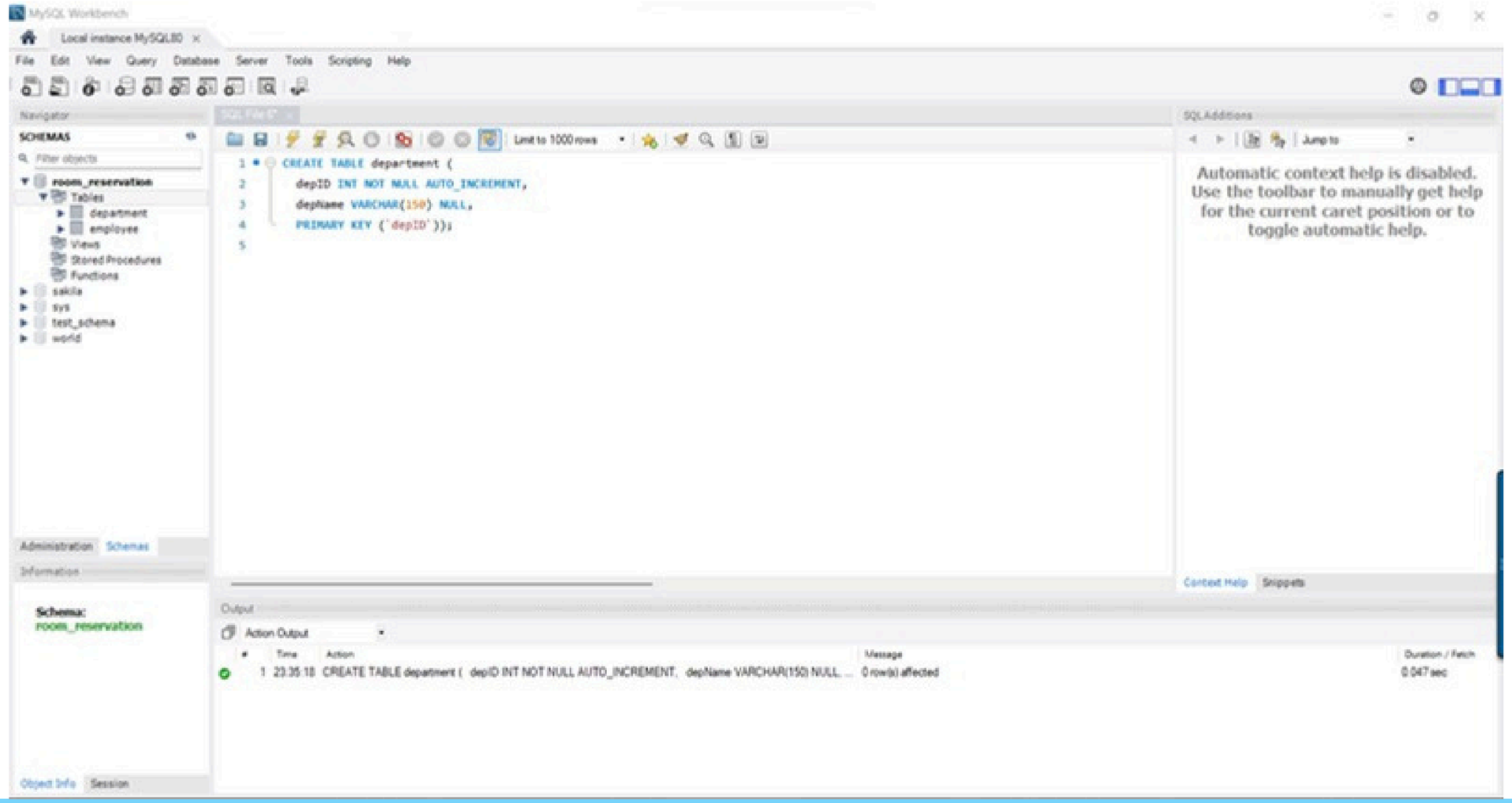
FOLLOWING THE SQL QUERY TO USE SCHEMA. BELOW IS THE SCREEN SHOT OF MYSQL WORKBENCH TO USE SCHEMA.



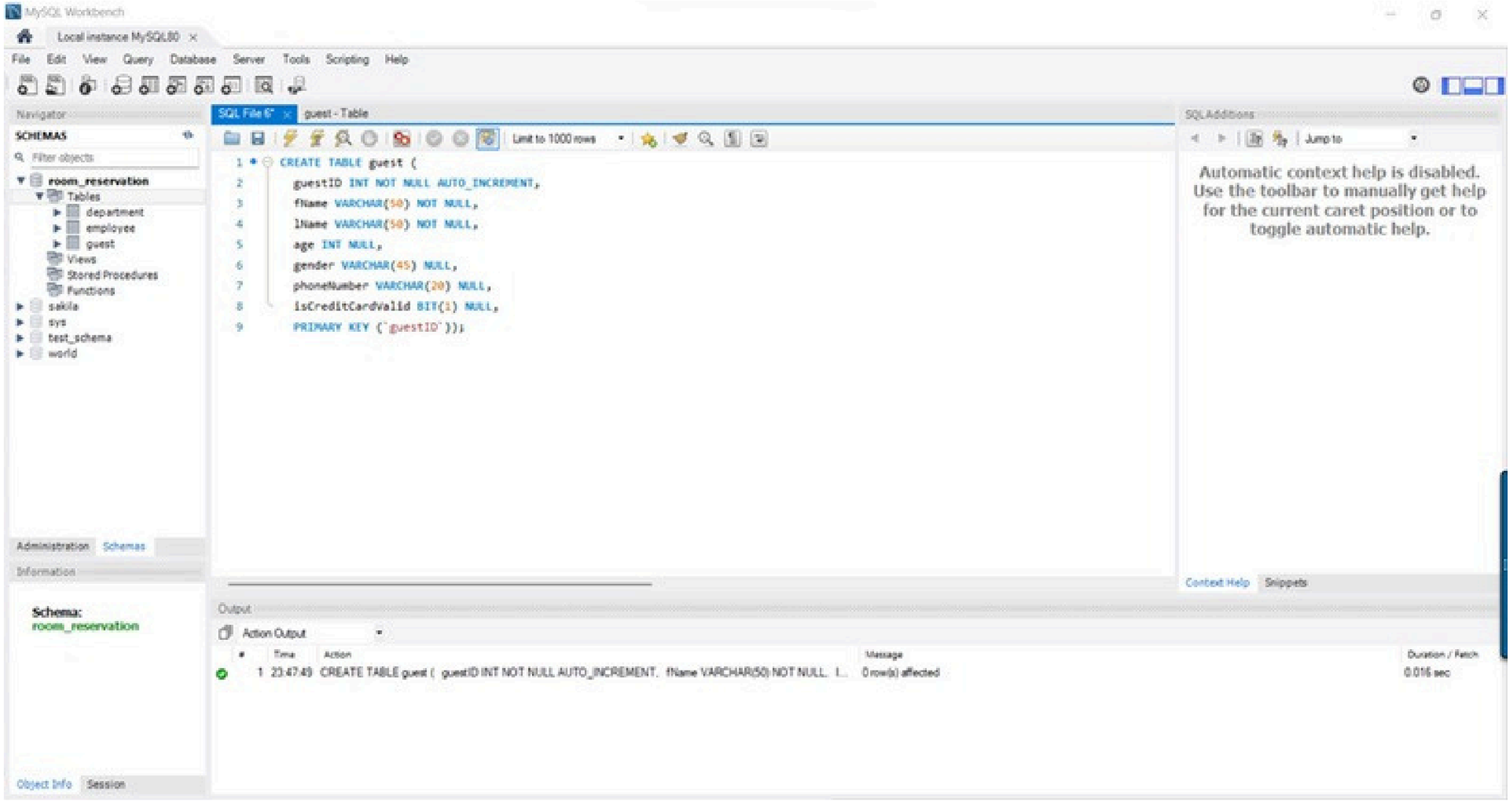
CREATE EMPLOYEE TABLE:
BELOW IS THE WOKBENCH SCREEN SHOT FOR CREATE EMPLOYEE TABLE.



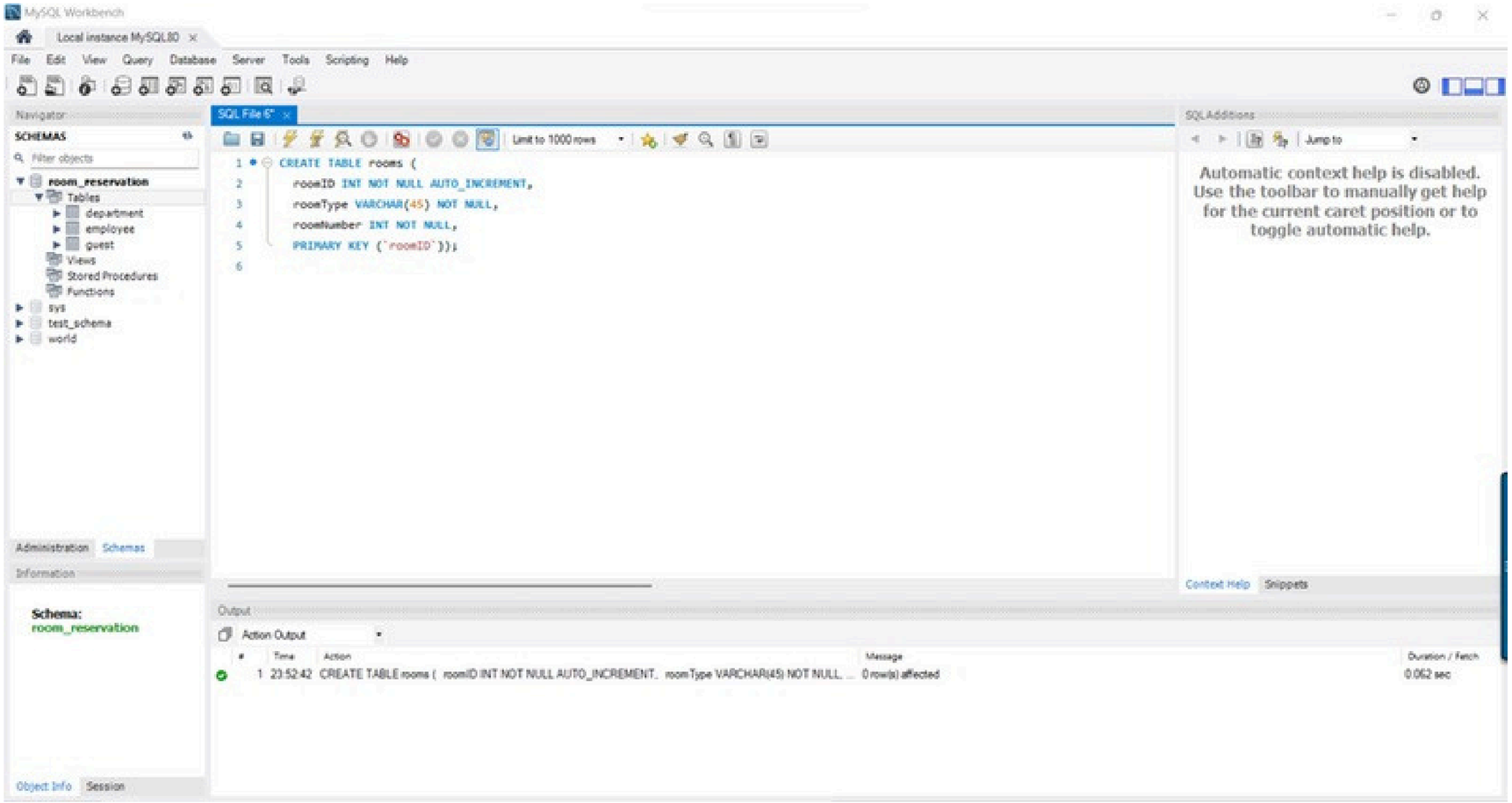
DEPARTMENT TABLE:
FOLLOWING SCREEN SHOT IS MYSQL WORKBENCH TO CREATE DEPARTMENT TABLE.



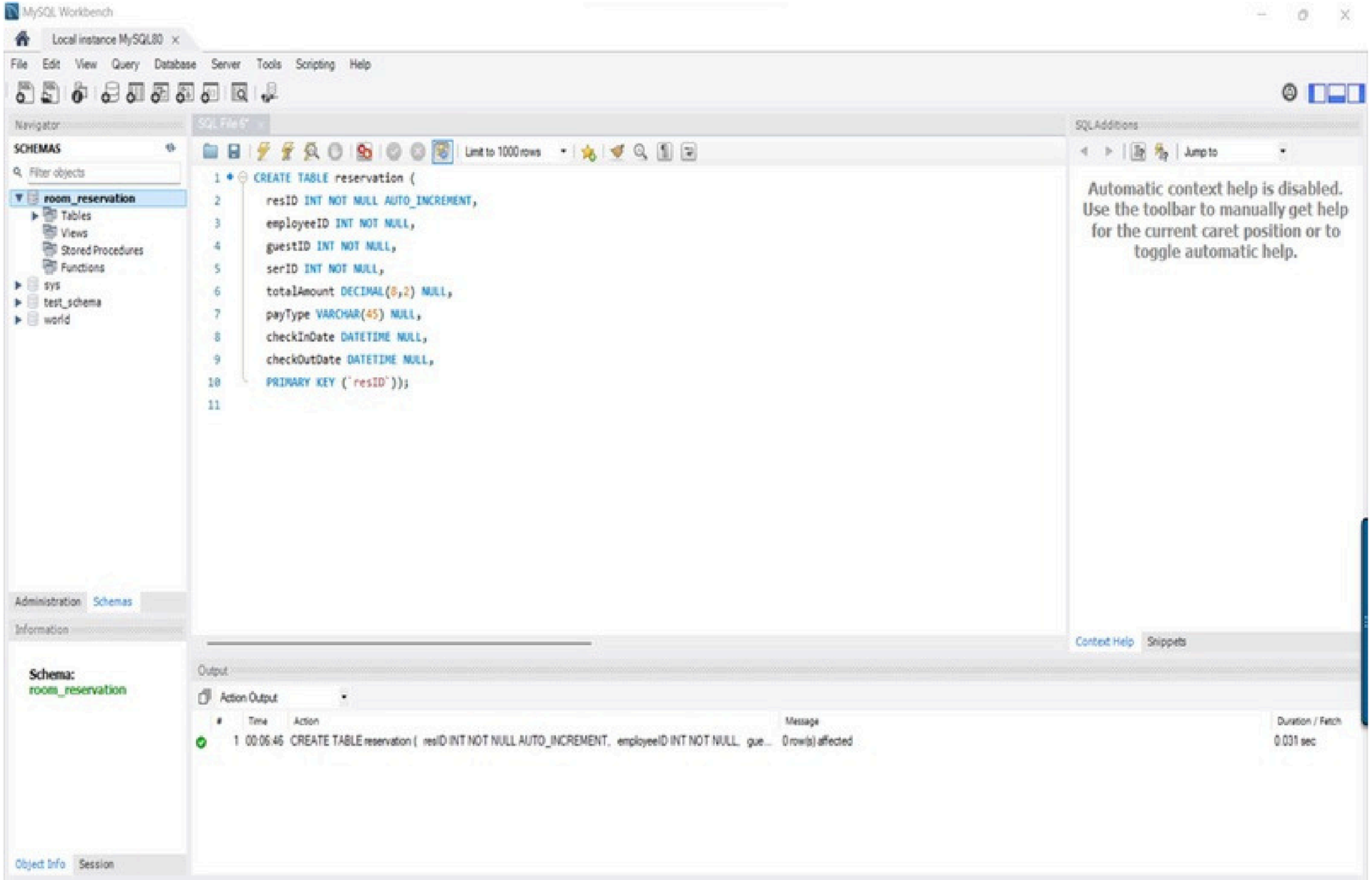
CREATE GUEST TABLE:
BELOW IS MYSQL WORKBENCH SCREEN SHOT.



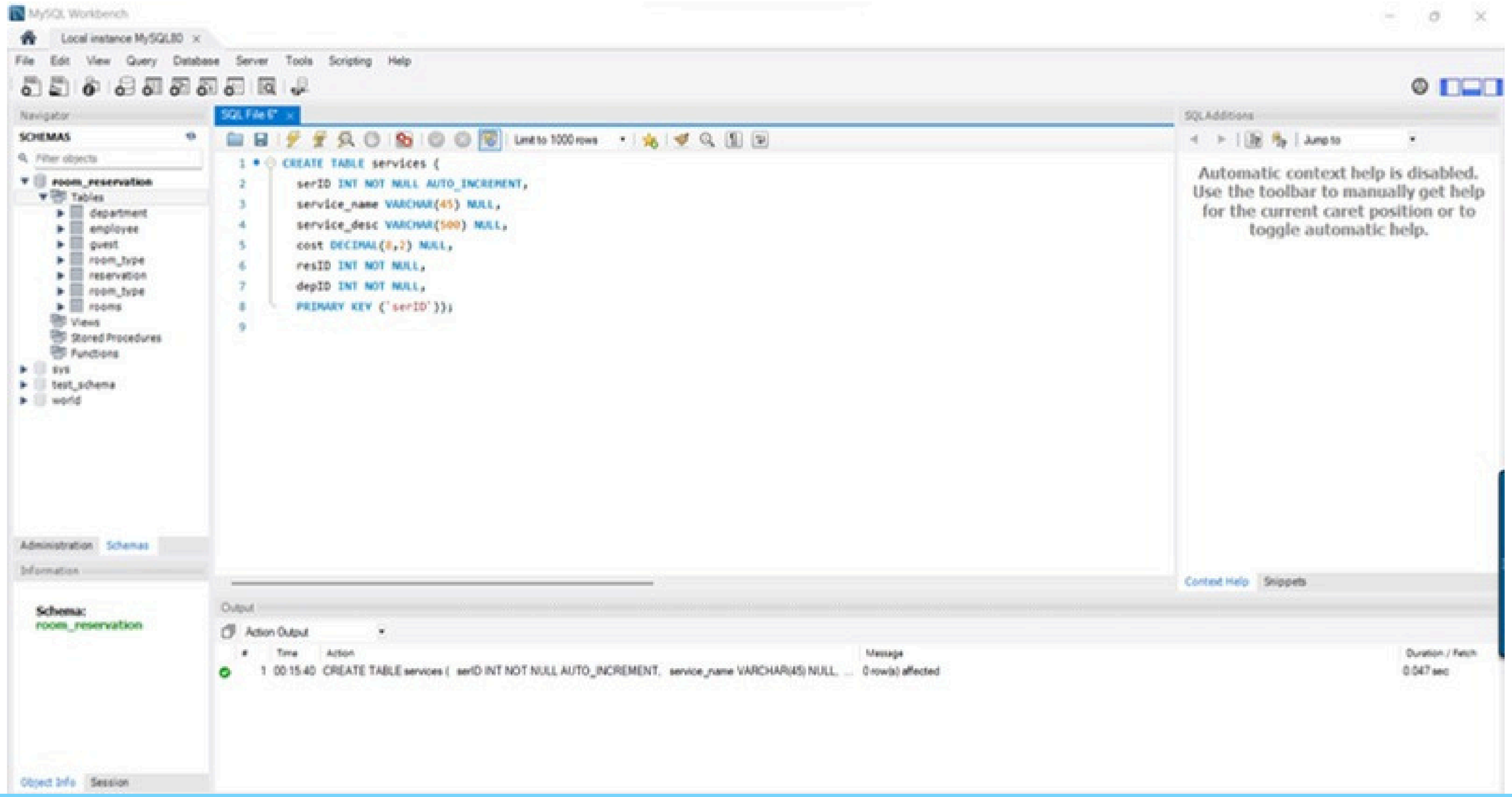
CREATE ROOMS TABLE: FOLLOWING IS MYSQL WORKBENCH SCREEN SHOT.



CREATE RESERVATION TABLE:
FOLLOWING IS MYSQL WORKBENCH SCREEN SHOT.

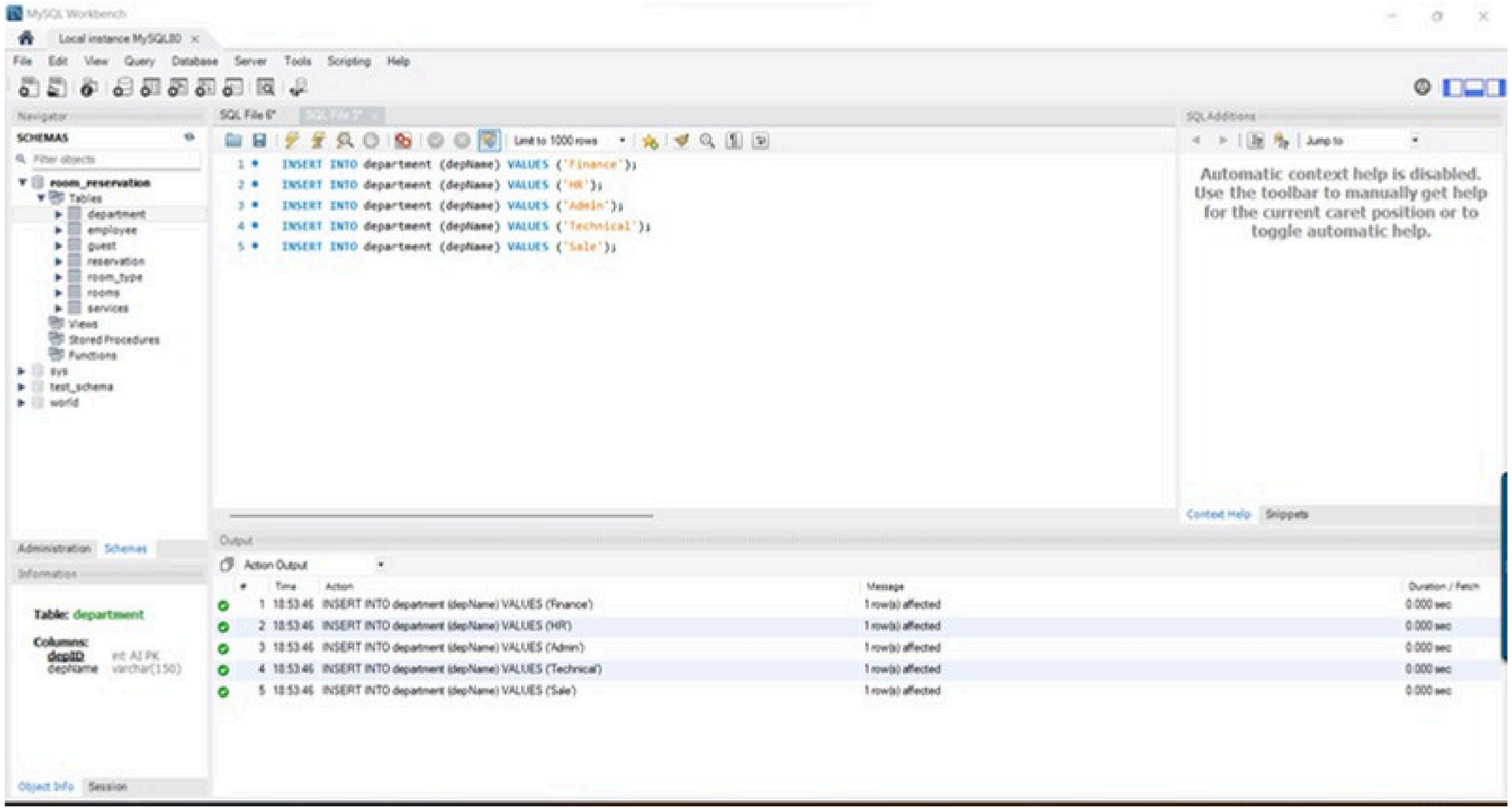


CREATE SERVICES TABLE:
BELOW IS MYSQL WORKBENCH SCREEN SHOT.

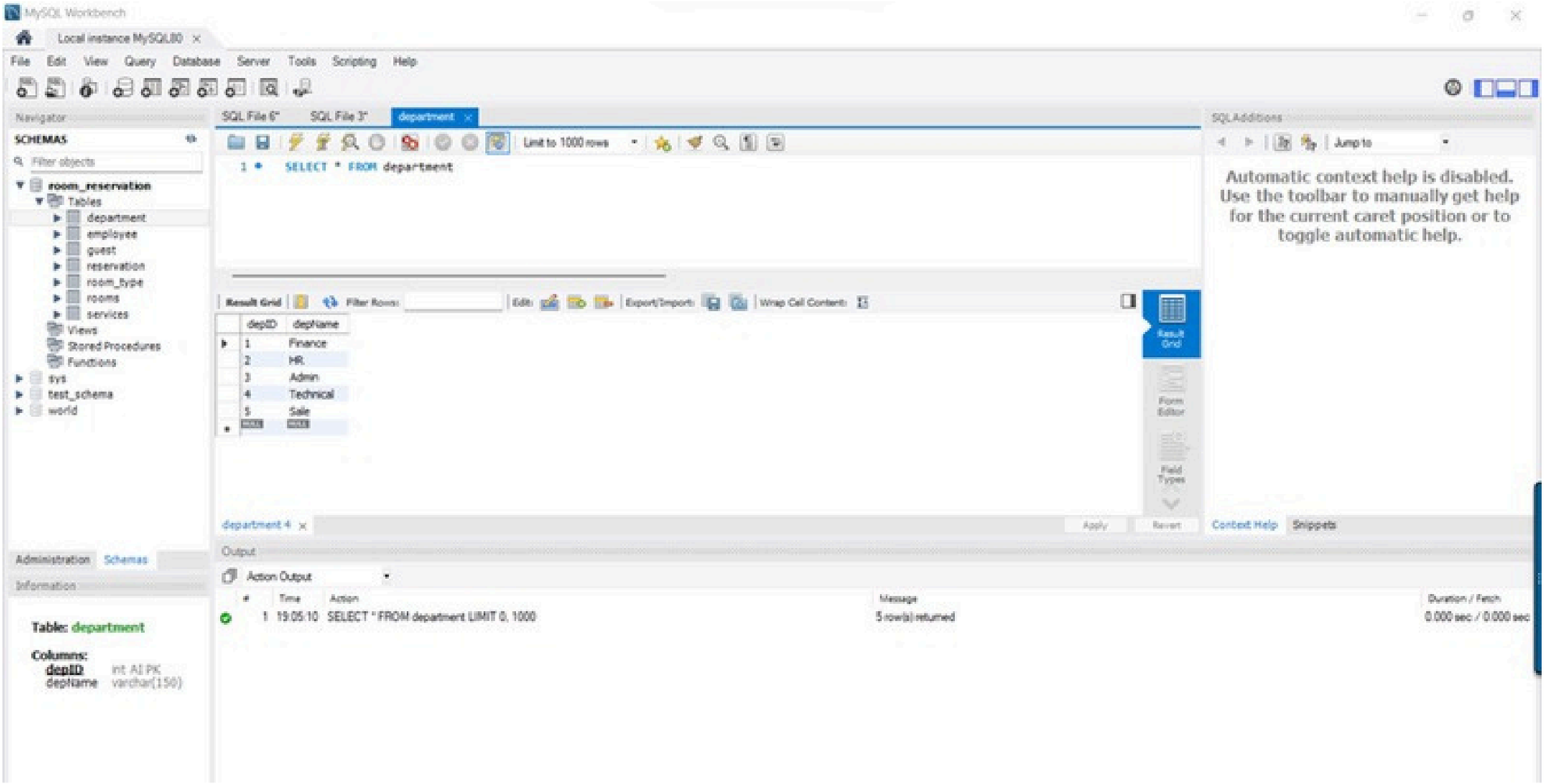


SQL data manipulation commands:

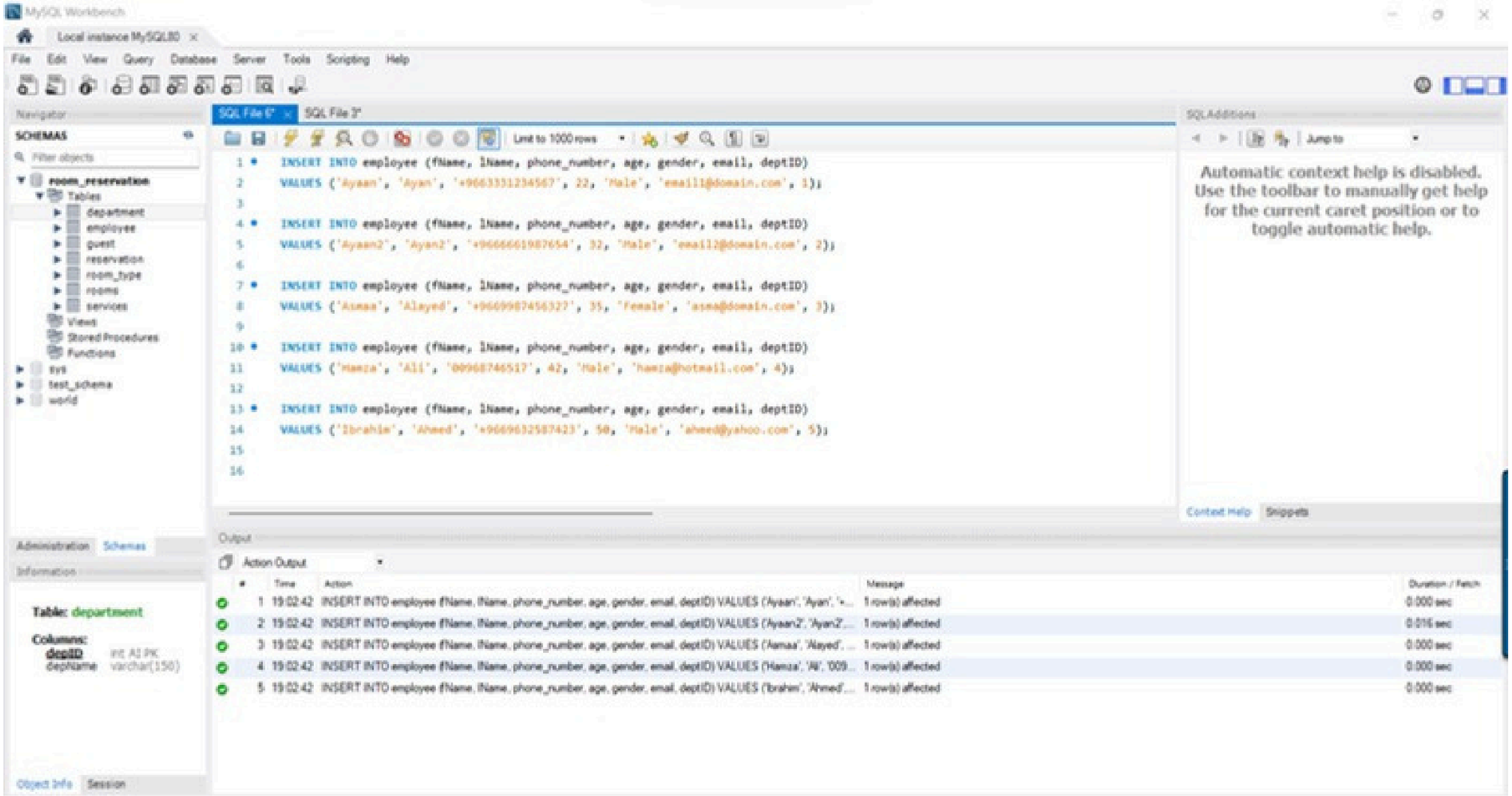
INSERT COMMANDS / QUERIES:
INSERT COMMAND FOR DEPARTMENT TABLE:
HERE ARE FIVE QUERIES/COMMAND TO INSERT FIVE RECORDS IN TABLE.



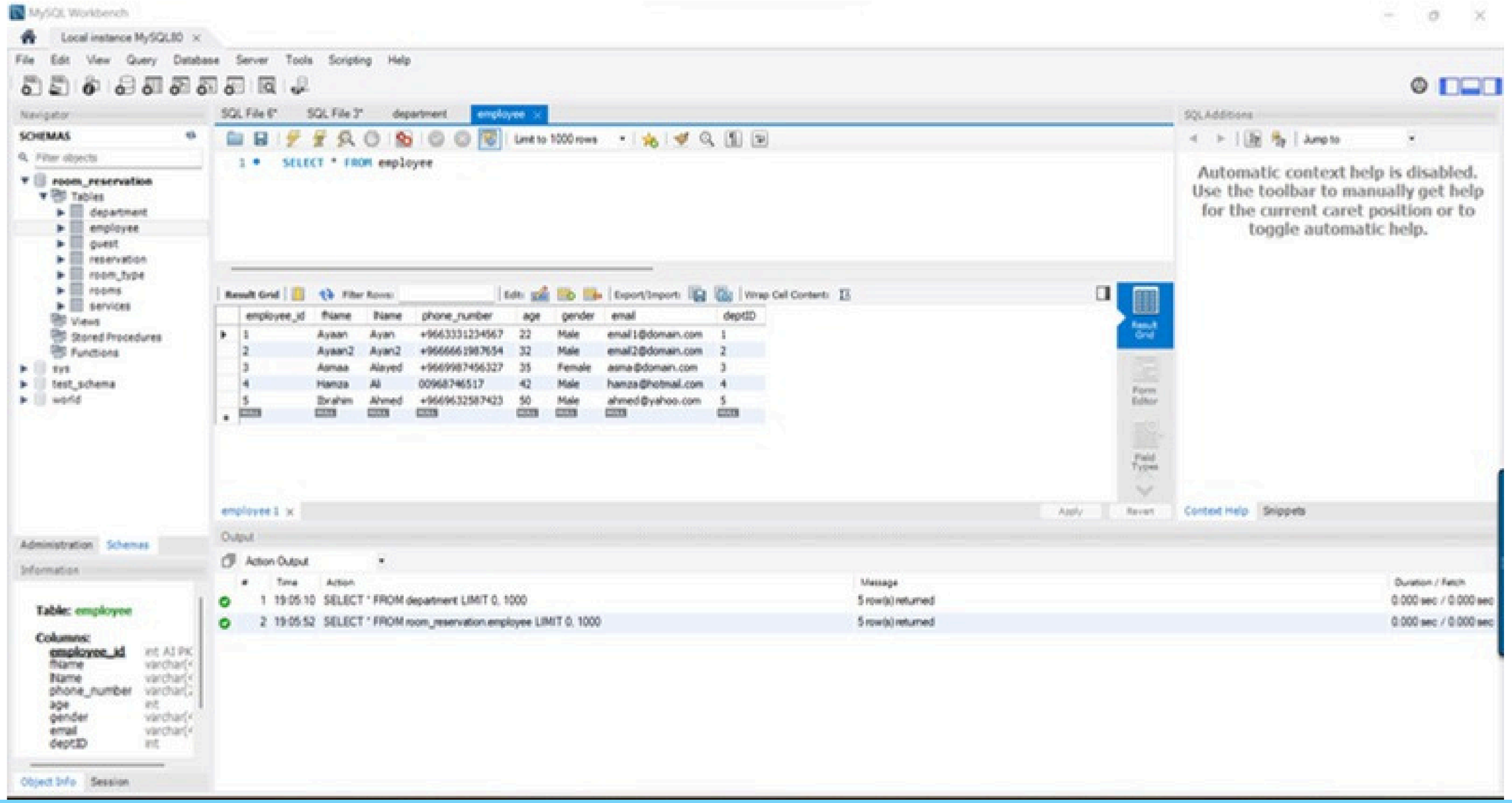
FOLLOWING SCREEN SHOT SHOWS FIVE INSERTED RECORDS IN DEPARTMENT TABLE.



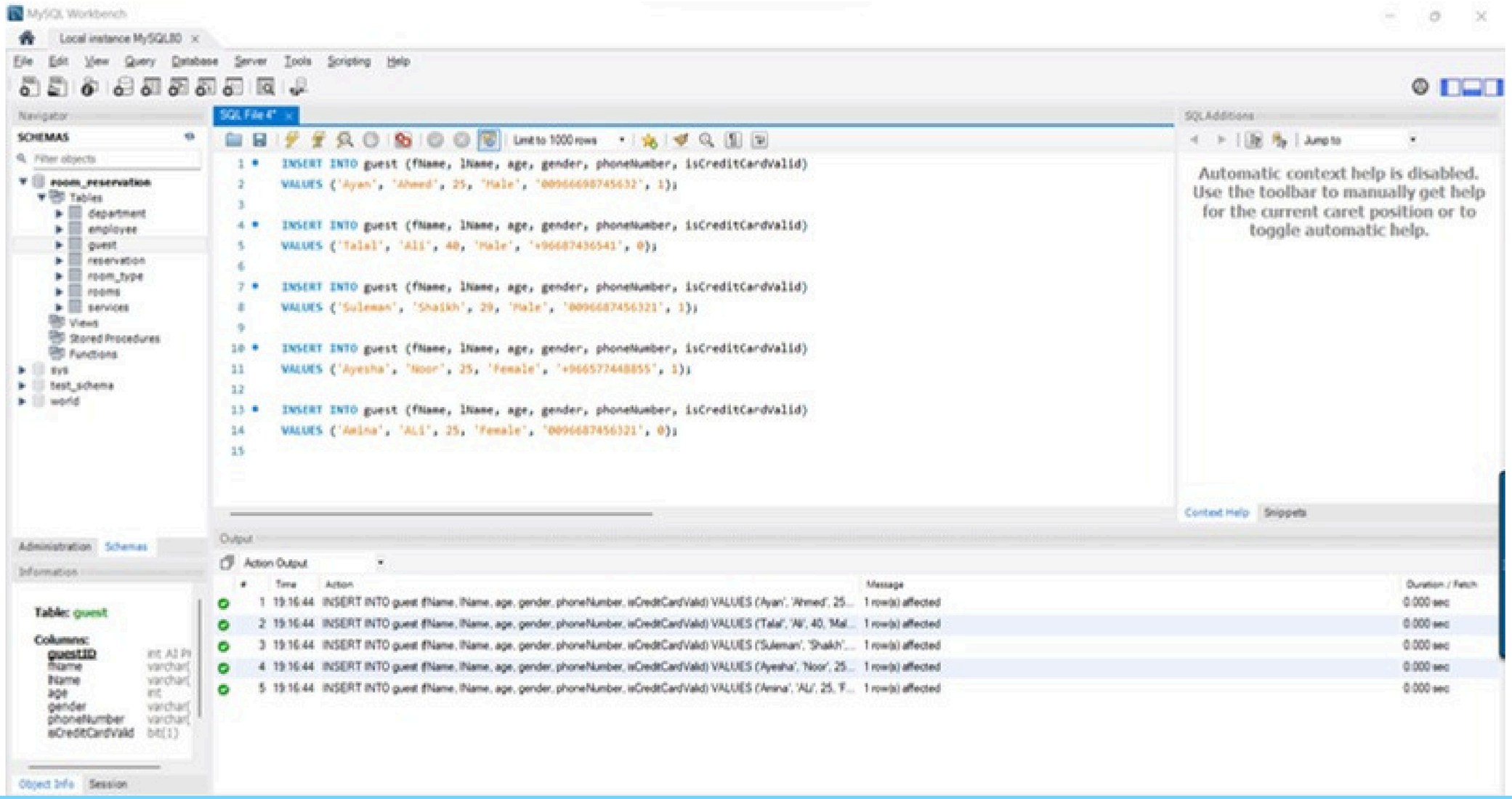
INSERT COMMAND FOR EMPLOYEE TABLE:
BELOW SCREEN SHOT SHOWS INSERT COMMANDS FOR EMPLOYEE TABLE.



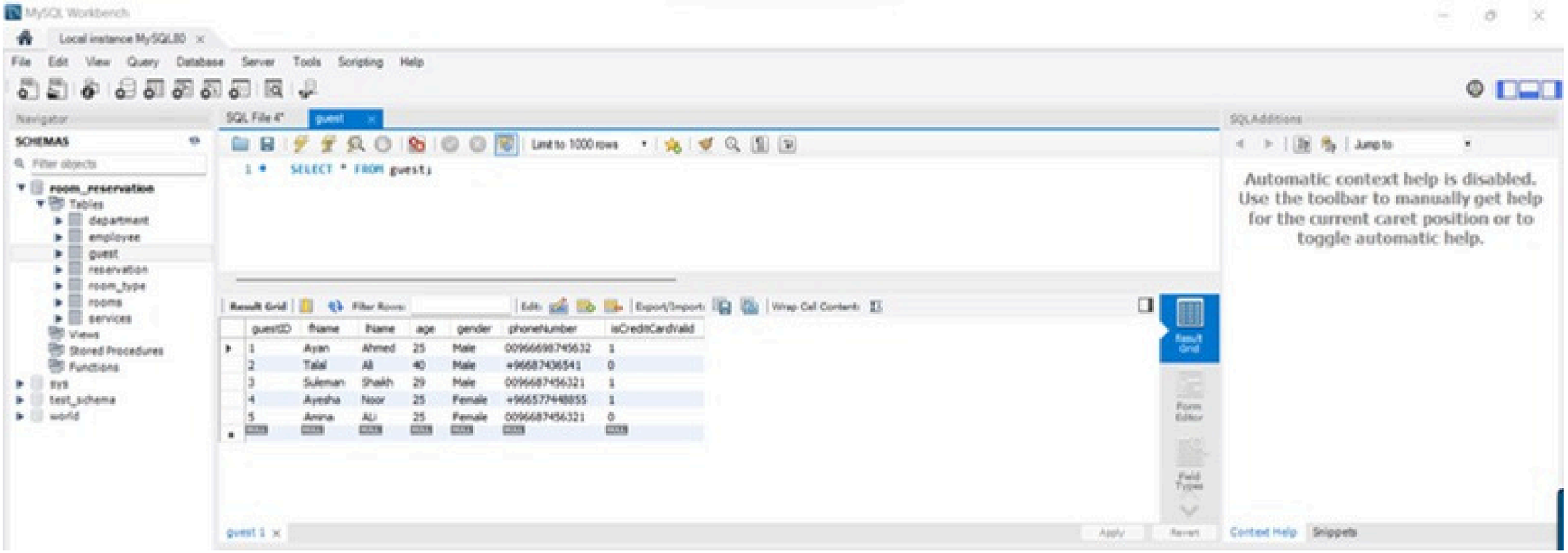
FOLLOWING SCREEN SHOT SHOWS FIVE INSERTED RECORDS IN EMPLOYEE TABLE.



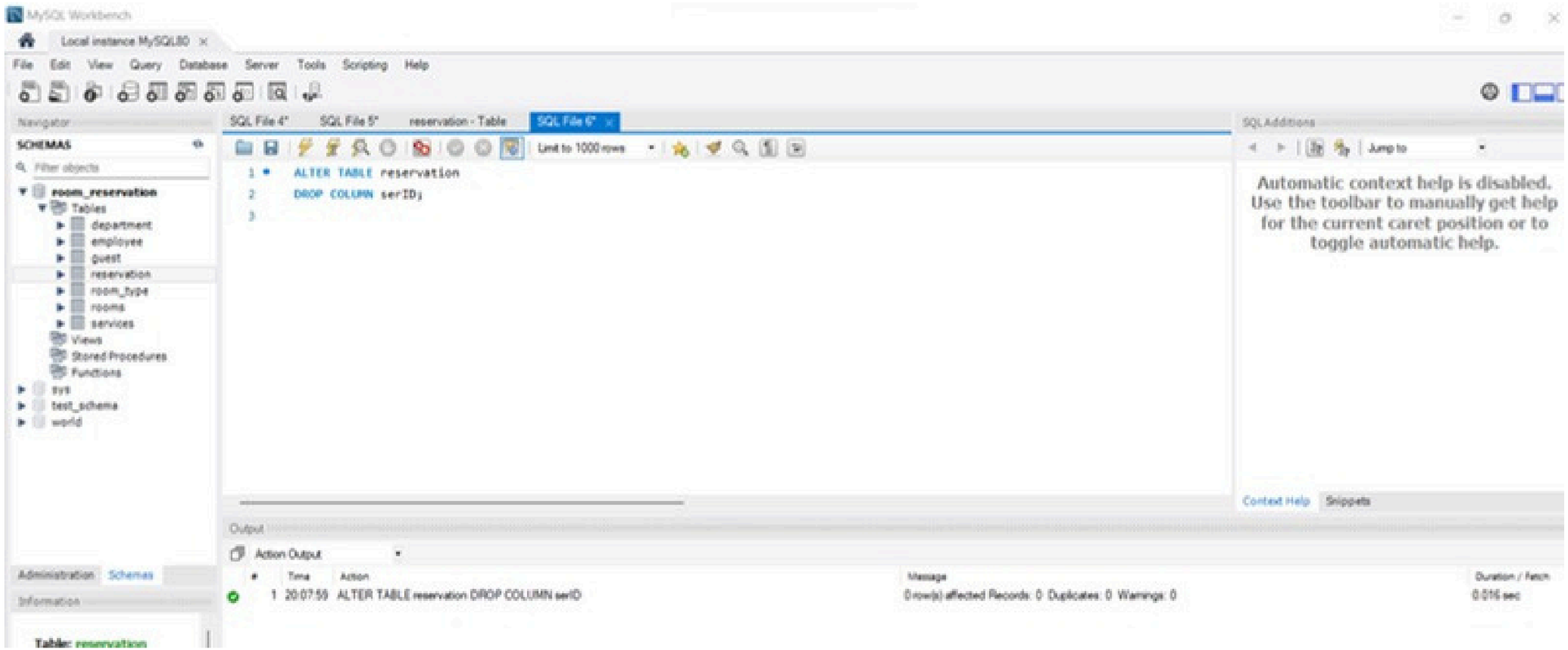
INSERT COMMANDS FOR GUEST TABLE:



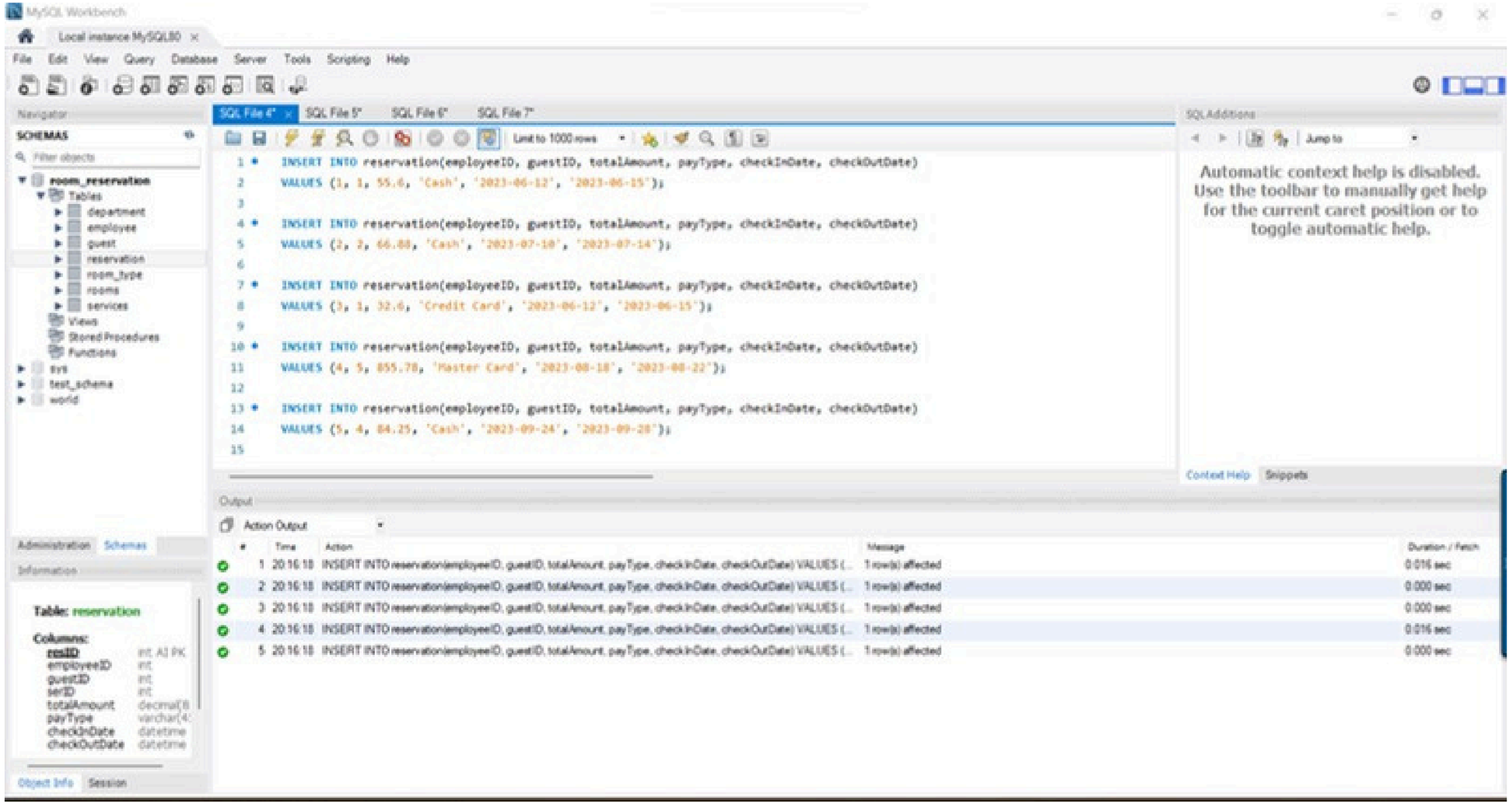
FOLLOWING SCREEN SHOT SHOWS FIVE INSERTED RECORDS IN GUEST TABLE.



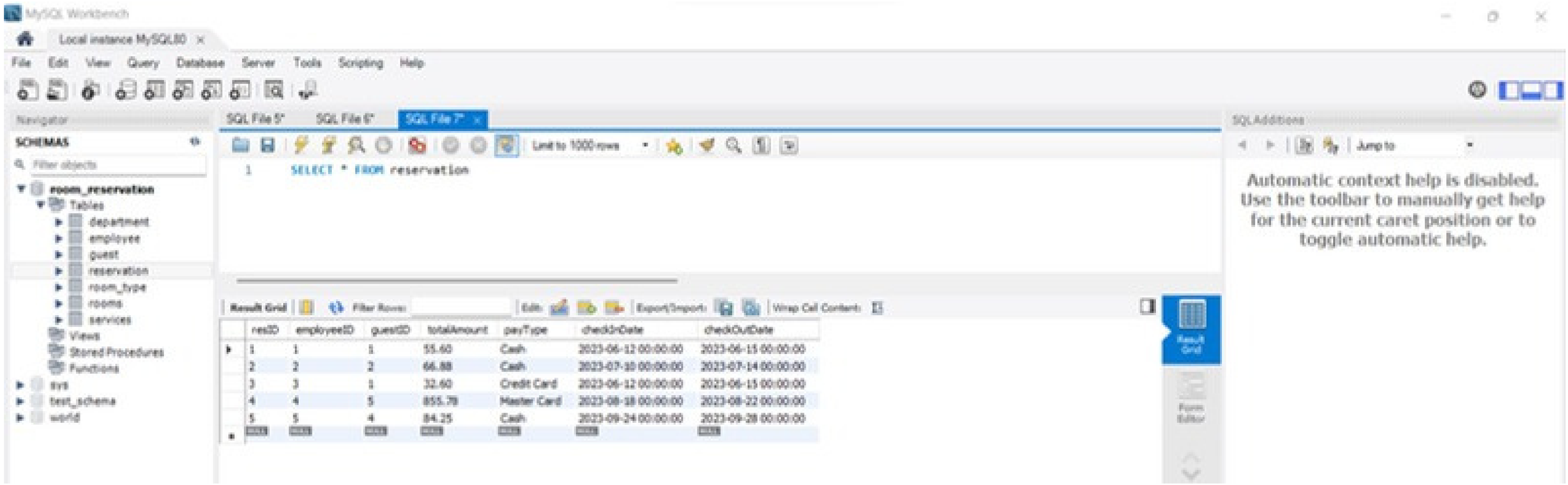
RESERVATION TABLE:
THERE IS NO NEED FOR SERID COLUMN IN THIS TABLE BECAUSE THIS TABLE PRIMARY KEY IS FOREIGN KEY OF SERVICE TABLE, SO DROPPING THE SERID COLUMN FROM RESERVATION TABLE.



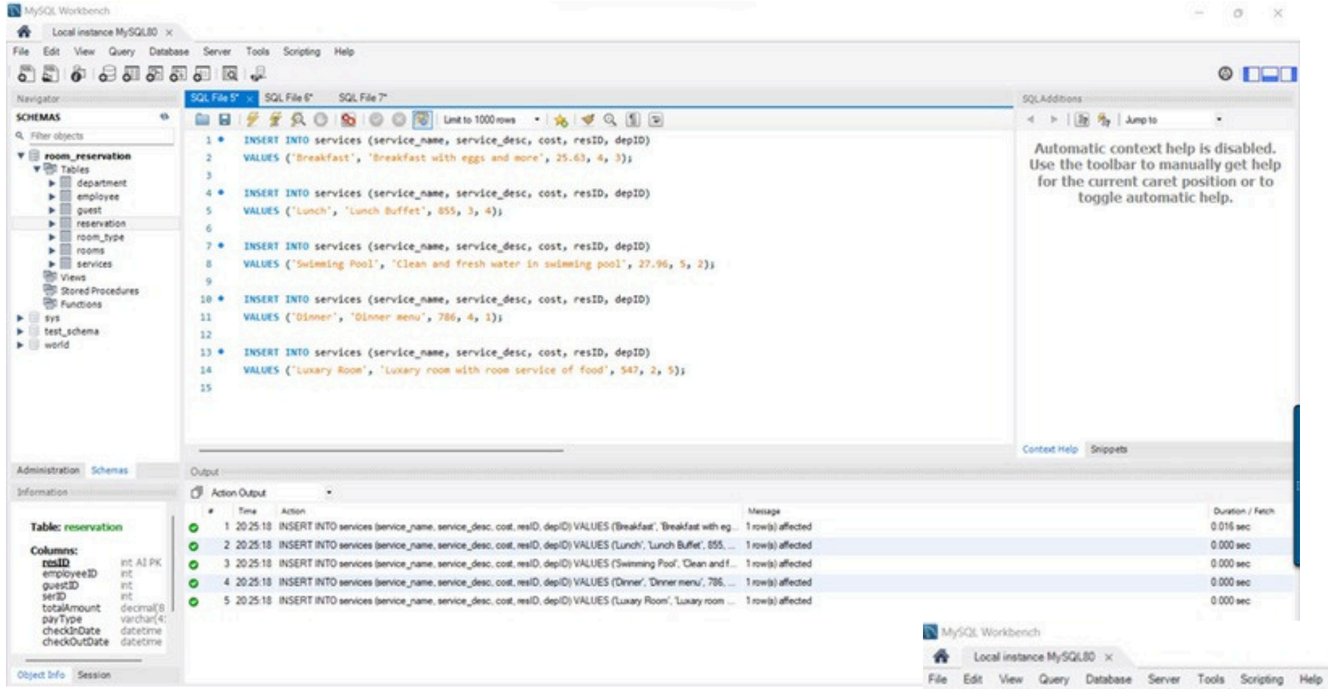
INSERT COMMANDS TO INSERT DATA IN RESERVATION TABLE.

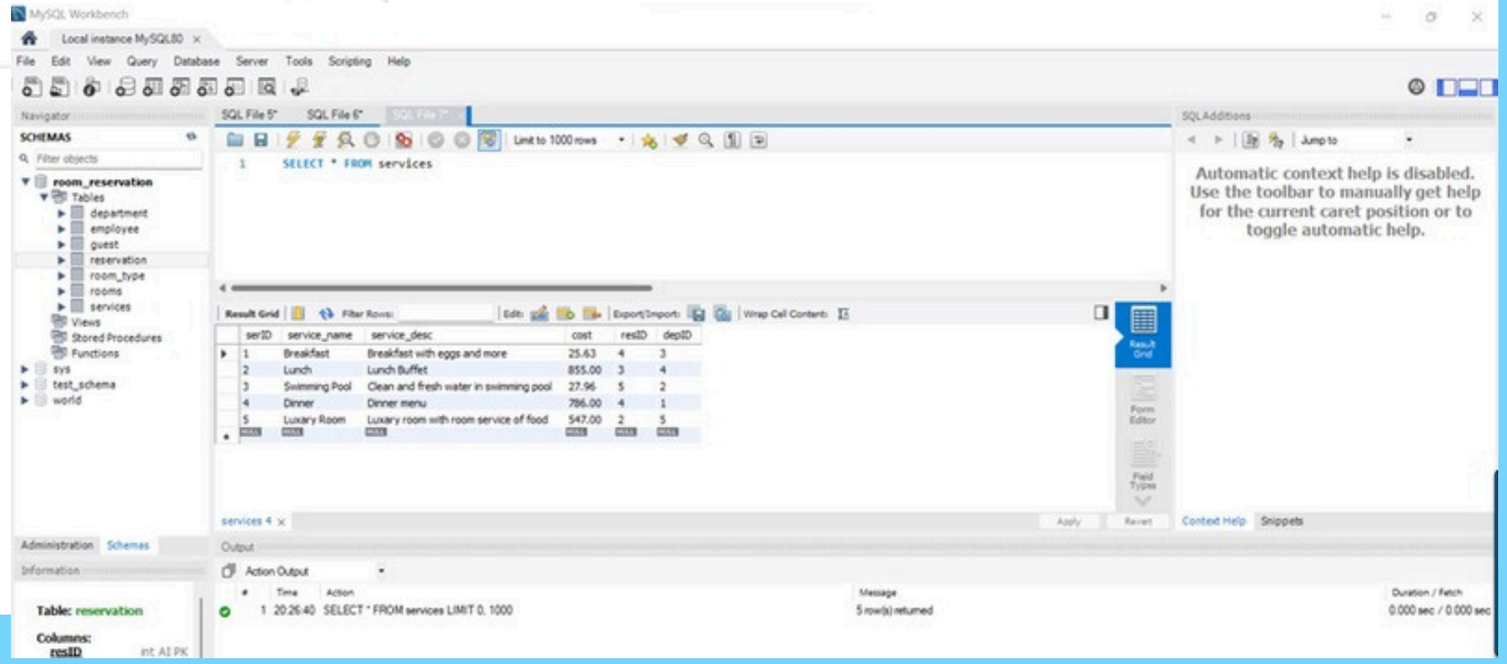


FOLLOWING SCREEN SHOT SHOWS FIVE INSERTED RECORDS IN RESERVATION TABLE.

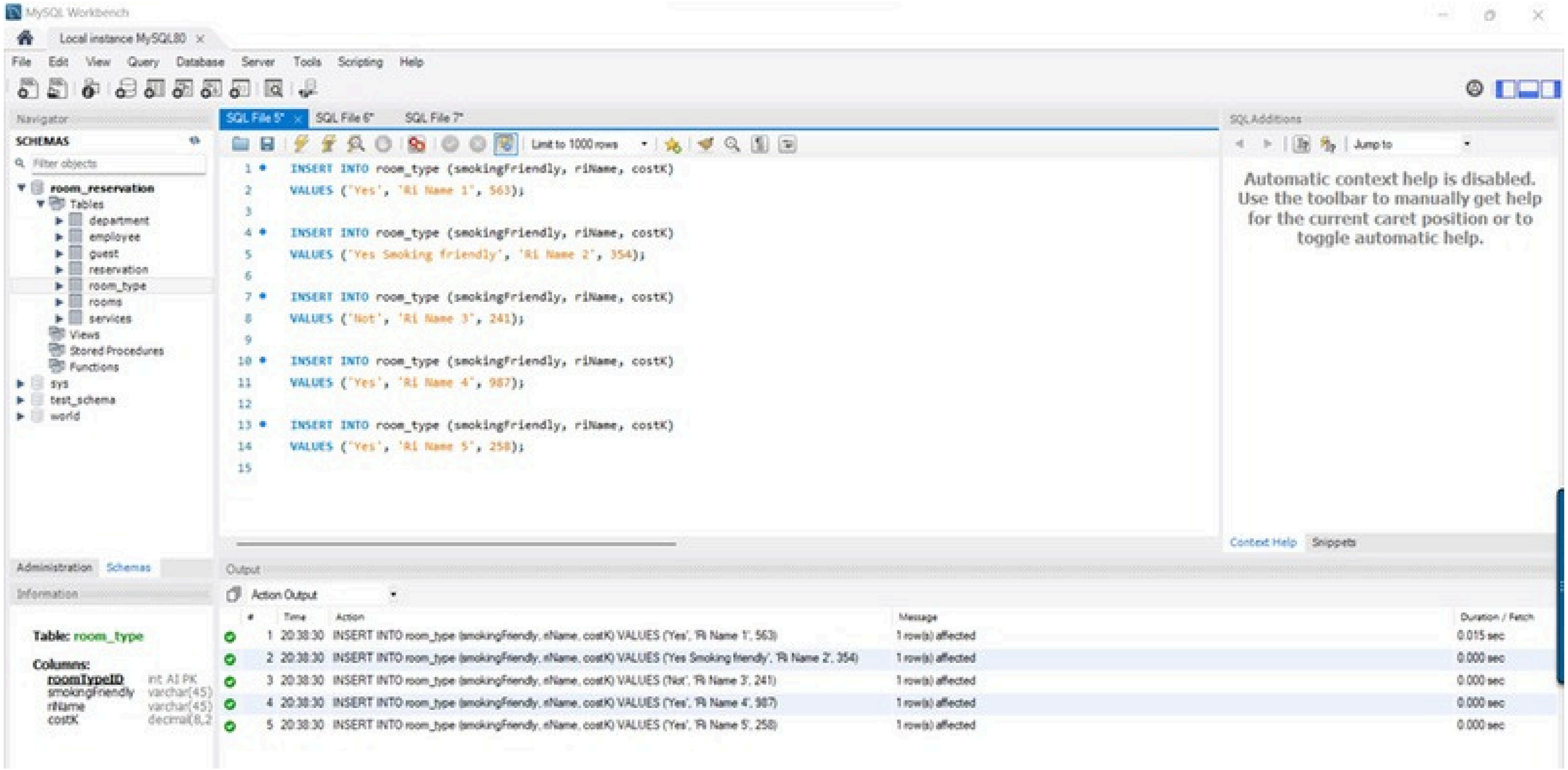


SERVICES TABLE:

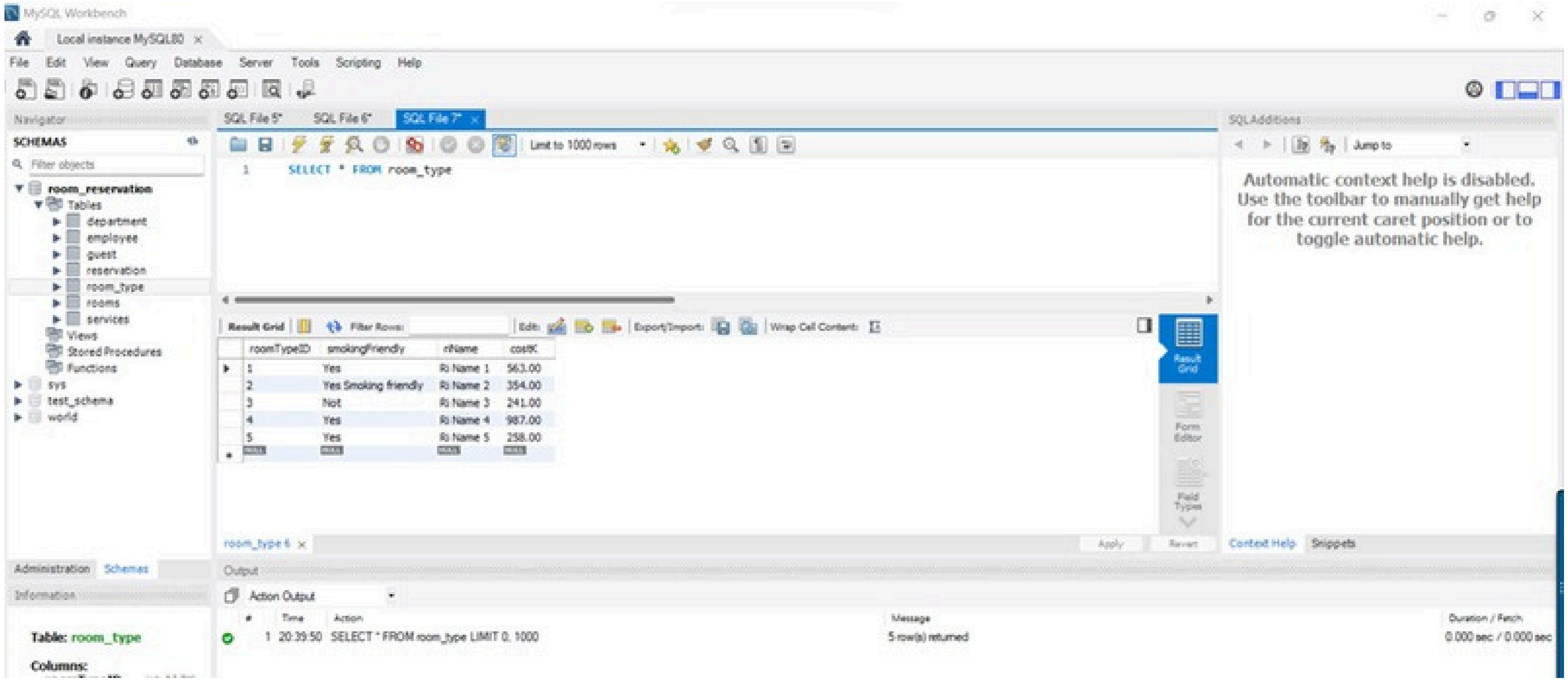




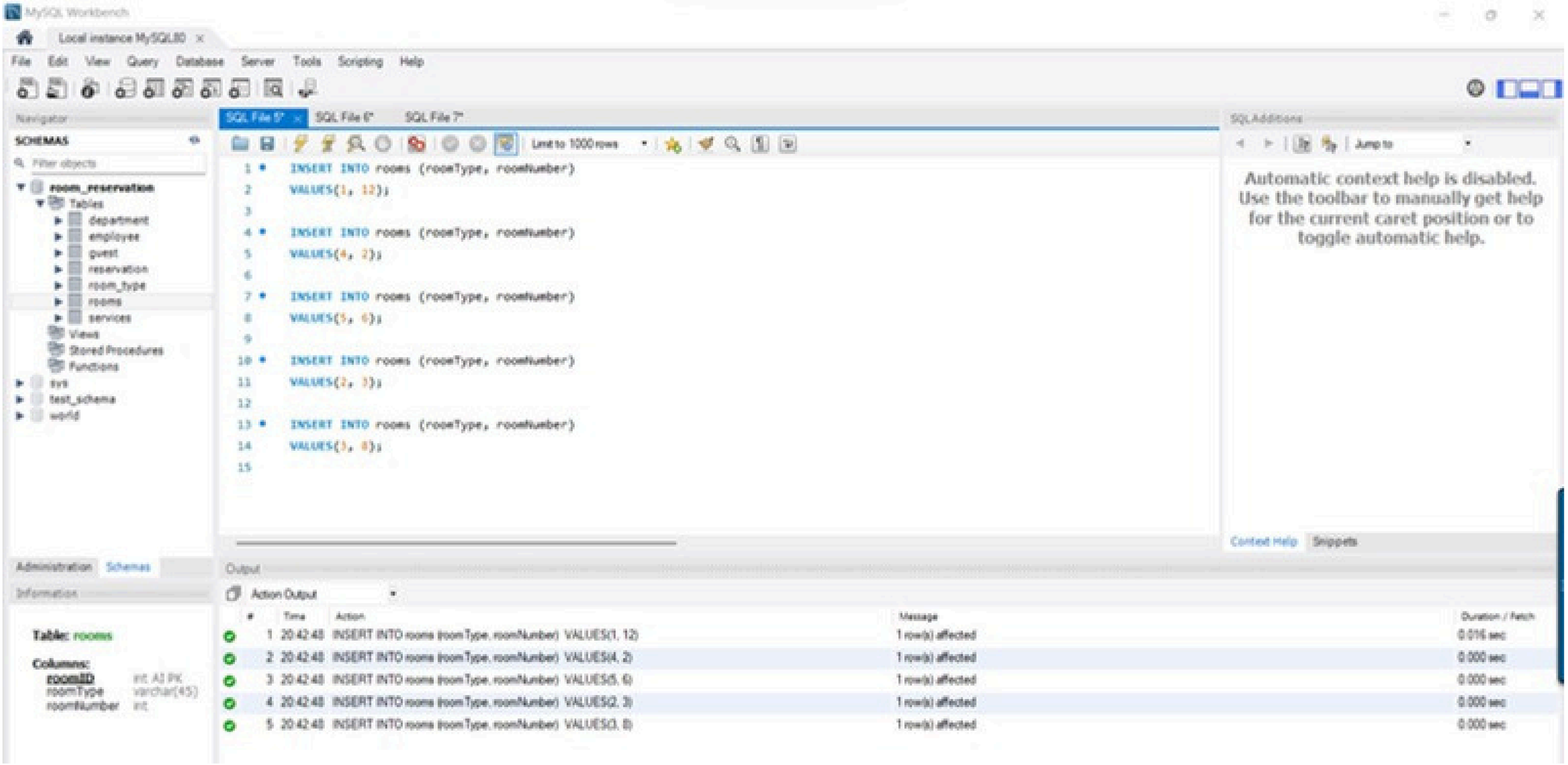
INSERT COMMANDS FOR ROOMTYPE TABLE:



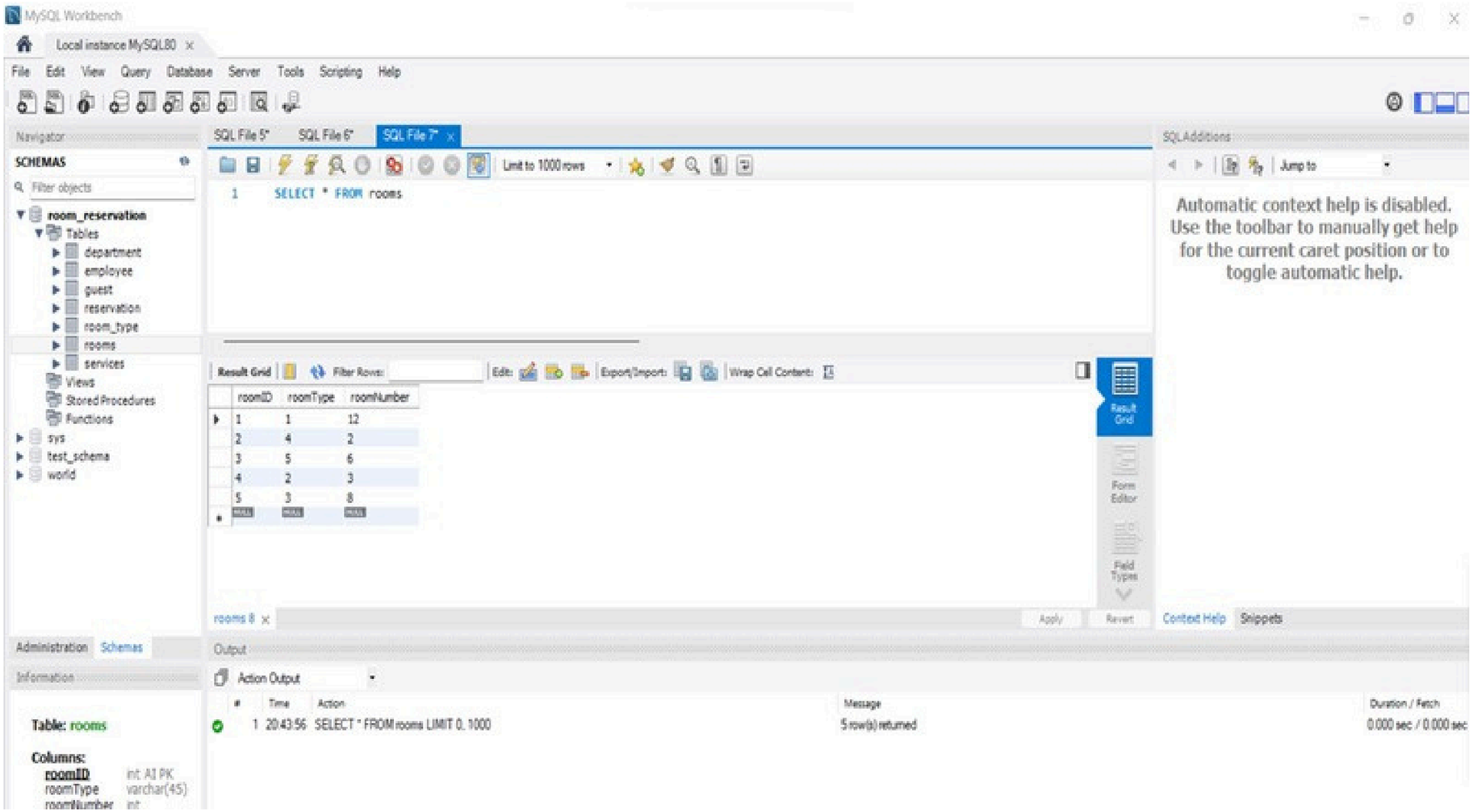
BELOW SCREEN SHOT SHOWS ROOM_TYPE TABLE RECORDS.



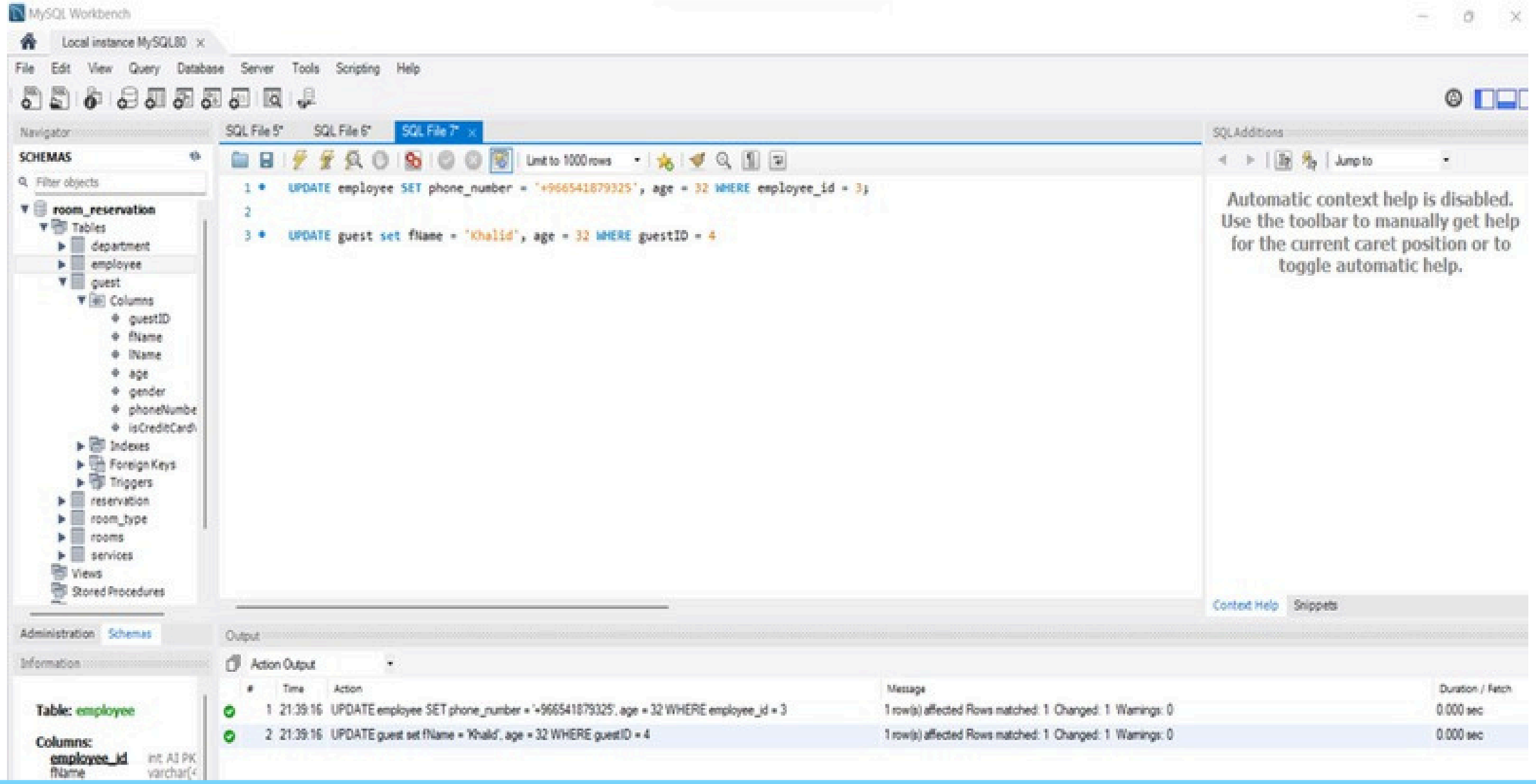
INSERT COMMANDS FOR ROOMS TABLE:



BELOW SCREEN SHOT SHOWS ROOMS TABLE RECORDS.



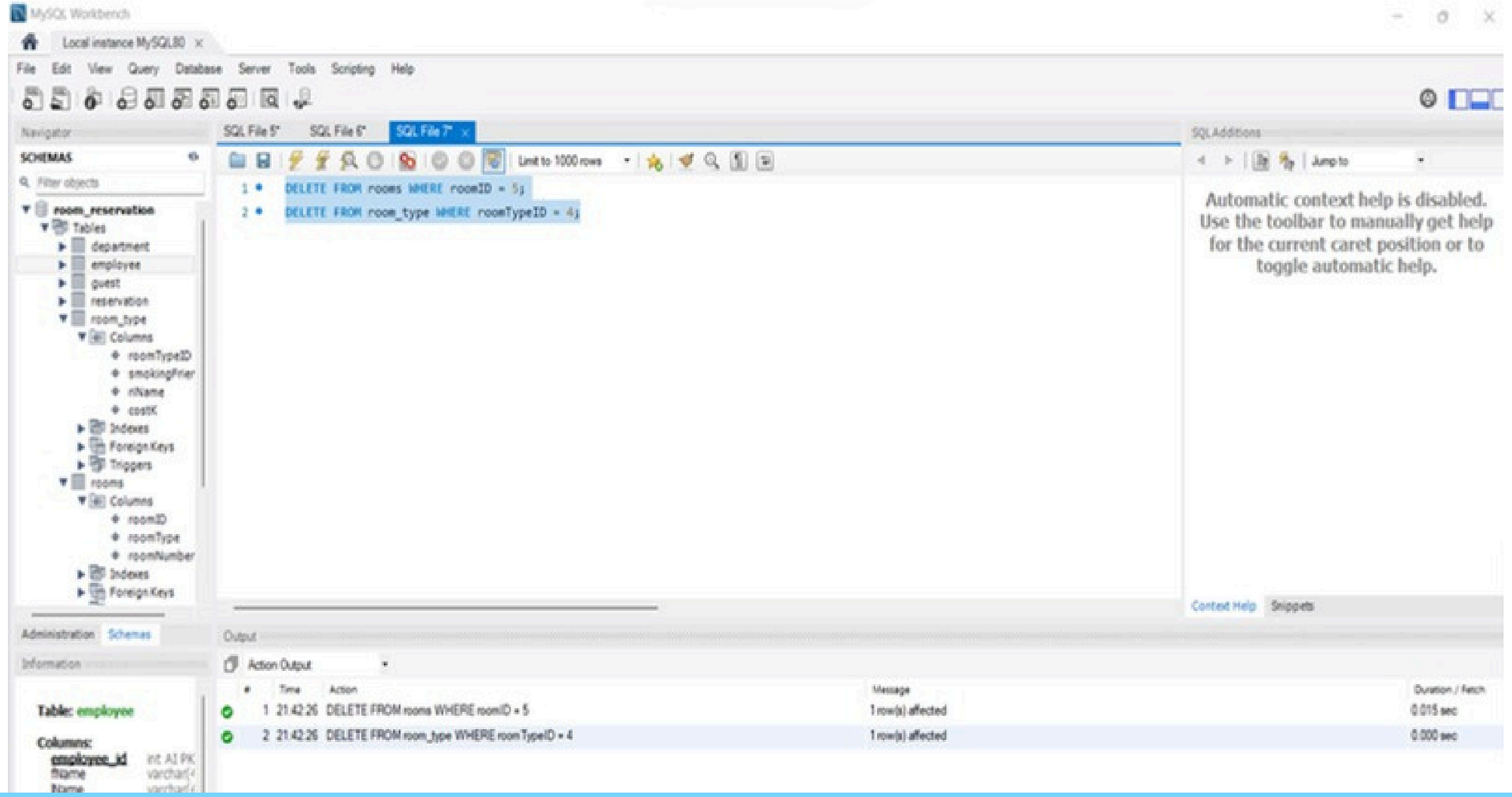
UPDATE QUERIES/COMMANDS:



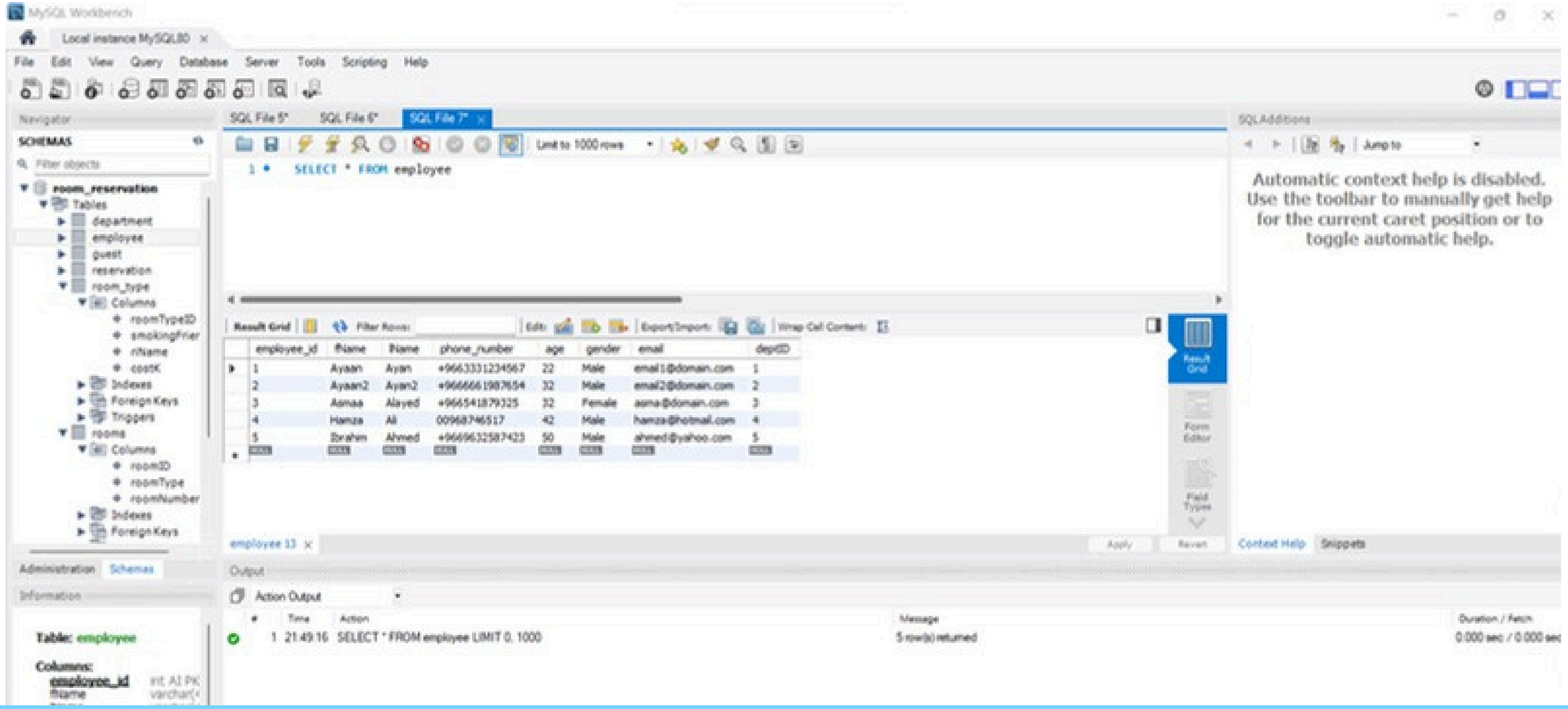
DELETE COMMANDS:

FOLLOWING QUERY DELETE THE RECORD OF ROOMS TABLE WHERE ROOMID IS EQUAL TO FIVE.

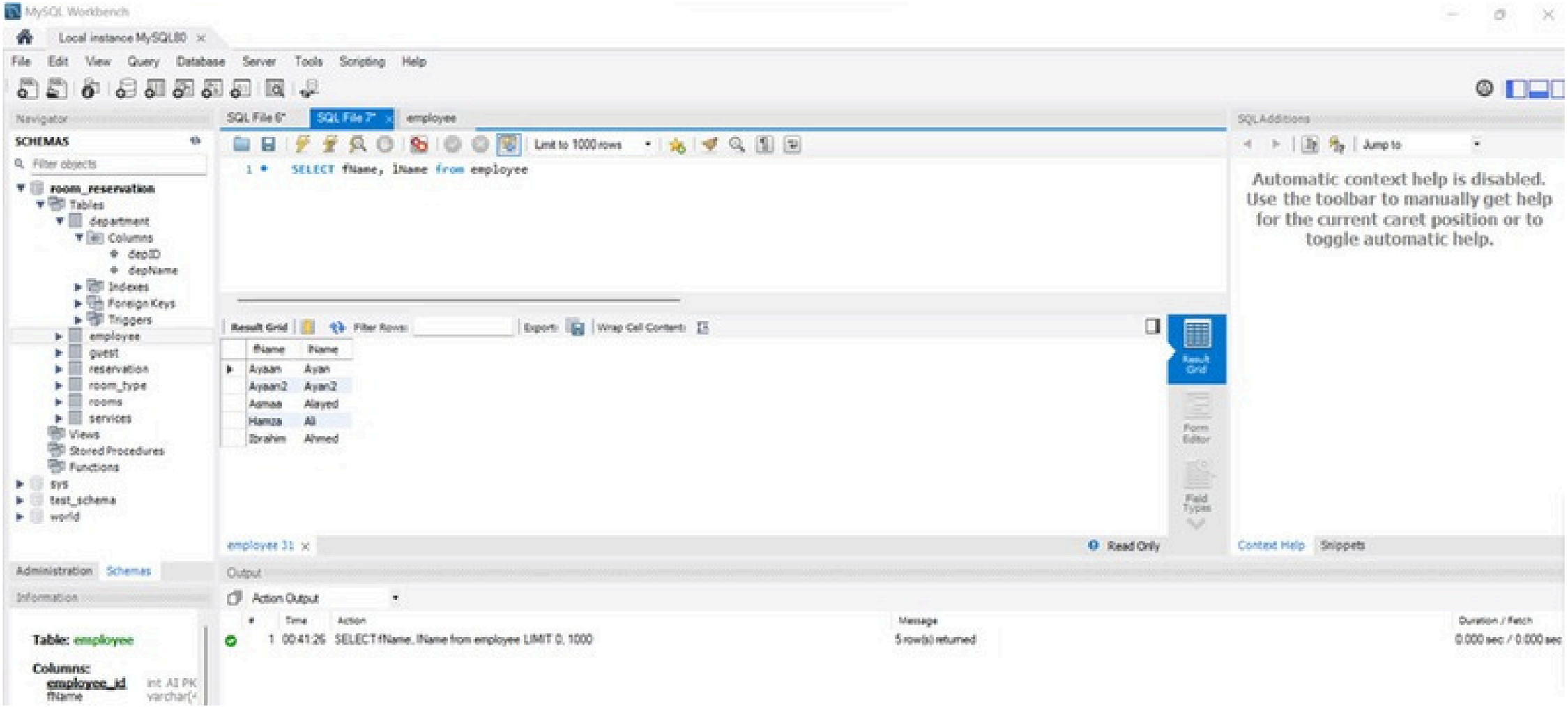
BELOW IS THE MYSQL WORKBENCH SCREEN SHOT.



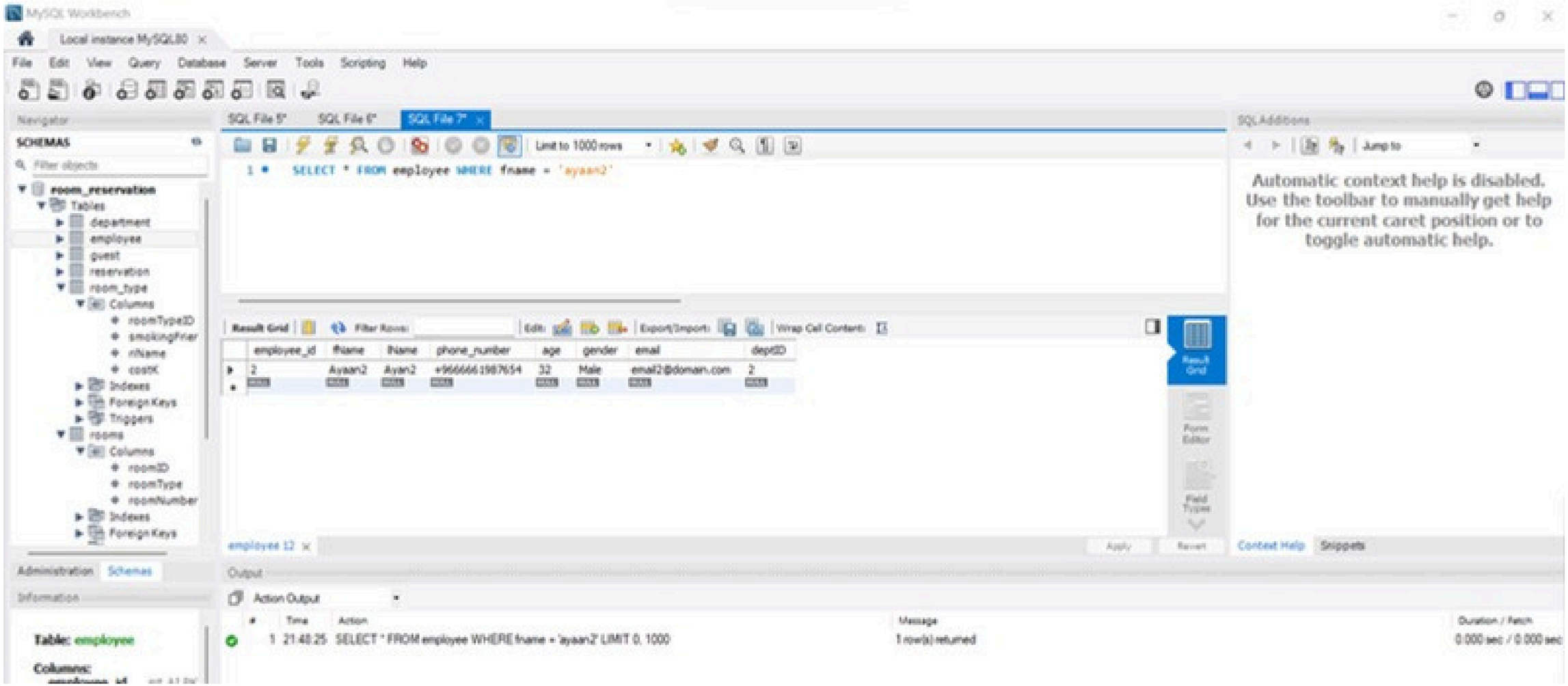
SQL QUERIES USING SELECT COMMAND:
SELECT ALL COLUMNS AND RECORDS COMMAND:
FOLLOWING QUERY RETRIEVE ALL COLUMNS AND ROWS OF EMPLOYEE TABLE.



SELECT SPECIFIC COLUMNS:
FOLLOWING IS THE QUERY TO SELECT SPECIFIC COLUMNS OF THE TABLE:

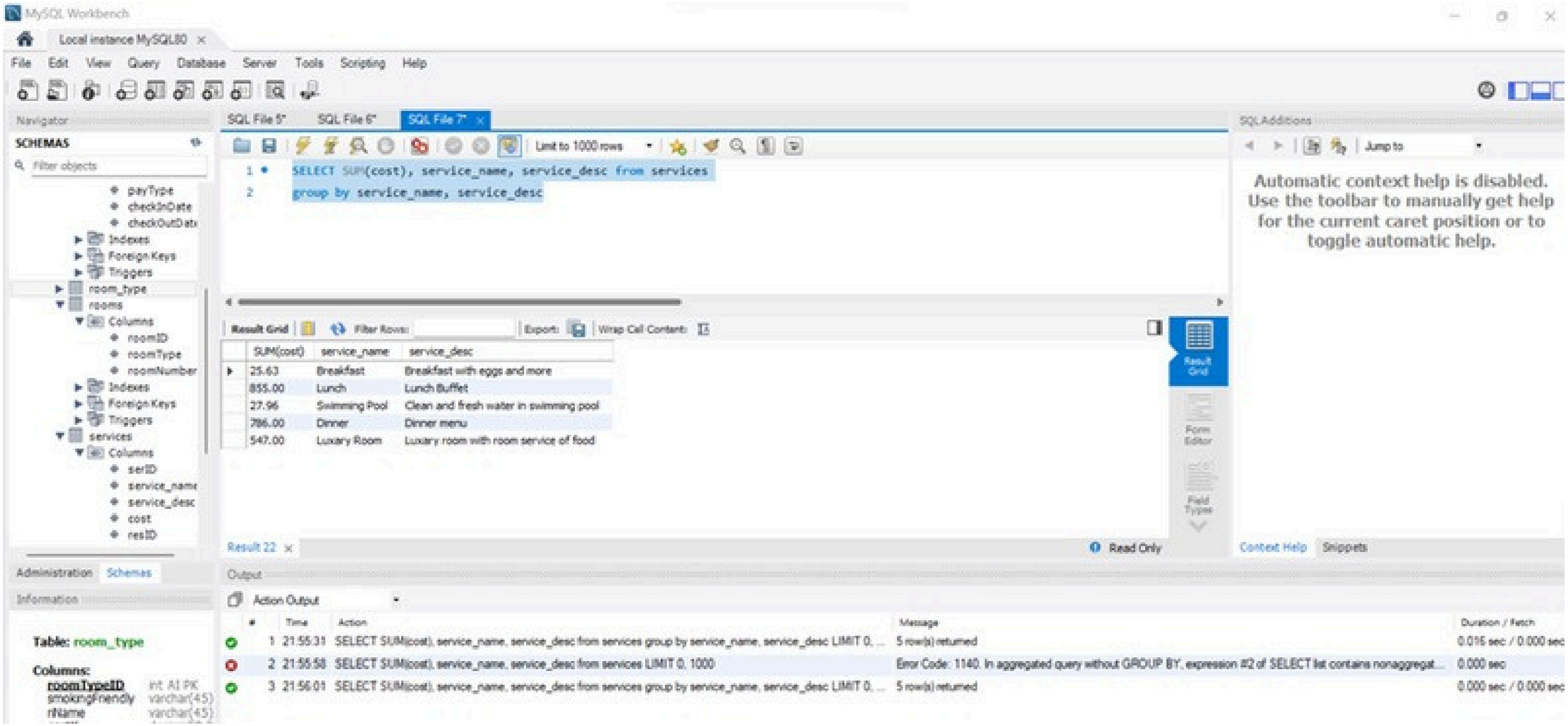


SELECT WITH WHERE COMMAND:
FOLLOWING QUERY RETRIEVE ALL COLUMNS OF EMPLOYEE TABLE AND THOSE RECORDS OF THE EMPLOYEE WHICH HAS FIRST NAME AS AYAAN2.

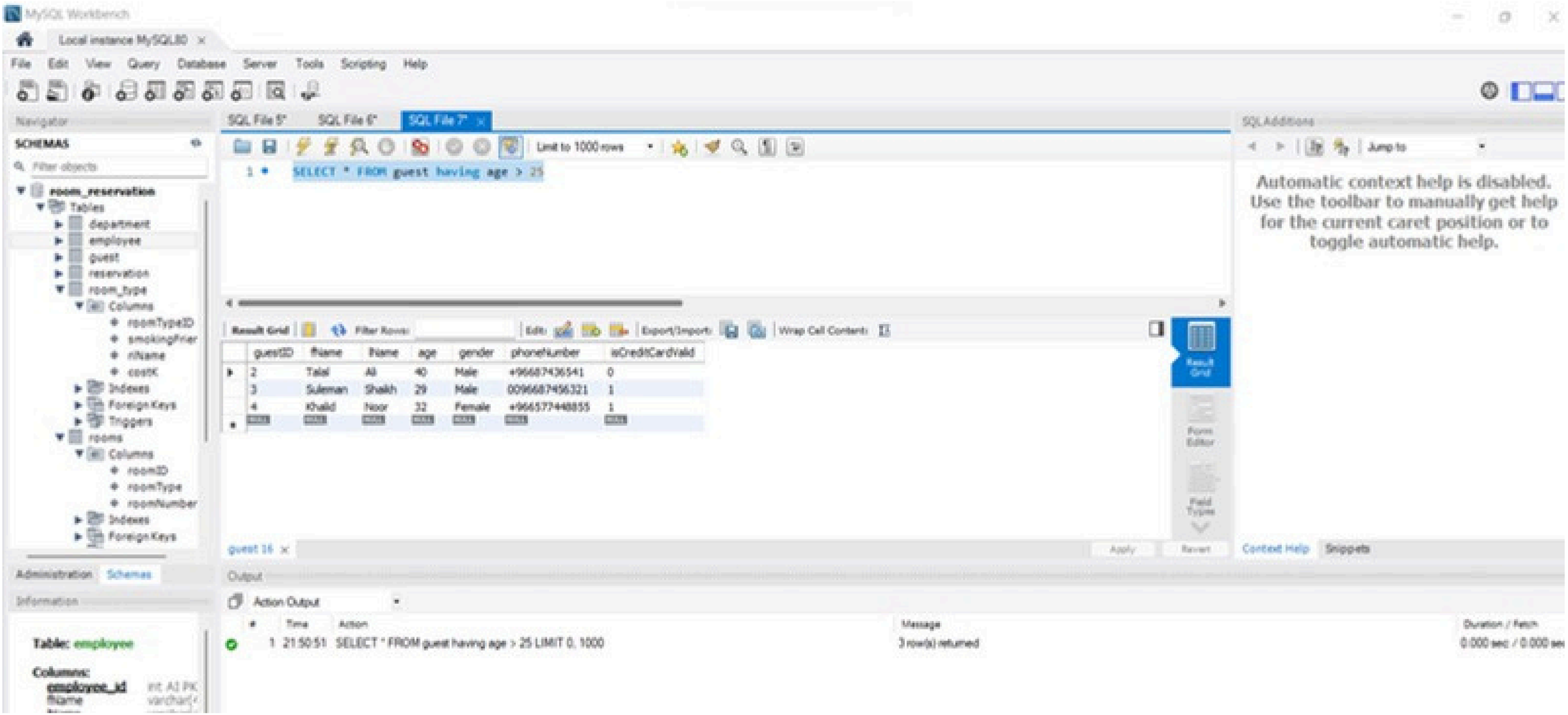


GROUP BY:

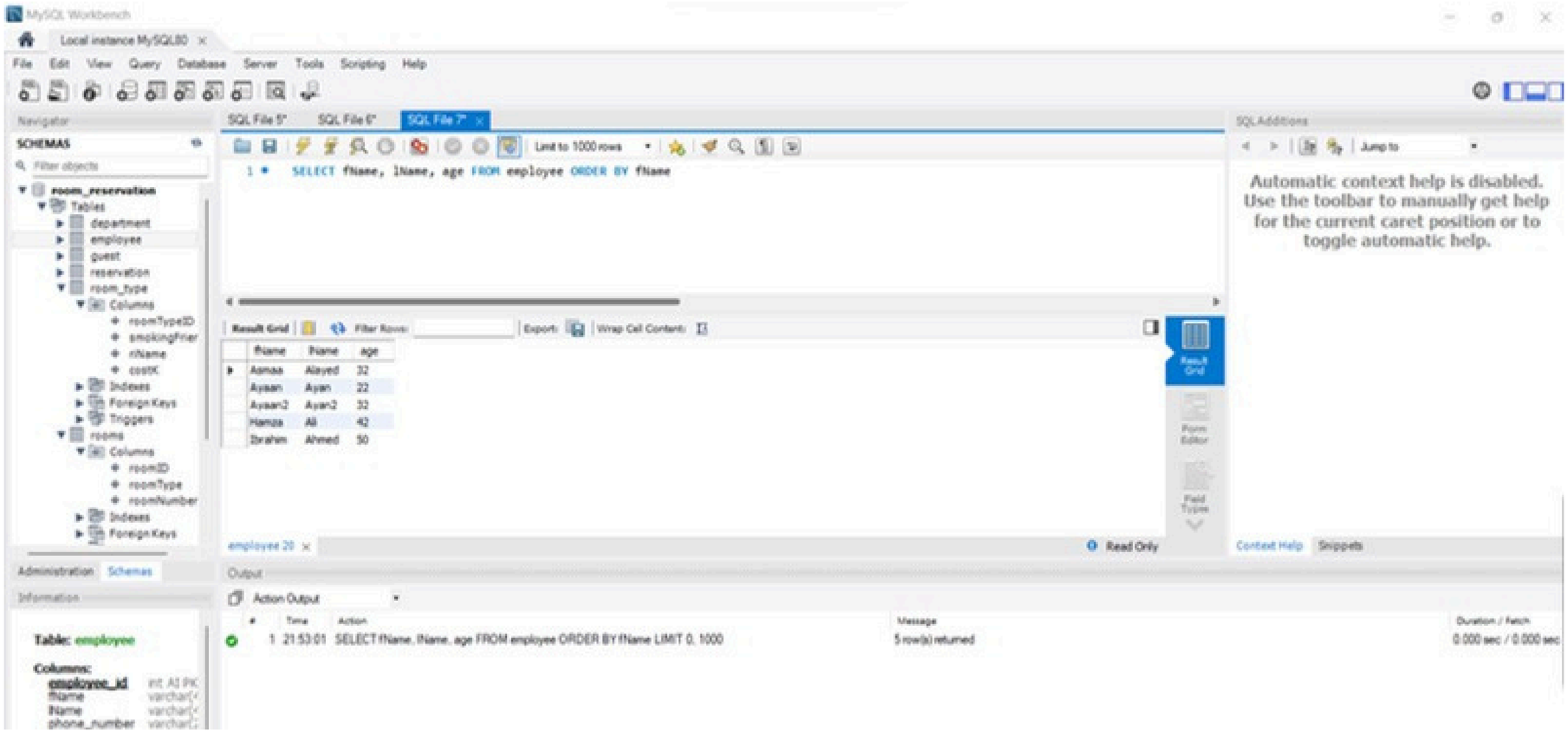
THE GROUP BY COMMAND IS NORMALLY USED WITH AGGREGATE FUNCTIONS (COUNT() , MAX() , MIN() , SUM() , AVG()) TO GROUP THE RESULT-SET BY ONE OR MORE COLUMNS.



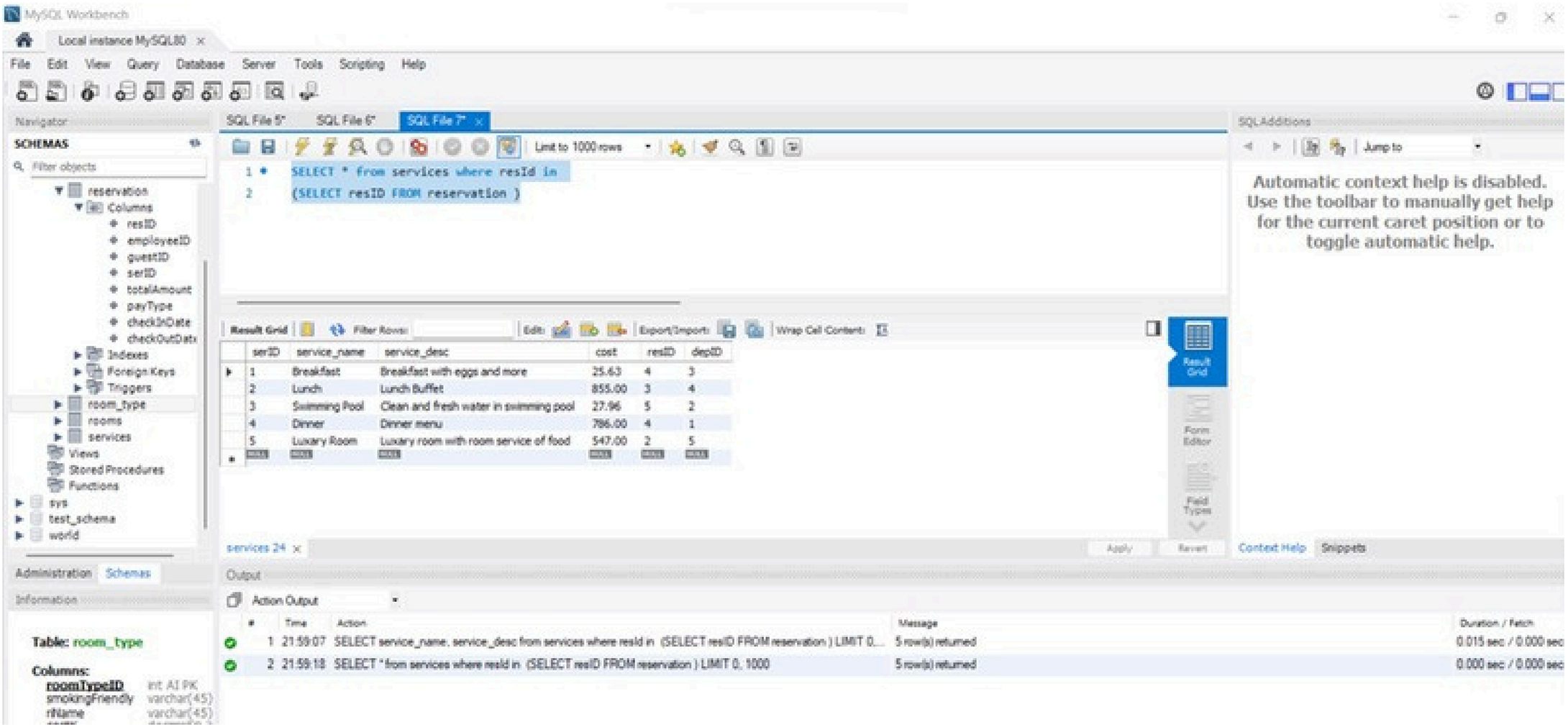
HAVING IN SELECT COMMAND:



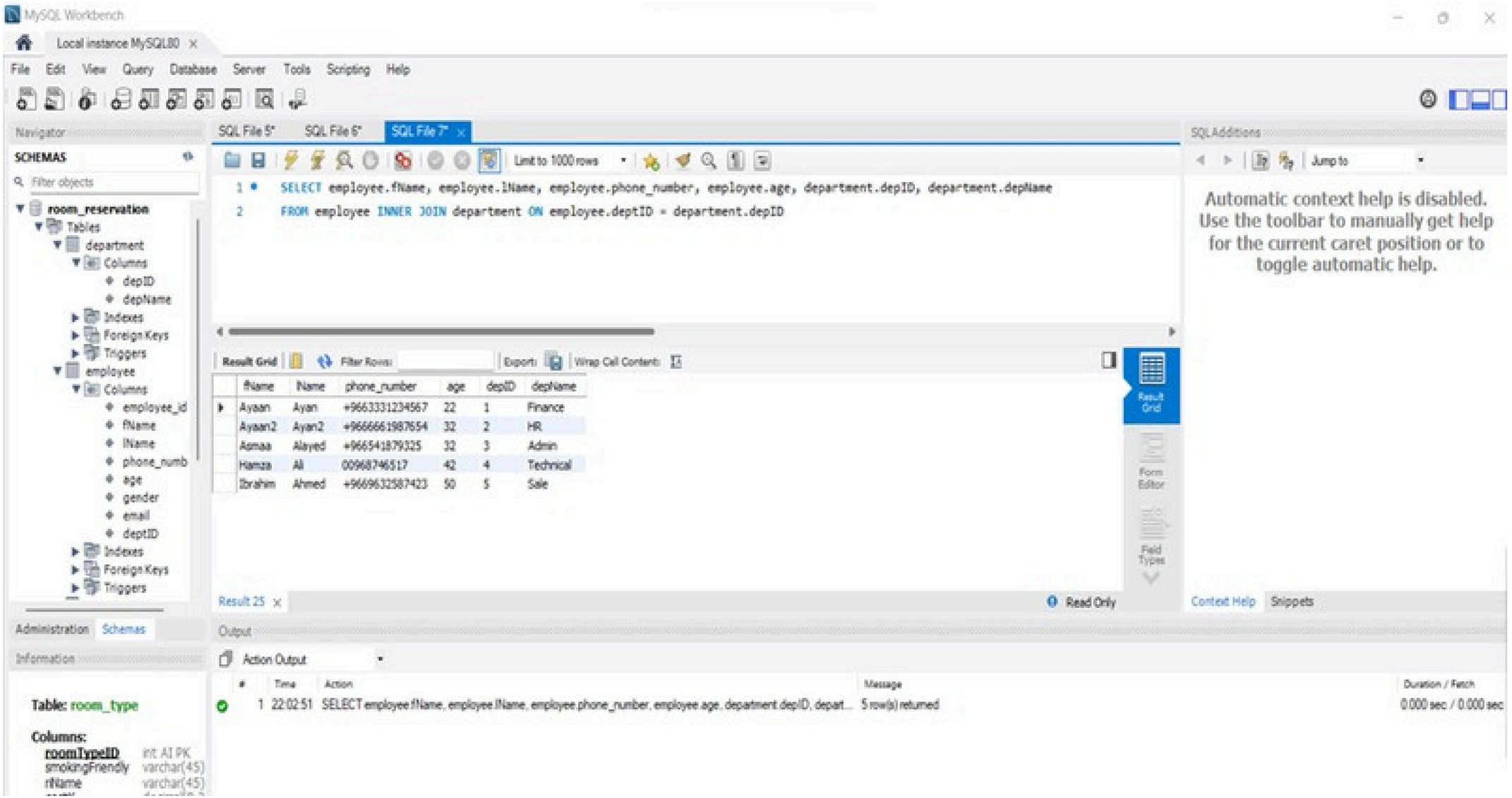
ORDER BY:


MySQL Workbench
Local instance MySQL80
File Edit View Query Database Server Tools Scripting Help
SQL File 5* SQL File 6* SQL File 7*
SCHEMAS
room_reservation
Tables
department
employee
guest
reservation
room_type
rooms
SELECT fname, lname, age FROM employee ORDER BY fname
fname lname age
Aamaa Alayed 32
Ayaan Ayan 22
Ayaan2 Ayan2 32
Hamza Ali 42
Ibrahim Ahmed 50
Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.
Output
Action Output
1 21:53:01 SELECT fname, lname, age FROM employee ORDER BY fname LIMIT 0, 1000
5 row(s) returned
0.000 sec / 0.000 sec
Table: employee
Columns:
employee_id
fname
lname
phone_number

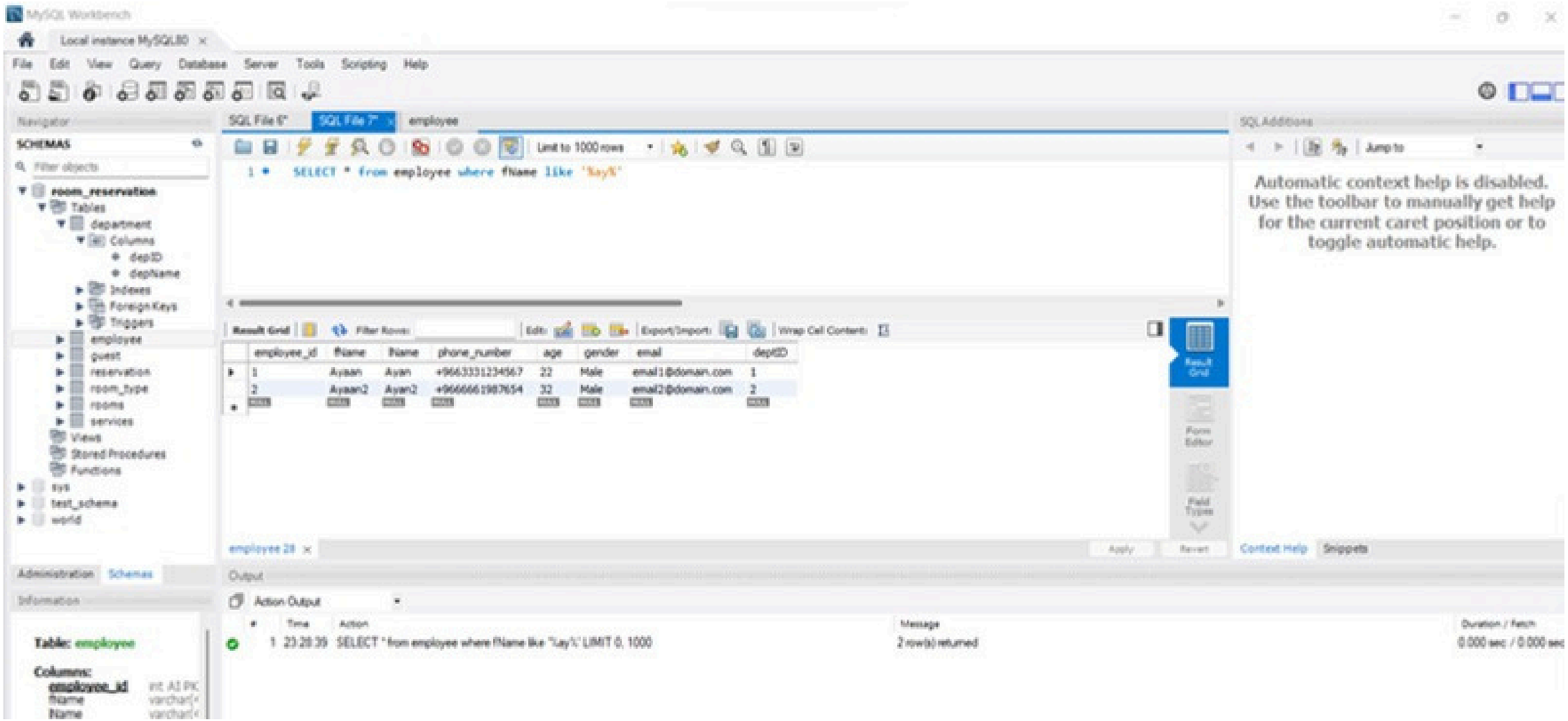
SUBQUERY:


MySQL Workbench
Local instance MySQL80
File Edit View Query Database Server Tools Scripting Help
Navigator
SCHEMAS
Filter objects
reservation
Columns
resID
employeeID
guestID
serID
totalAmount
payType
checkInDate
checkOutDate
Indexes
Foreign Keys
Triggers
room_type
rooms
services
Views
Stored Procedures
Functions
sys
test_schema
world
Administration Schemas
Information
Table: room_type
Columns:
roomTypeID int AI PK
smokingFriendly varchar(45)
rtName varchar(45)
SQL File 5* SQL File 6* SQL File 7*
Limit to 1000 rows
1 SELECT * from services where resId in
2 (SELECT resID FROM reservation)
Result Grid Filter Rows: Edit: Export/Import: Wrap Cell Content:
serID service_name service_desc cost resID depID
1 Breakfast Breakfast with eggs and more 25.63 4 3
2 Lunch Lunch Buffet 855.00 3 4
3 Swimming Pool Clean and fresh water in swimming pool 27.96 5 2
4 Dinner Dinner menu 786.00 4 1
5 Luxary Room Luxary room with room service of food 547.00 2 5
services 24
Apply Revert
SQL Additions
Jump to
Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.
Context Help Snippets
Output
Action Output
Time Action Message Duration / Fetch
1 21:59:07 SELECT service_name, service_desc from services where resId in (SELECT resID FROM reservation) LIMIT 0... 5 row(s) returned 0.015 sec / 0.000 sec
2 21:59:18 SELECT * from services where resId in (SELECT resID FROM reservation) LIMIT 0, 1000 5 row(s) returned 0.000 sec / 0.000 sec

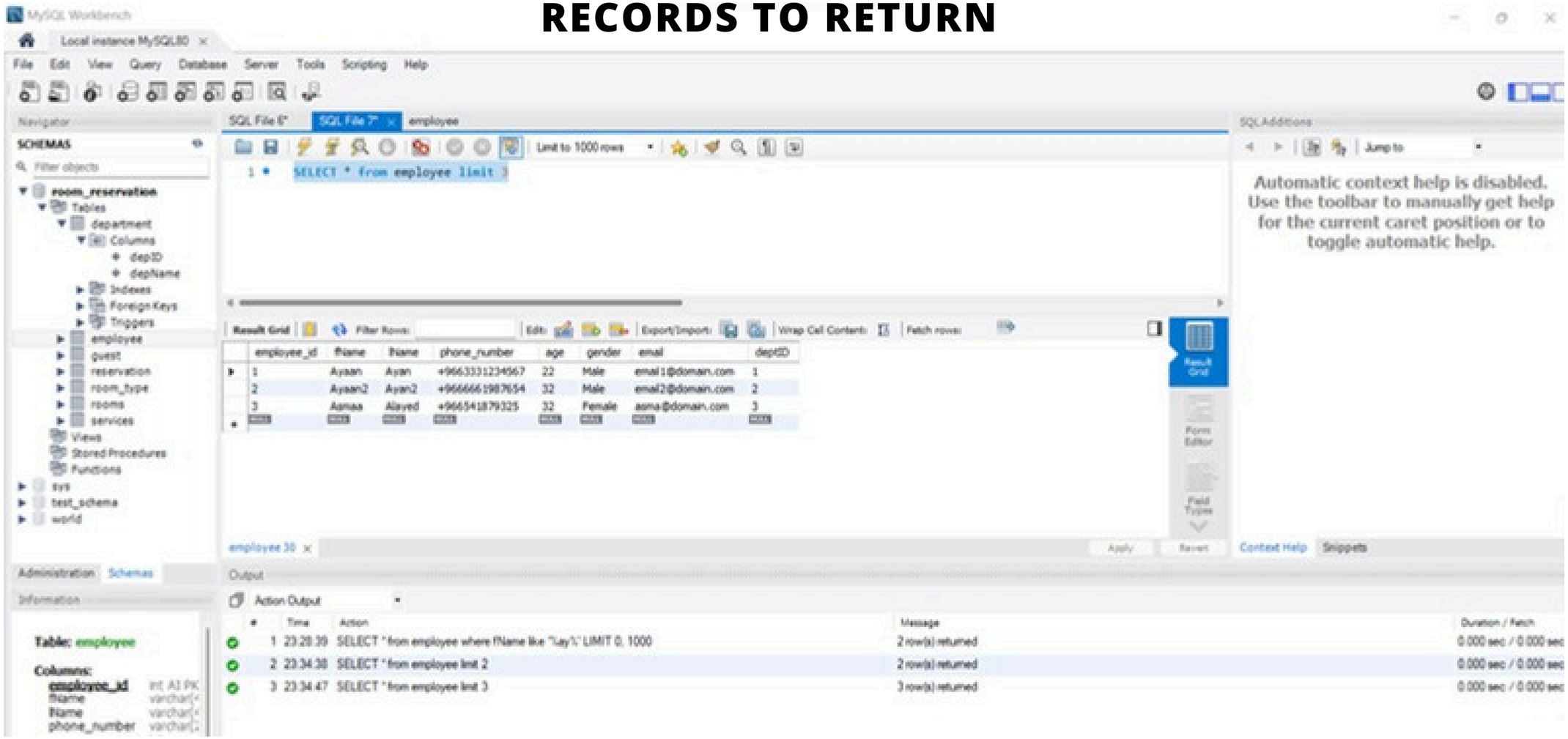
INNER JOIN:


MySQL Workbench
Local instance MySQL80
File Edit View Query Database Server Tools Scripting Help
SQL File 5* SQL File 6* SQL File 7*
SCHEMAS
room_reservation
Tables
department
Columns
depID
depName
Indexes
Foreign Keys
Triggers
employee
Columns
employee_id
fName
lName
phone_numb
age
gender
email
deptID
Indexes
Foreign Keys
Triggers
SELECT employee.fName, employee.lName, employee.phone_number, employee.age, department.depID, department.depName
FROM employee INNER JOIN department ON employee.deptID = department.depID
fName lName phone_number age depID depName
Ayaan Ayan +9663331234567 22 1 Finance
Ayaan2 Ayan2 +9666661987654 32 2 HR
Asmaa Alayed +966541879325 32 3 Admin
Hamza Ali 00968746517 42 4 Technical
Ibrahim Ahmed +9669632587423 50 5 Sale
Result 25
Read Only
Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.
Table: room_type
Columns:
roomTypeID int AI PK
smokingFriendly varchar(45)
rName varchar(45)
Action Output
1 22:02:51 SELECT employee.fName, employee.lName, employee.phone_number, employee.age, department.depID, depart... 5 row(s) returned
0.000 sec / 0.000 sec

LIKE:
LIKE IN SELECT QUERY RETRIEVES ALL RECORDS WHICH CONTAINS SPECIFIED TEXT/STRING.



LIMIT:
THE SELECT LIMIT COMMAND IS USED TO SPECIFY THE NUMBER OF RECORDS TO RETURN



THANK YOU FOR BEING WITH US IN THIS JOURNEY IT'S TIME TO SAY GOODBYE & BLOW OUT THE CANDLES