

Database Project

Info 6210

Chilkepalli Aashrith

NU ID : 001854964

Title : Database Design for Hotel Management System

Scope:

Most of the hotels need to have a proper integrated system built in order to always be functional and effective. As hotel business is a part of a competitive and brutal market , any inconvenience caused to the customers will effect them in the long run . The first step in achieving this would be the hotel should be effective in providing the techniques . Unfortunately , Many hotel management systems lack this as their systems are ineffective , redundant and not economical.

Solution :

Managing hotel system has a technology system associated with it . With so many records at stake , designing a database that is really effective is imperative . The objective of this project is to build an error free and efficient database. So no records are lost or corrupted.

Entities:

The main entities in the project proposed are :

Customer : Details of the Customer(Guest)such as FirstName, LastName, City,Country and Email.

Customer Allotted Room : This entity records data pertaining to every customer who has been assigned to a particular room.

Room : This entity consists all the details of the room the kind of room and its status whether its Available or not .

Reservation : A customer can register for a room , they cannot register for a room its been used by someone . And if any user tries to cancel the registration the room becomes vacant.

Medical Emergency to a Customer: This consists of the details of the emergency a customer has faced during his visit to the Hotel.

Customer Restaurant : This entity consists of the every customer who has visited any restaurant at any date.

Customer Room Service : This entity consists of various kinds of room service every customer has received.

Laundry : Consists of all the details of every laundry that has been done by a customer and which employee has handled that laundry.

Restaurant Type : This entity consists of all the different types of restaurants that are present in the hotel.

Bill : This consists of the total bill details of every customer , which has amount of every activity he has done in the hotel.

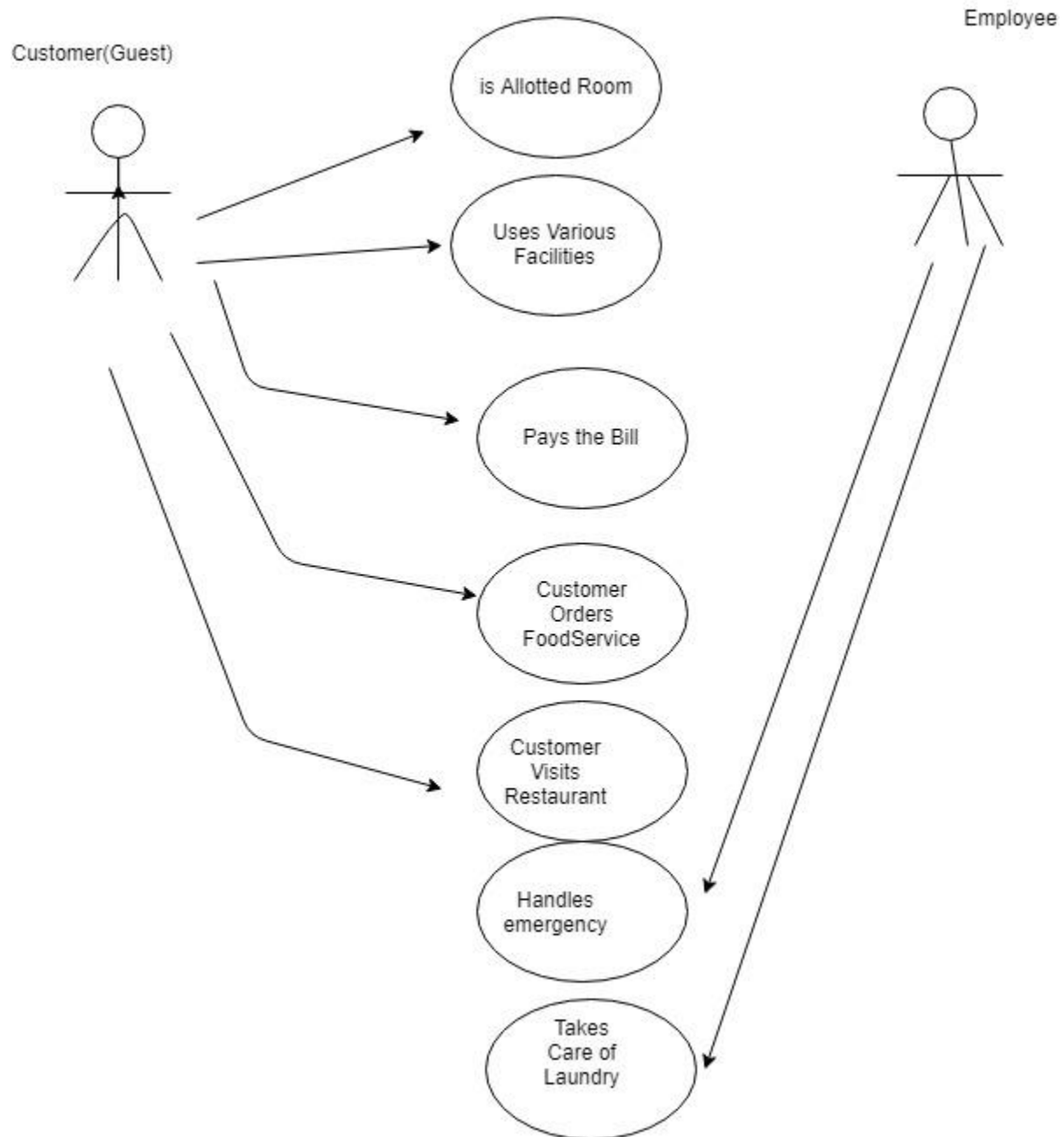
Customer Facility : This consists of every facility the customer has used.

Employee : Consists of details of every employee working in the hotel.

Department : Consists of all the departments under which employees work.

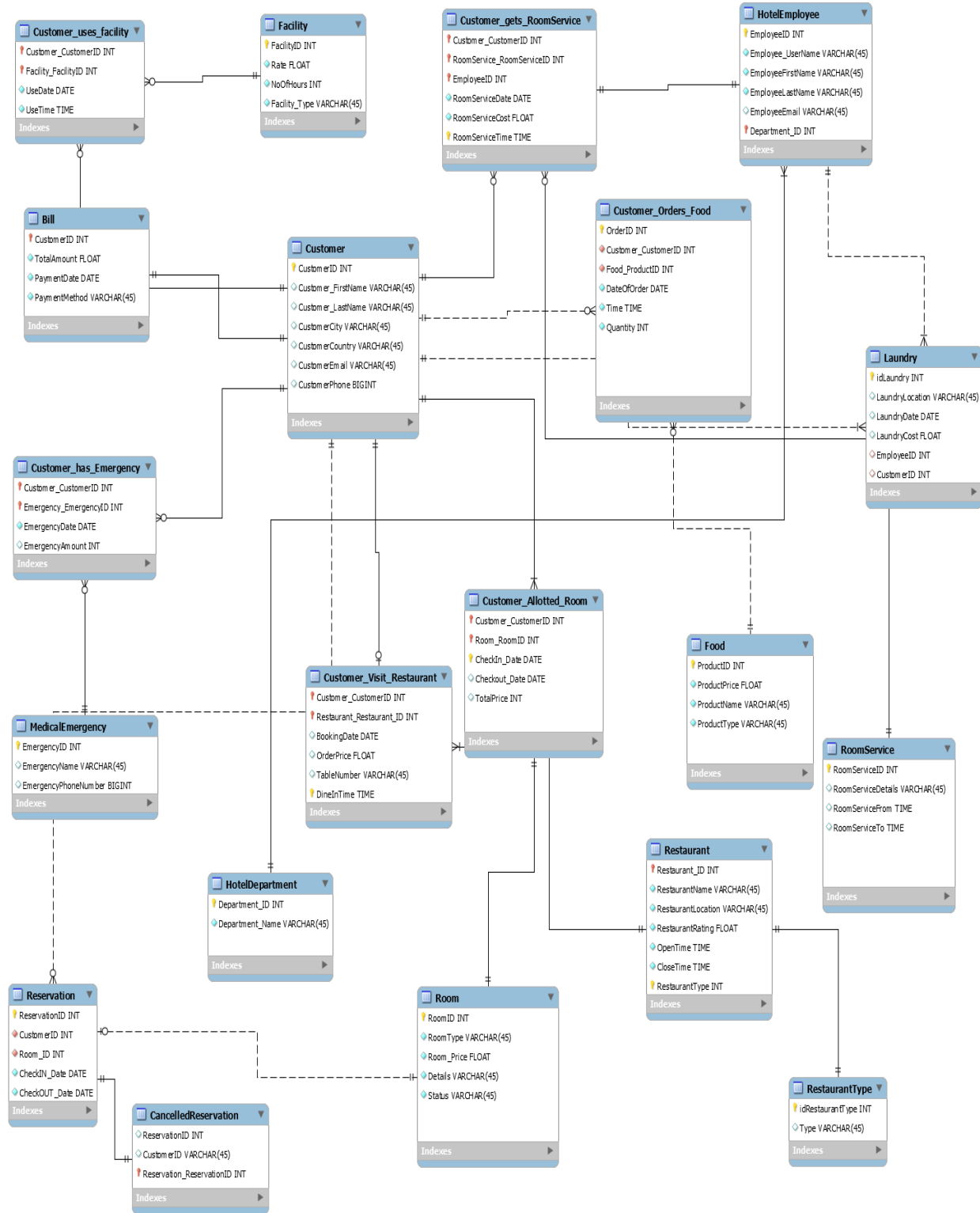
Customer Food : Consists of all the customers who had snacks .

Use Case Diagram :



The main uses cases in this project are a customer(Guest to the Hotel) and an Employee(Waiter,Manager etc) who take care of the efficient functioning .

ER Diagram :



Description between the various entities and relationship between them.

The customer table has the basic details about the customer (Guest) and has CustomerID as the Primary key.

Customer Allotted Room . A customer can have more than one or more rooms allotted to him so it is a 1:n Relationship between the two entities.CID,RoomID and CheckIN is the composite key.

The room tables consists of the details of a room , it has roomID(Room Number) as the primary key.Room to Customer Allotted room is a 1:1 relation as one room has one unique customer.

The restaurant table consists of the details of the restaurant with restaurantID and restaurantType being its primary key , A restaurant should belong to one of the cuisines so it is a 1:1 relation to the restaurant Type.

Reservation consists of ReservationID as the primary key ,with columns consisting of customer and room as the Foreign keys , A customer cannot reserve a room if its already booked.Any reservation that is cancelled will change the availability to Y. A room can or cannot have a reservation.

Customer has Emergency consists of customerID , EmergencyID as the concatenated key. This is as associative table that has these columns as foreign keys pointing to emergency and customer tables.(Many to Many Relationship).

A customer to Facility is also a many to many relationship as a customer can use many facilities and a facility can be used by many customers.

Many to many relation exists between a customer and food product as a food can be ordered by many customers and a customer can order many food products.

One to Many relation exists between an employee and a laundry as he can take care of many laundry .

Restaurant to customer is a one to many as a restaurant can have many customers.

Solutions to different questions can be obtained with the help of procedures, triggers, views.

Triggers

This trigger ensures that as soon as a customer is allotted a room, the room's status changes to not available and his bill is initiated, and based on the initial deposit either the customer should pay by card or cash.

```
13 DELIMITER //
14 begin
15     set @amount:= NEW.TotalPrice;
16     if @amount > 1000
17     then
18         Update room
19         set status = 'N'
20         where NEW.Room_RoomID = RoomID;
21     insert into bill values(NEW.Customer_CustomerID,NEW.TotalPrice,NEW.Checkout_Date,'Card Payment');
22     elseif @amount < 1000
23     then
24         Update room
25         set status = 'N'
26         where NEW.Room_RoomID = RoomID;
27     insert into bill values(NEW.Customer_CustomerID,NEW.TotalPrice,NEW.Checkout_Date,'Cash Payment');
28     end if;
29 END
```

4	2421.35	2017-12-13	Card Payment
5	2455	2017-12-13	Card Payment
6	2525	2017-12-13	Card Payment
7	1576	2017-12-13	Card Payment
8	1500	2017-12-12	Card Payment
9	1500	2017-12-06	Card Payment
10	500	2017-12-13	Cash Payment
NULL	NULL	NULL	NULL

63 x Apply Rev

201	Double Bedroom	1500	Sun View	N
...

The room's Status is changed to N (Not Available) as soon its Reserved.

Trigger 2 :

Before the Room is checked, It is checked whether the room is available, if it is it is Assigned else error is displayed

```

11
12 DELIMITER //
13 • create trigger on_reserved
14 after insert on reservation
15 for each row begin
16   set @roomID = new.Room_id;
17   set @status = (select status from room r inner join reservation rr on r.RoomID = @roomID);
18   if @status = 'N'
19   then
20     signal sqlstate '02000' set message_text = 'Sorry room already occupied!!';
21   else
22     update room
23     set room.Status = 'N'
24     where room.RoomID = NEW.Room_ID;
25   end if;
26 END
27 //
28 DELIMITER ;
29
30

```

141 23:14:08 insert into reservation values(15,11,207,2017-07-07,2018-09-09) Error Code: 1643. Sorry room already occupied!! 0.062 sec

Trigger 3

If a reservation is cancelled the room status gets updated back to available.

```

DELIMITER //
create trigger on_unreservedd
after delete on reservation
for each row begin
insert into cancelledreservation values(OLD.ReservationID,OLD.CustomerID);
update room
set room.Status = 'Y'
where room.RoomID = OLD.Room_ID;
END
//
DELIMITER ;
Delete from reservation where reservationID = 15

```

Result Grid				
Filter Rows:				
Edit: Export/Import: Wrap Cell Content:				
RoomID	RoomType	Room_Price	Details	Status
101	Single Bedroom	1200	Ocean View	N
102	Single Bedroom	1200	Ocean View	N
103	Single Bedroom	1200	Ocean View	N
104	Single Bedroom	1200	Ocean View	Y
105	Single Bedroom	1200	Ocean View	N
106	Single Bedroom	1200	Ocean View	N
107	Single Bedroom	1200	Ocean View	N
108	Single Bedroom	1200	Ocean View	N
109	Single Bedroom	1200	Ocean View	N
110	Single Bedroom	1200	Ocean View	N

View :

Customer who had reservations

```
create view customerReservation
as
select R.ReservationID,C.Customer_FirstName,C.Customer_Lastname ,RR.RoomID as RoomNumber,RR.RoomType as TypeOfRoomReserved
from Reservation R
inner join Customer C
on R.CustomerID = C.CustomerID
inner join Room RR
on R.room_id = RR.roomID;
```

ReservationID	Customer_FirstName	Customer_Lastname	RoomNumber	TypeOfRoomReserved
15	Pramod	Nagare	104	Single Bedroom

}

Procedure

It displays all the available rooms for a given date.

```
185 DELIMITER //
186 CREATE PROCEDURE get_available_rooms (IN room_type VARCHAR(50))
187 BEGIN
188 (
189 Select * from Room where room.RoomType = room_type
190 and room.Status = 'Y');
191 END//
192
193
194 call get_available_rooms('Single Bedroom');
195
```

RoomID	RoomType	Room_Price	Details	Status
111	Single Bedroom	1200	Ocean View	Y

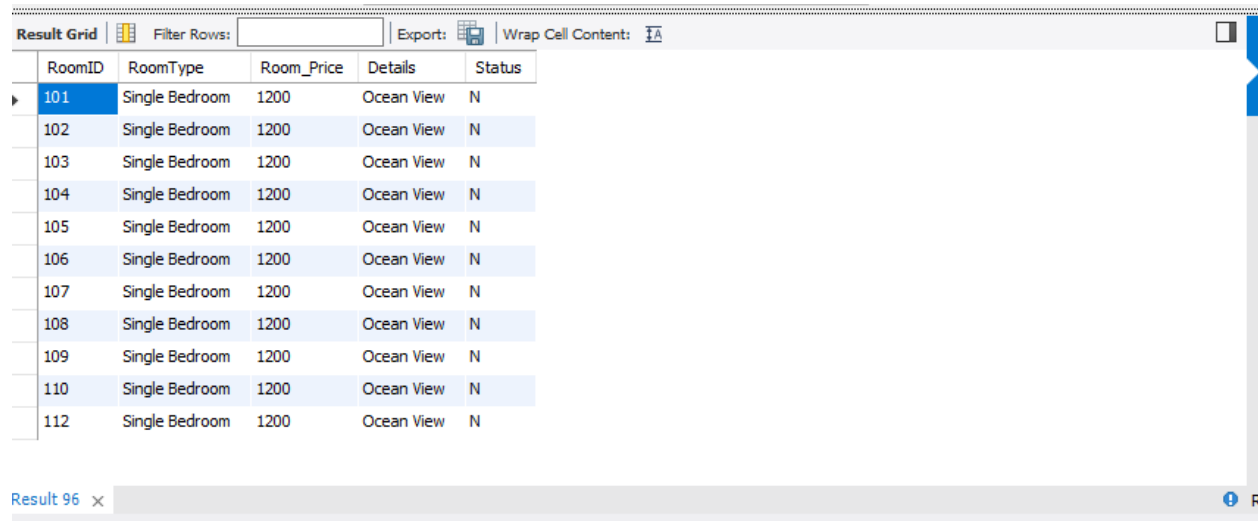
Result Grid
Form Editor
Field Tripas

Result 95 x Read Only

Procedure:

Get available rooms for a given room_type

```
184 DELIMITER //
185 • CREATE PROCEDURE get_booked_rooms (IN room_type VARCHAR(50))
186 BEGIN
187 (
188 select * from Room where room.RoomType = room_type
189 and room.Status = 'N'
190 );
191 );
192 END//
193
194 • call get_booked_rooms('Single Bedroom')
195
```

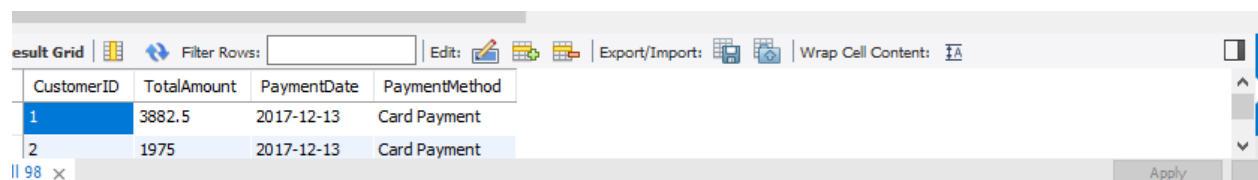


RoomID	RoomType	Room_Price	Details	Status
101	Single Bedroom	1200	Ocean View	N
102	Single Bedroom	1200	Ocean View	N
103	Single Bedroom	1200	Ocean View	N
104	Single Bedroom	1200	Ocean View	N
105	Single Bedroom	1200	Ocean View	N
106	Single Bedroom	1200	Ocean View	N
107	Single Bedroom	1200	Ocean View	N
108	Single Bedroom	1200	Ocean View	N
109	Single Bedroom	1200	Ocean View	N
110	Single Bedroom	1200	Ocean View	N
112	Single Bedroom	1200	Ocean View	N

TRIGGER:

Every time a customer orders a laundry the laundry amount is added to his bill.

```
184 DELIMITER //
185 • create trigger on_laundry_insertt
186 after insert on laundry
187 for each row begin
188 update bill
189 set bill.totalamount = bill.totalamount + NEW.laundryCost
190 where bill.customerID = NEW.customerID;
191 END
192 //
193 DELIMITER ;
194 •
```



CustomerID	TotalAmount	PaymentDate	PaymentMethod
1	3882.5	2017-12-13	Card Payment
2	1975	2017-12-13	Card Payment

Bill gets Updated.

PROCEDURE

```
119 • CREATE PROCEDURE totalLaundryFromSingleDate (IN date_given Date)
120 BEGIN
121 (
122 select sum(LaundryCost) from laundry
123 where laundryDate = date_given
124 );
125 end
126 //
127
128 call totalLaundryFromSingleDate('2017-12-12');
129
130 select * from bill;
131
132
133 DELIMITER //
134 • create trigger on visit customer
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

sum(LaundryCost)

465

Trigger to Update the Bill Amount of the Customer

```
84 DELIMITER //
85 • create trigger on_visit_customer
86 after insert on customer_visit_restaurant
87 for each row begin
88 update bill
89 set bill.totalamount = bill.totalamount + NEW.OrderPrice
90 where bill.customerID = NEW.Customer_CustomerID;
91 END
92 //
93 DELIMITER ;
94
```

View

-- list of all customers who had food in restaurants

```
184 • create view customerRestaurant
185 as
186 select Customer_FirstName,R.RestaurantName,rt.Type,CV.DineInTime,CV.BookingDate as DateOfVisit
187 from customer_visit_restaurant CV
188 inner join Customer C
189 on C.CustomerID = CV.Customer_CustomerID
190 inner join Restaurant R
191 on CV.Restaurant_Restaurant_ID = R.Restaurant_ID
192 inner join restauranttype rt
193 on R.RestaurantType = rt.idRestaurantType;
194
195 • select * from customerRestaurant
196
```

Customer_FirstName	RestaurantName	Type	DineInTime	DateOfVisit
Aashrith	Sangeetha	Chinese	15:00:00	2017-12-12
Dinesh	Shanus	Asian	15:00:00	2017-12-12
balu	Chutney	Thai	15:00:00	2017-12-12
sumanth	Bombay	Italian	15:00:00	2017-12-12

VIEW

```

183
184 • create view customerOrders
185 as
186 select C.OrderID,X.Customer_FirstName,X.Customer_LastName,f.productType,C.quantity
187 from customer_orders_food C
188 inner join food f
189 on c.food_productID = f.productID
190 inner join customer x
191 on c.customer_customerID = x.customerID;
192
193 • select * from customerOrders
194 •

```

OrderID	Customer_FirstName	Customer_LastName	productType	quantity
1	Aashrith	Chilkepalli	Samosa Chat	4
2	Anand	Ganesh	MIXED Chat	4
3	Dinesh	Maria	Samosa Chat	3
4	balu	ganesh	Dahi Chat	3
5	Dinesh	Maria	Cutlet Chat	4
6	karan	mailk	Sev Chat	4

customerOrders 104 x Read Only

Output

VIEW

```

95 • create view EmergencyView
96 as
97 select C.Customer_FirstName,C.Customer_LastName,E.EmergencyName,CE.EmergencyDate
98 from customer_has_emergency CE
99 inner join Customer C
100 on CE.Customer_CustomerID = C.CustomerID
101 inner join MedicalEmergency E
102 on CE.Emergency_EmergencyID = E.EmergencyID;
103

```

Customer_FirstName	Customer_LastName	EmergencyName	EmergencyDate
Aashrith	Chilkepalli	Fever	2017-08-08
Anand	Ganesh	Heart Pain	2017-08-18
Dinesh	Maria	Chest Pain	2017-08-28
balu	ganesh	Heart Pain	2017-08-08

PROCEDURE

```

194
195 DELIMITER //
196 CREATE PROCEDURE get_customers(IN today_date date)
197 BEGIN
198     SELECT x.Customer_CustomerID,x.Room_RoomID,c.customer_firstName
199     FROM customer_allotted_room x INNER JOIN Customer c
200     on x.customer_customerID = C.CustomerID
201     WHERE
202     x.checkout_date >= today_date AND x.checkin_date <= today_date ;
203 END
204 //
205 DELIMITER ;
206

```

Customer_CustomerID	Room_RoomID	customer_firstName
1	101	Aashrith
2	103	Anand
3	106	Dinesh
4	108	balu
5	110	sumanth
6	203	karan
7	205	varun
10	201	Tom

result 107 x

VIEW

All the registered customers who don't have a room allotted yet.

```

94
95 create view AllCustomers
96 as
97 select * from Customer C
98 where C.CustomerID NOT in (select customer_allotted_room.Customer_CustomerID from customer_allotted_room);
99 select * from allcustomers
100

```

8	sumanth krishna	singh	varanasi	Dubai	sumanth@gmail.com	23232322
11	Pramod	Nagare	Delhi	Taiwan	pramod@gmail.com	12121212

customers 108 x

TRIGGER

```

DELIMITER //
:create trigger customer_uses_Facilityy
after insert on customer_uses_facility
for each row
begin
set @ID := NEW.facility_facilityID;
set @price := (select rate from facility where FacilityID = @ID);
update bill
set bill.totalamount = bill.totalamount + @price
where bill.customerID = (NEW.Customer_CustomerID);
END
//
DELIMITER ;

select * from facility;

```

VIEW

```

193 • create view numberofFacByCustomer
194 • as
195 • select C.customer_customerID,CC.Customer_FirstName,f.facility_type ,count(*) as ActivityCount
196 • from customer_uses_facility C
197 • inner join Customer CC
198 • on C.customer_customerID = cc.CustomerID
199 • inner join facility f
200 • on C.Facility_FacilityID = f.facilityID
201 • group by C.customer_customerID,f.facility_type;
202 •
203 • select * from numberofFacByCustomer;
204 •

```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

customer_customerID	Customer_FirstName	facility_type	ActivityCount
1	Aashrith	Private Lawn	1
1	Aashrith	Spa	1
5	sumanth	Deluxe Swimming	1
6	karan	WIFI	1
7	varun	Conference Room 2	1
7	varun	Private Lawn	1

umberofFacByCustomer 112 x

PROCEDURE

```
193 Delimiter //
194 • create procedure moneyFromFacility(IN today_date date)
195 begin
196   select sum(rate) as totalMoneyFromFacility from facility f
197   inner join customer_uses_facility c
198   on c.Facility_FacilityID = f.FacilityID
199   where c.UseDate = today_date;
200 end
201 //
202
203 • call moneyFromFacility('2017-12-12');
204 •
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

totalMoneyFromFacility
226

Result 114 x

Read

PROCEDURE

List of all customers who had room service on a particular day.

```
207 Delimiter //
208 • create procedure RoomServiceDetails(IN today_date date)
209 begin
210   select C.customer_FirstName,C.customer_lastName,E.EmployeeFirstName as 'Employee Handled',CR.RoomServiceCost as To
211   from customer_gets_roomservice CR
212   inner join Customer C
213   on C.CustomerID = CR.Customer_CustomerID
214   inner join hotelemmployee E
215   on CR.EmployeeID = E.EmployeeID
216   inner join roomservice RS
217   on CR.RoomService_RoomServiceID = RS.RoomServiceID
218   where CR.RoomServiceDate = today_date;
219 end
220
```

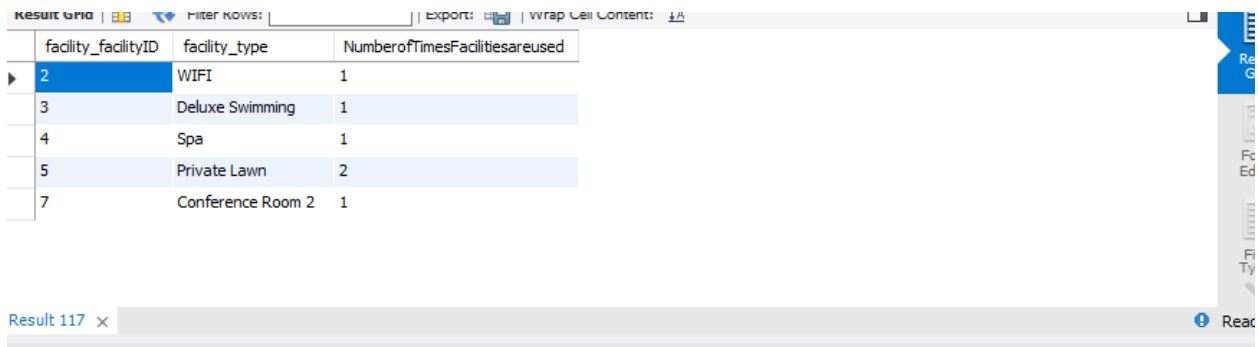
customer_FirstName	customer_LastName	Employee Handled	RoomServiceDate	RoomServiceDate	RoomServiceDetails
Aashrith	Chilkepalli	Rahul	2017-12-12	2017-12-12	Dusting
balu	ganesh	Rahul	2017-12-12	2017-12-12	Dusting
sumanth krishna	singh	Rahul	2017-12-12	2017-12-12	Dusting

Result 115 x

Read Only

No of facilities used by customers

```
Select f.facility_facilityID,ff.facility_type,count(*) as NumberofTimesFacilitiesareused
from customer_uses_facility f
inner join facility ff
on f.facility_facilityID = ff.facilityID
group by f.facility_facilityID
```



The screenshot shows a database query result in a software interface. The interface includes a toolbar with options like 'Filter Rows', 'Export', and 'Wrap Cell Content'. The result is displayed in a table with three columns: 'facility_facilityID', 'facility_type', and 'NumberofTimesFacilitiesareused'. The table contains six rows of data. A status bar at the bottom indicates 'Result 117' and 'Reac'.

facility_facilityID	facility_type	NumberofTimesFacilitiesareused
2	WIFI	1
3	Deluxe Swimming	1
4	Spa	1
5	Private Lawn	2
7	Conference Room 2	1

Giving privileges to a User :

```
422 create user 'hr'@'localhost' identified by 'hr12'
423 grant select,insert,update on mydb.hotelEmployee to 'hr'
424
```

```
mysql> use mydb;
Database changed
mysql> insert into hotelEmployee values(1,'pramod','asas','asa','asa','4');
Query OK, 1 row affected (0.17 sec)

mysql> insert into facility values(1,12,12,2);
ERROR 1142 (42000): INSERT command denied to user 'hr'@'localhost' for table 'facility'
mysql>
```

If the hotel manager HR tries to access any other table in the db , his access is denied.

Transaction:

```
start transaction;
update bill set billStatus = 'Processing'
where
customerID = 1

DELIMITER //
create trigger onDelete
after delete on bill
for each row
begin
insert into completedBill values(Old.CustomerID,Old.totalAmount,'Completed');
end
//

delete from bill where customerID = 1;
commit
```

If the transaction is not successful it will be rolledback automatically

Result Grid			Filter Rows:	Export:	Wrap Cell Content:
CustomerID	billAmount	Status			
1	3882	Completed			

completedBill 126 x

Total sum of each customer using rollup in laundry Service

```
select l.customerID,sum(l.LaundryCost) as totalcost
from laundry l
group by customerID
with rollup;
```

Result Grid			Filter Rows:	Export:	Wrap Cell Content:
customerID	totalcost				
1	650				
2	75				
3	55				
4	65				
5	55				
6	65				
NULL	965				

Result 134 x

MYSQLDUMP

```
CREATE DATABASE IF NOT EXISTS `mydb` /*!40100 DEFAULT CHARACTER SET  
utf8 */;
```

```
USE `mydb`;
```

```
-- MySQL dump 10.13 Distrib 5.6.23, for Win64 (x86_64)
```

```
--
```

```
-- Host: 127.0.0.1 Database: mydb
```

```
-- -----
```

```
-- Server version 5.6.37
```

```
/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
```

```
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS  
*/;
```

```
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION  
*/;
```

```
/*!40101 SET NAMES utf8 */;
```

```
/*!40103 SET @OLD_TIME_ZONE=@@TIME_ZONE */;
```

```
/*!40103 SET TIME_ZONE='+00:00' */;
```

```
/*!40014 SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS,  
UNIQUE_CHECKS=0 */;
```

```
/*!40014 SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS,  
FOREIGN_KEY_CHECKS=0 */;
```

```
/*!40101 SET @OLD_SQL_MODE=@@SQL_MODE,  
SQL_MODE='NO_AUTO_VALUE_ON_ZERO' */;
```

```
/*!40111 SET @OLD_SQL_NOTES=@@SQL_NOTES, SQL_NOTES=0 */;
```

```
--
```

```
-- Dumping data for table `bill`
```

```
--
```

```
LOCK TABLES `bill` WRITE;
```

```
/*!40000 ALTER TABLE `bill` DISABLE KEYS */;
```

```
INSERT INTO `bill` VALUES (2,1975,'2017-12-13','Card Payment','P'),(3,3035,'2017-12-13','Card Payment','P'),(4,2421.35,'2017-12-13','Card Payment','P'),(5,2455,'2017-12-13','Card Payment','P'),(6,2525,'2017-12-13','Card Payment','P'),(7,1653,'2017-12-13','Card Payment','P'),(8,1500,'2017-12-12','Card Payment','P'),(9,1500,'2017-12-06','Card Payment','P'),(10,500,'2017-12-13','Cash Payment','P');
```

```
/*!40000 ALTER TABLE `bill` ENABLE KEYS */;
```

```
UNLOCK TABLES;
```

```
/*!50003 SET @saved_cs_client = @@character_set_client */;
```

```
/*!50003 SET @saved_cs_results = @@character_set_results */;
```

```
/*!50003 SET @saved_col_connection = @@collation_connection */;
```

```
/*!50003 SET character_set_client = utf8 */;
```

```
/*!50003 SET character_set_results = utf8 */;
```

```
/*!50003 SET collation_connection = utf8_general_ci */;
```

```
/*!50003 SET @saved_sql_mode = @@sql_mode */;
```

```

/*!50003 SET sql_mode          = 'NO_ENGINE_SUBSTITUTION' */ ;

DELIMITER ;;

/*!50003 CREATE*/ /*!50017 DEFINER=`root`@`localhost`*/ /*!50003 trigger
onDelete

after delete on bill

for each row

begin

insert into completedBill values(Old.CustomerID,Old.totalAmount,'Completed');

end */;;

DELIMITER ;

/*!50003 SET sql_mode          = @saved_sql_mode */ ;

/*!50003 SET character_set_client = @saved_cs_client */ ;

/*!50003 SET character_set_results = @saved_cs_results */ ;

/*!50003 SET collation_connection = @saved_col_connection */ ;

--

-- Dumping data for table `cancelledreservation`

--

LOCK TABLES `cancelledreservation` WRITE;

/*!40000 ALTER TABLE `cancelledreservation` DISABLE KEYS */;

INSERT INTO `cancelledreservation` VALUES (1,'1'),(5,'1'),(10,'8'),(15,'11');

```

```
/*!40000 ALTER TABLE `cancelledreservation` ENABLE KEYS */;
```

```
UNLOCK TABLES;
```

```
--
```

```
-- Dumping data for table `completedbill`
```

```
--
```

```
LOCK TABLES `completedbill` WRITE;
```

```
/*!40000 ALTER TABLE `completedbill` DISABLE KEYS */;
```

```
INSERT INTO `completedbill` VALUES (1,3882,'Completed');
```

```
/*!40000 ALTER TABLE `completedbill` ENABLE KEYS */;
```

```
UNLOCK TABLES;
```

```
--
```

```
-- Dumping data for table `customer`
```

```
--
```

```
LOCK TABLES `customer` WRITE;
```

```
/*!40000 ALTER TABLE `customer` DISABLE KEYS */;
```

```
INSERT INTO `customer` VALUES
```

```
(0,NULL,'',NULL,NULL,NULL,NULL),(1,'Aashrith','Chilkepalli','Chennai','India','aashri  
th95@gmail.com',9196496534),(2,'Anand','Ganesh','Bangalore','India','anandgane
```

```
sh@gmail.com',23232232323232),(3,'Dinesh','Maria','Vizag','India','dinesh@gmail.com',23232323),(4,'balu','ganesh','rajamundry','India','baluganesh@gmail.com',12323232),(5,'sumanth','Tippa','kodai','Qatar','SumanthTippa@gmail.com',2323223),(6,'karan','mailk','delhi','dubai','karan@gmail.com',91127121),(7,'varun','verma','bangalore','India','varun@gmail.com',121212),(8,'sumanth krishna','singh','varanasi','Dubai','sumanth@gmail.com',23232322),(9,'Bob','prasad','pune','South Korea','bob@gmail.com',12121212),(10,'Tom','reddy','Delhi','North Korea','tom@gmail.com',1212323),(11,'Pramod','Nagare','Delhi','Taiwan','pramod@gmail.com',12121212);
```

```
/*!40000 ALTER TABLE `customer` ENABLE KEYS */;
```

```
UNLOCK TABLES;
```

```
--
```

```
-- Dumping data for table `customer_allotted_room`
```

```
--
```

```
LOCK TABLES `customer_allotted_room` WRITE;
```

```
/*!40000 ALTER TABLE `customer_allotted_room` DISABLE KEYS */;
```

```
INSERT INTO `customer_allotted_room` VALUES (1,101,'2017-12-12','2017-12-13',1200),(2,103,'2017-12-12','2017-12-13',1200),(3,106,'2017-12-12','2017-12-13',1200),(4,108,'2017-12-12','2017-12-13',1200),(5,110,'2017-12-12','2017-12-13',1200),(6,203,'2017-12-12','2017-12-13',1500),(7,205,'2017-12-12','2017-12-13',1500),(9,206,'2017-12-05','2017-12-06',1500),(10,201,'2017-12-12','2017-12-13',500);
```

```
/*!40000 ALTER TABLE `customer_allotted_room` ENABLE KEYS */;
```

```
UNLOCK TABLES;
```

```

/*!50003 SET @saved_cs_client      = @@character_set_client */;

/*!50003 SET @saved_cs_results     = @@character_set_results */;

/*!50003 SET @saved_col_connection = @@collation_connection */;

/*!50003 SET character_set_client  = utf8 */;

/*!50003 SET character_set_results = utf8 */;

/*!50003 SET collation_connection  = utf8_general_ci */;

/*!50003 SET @saved_sql_mode       = @@sql_mode */;

/*!50003 SET sql_mode              = 'NO_ENGINE_SUBSTITUTION' */;

DELIMITER ;;

/*!50003 CREATE*/ /*!50017 DEFINER=`root`@`localhost`*/ /*!50003 trigger
on_customer_allotted

after insert on customer_allotted_room

for each row begin

set @amount:= NEW.TotalPrice;

if @amount > 1000

then

Update room

set status = 'N'

where NEW.Room_RoomID = RoomID;

insert into bill
values(NEW.Customer_CustomerID,NEW.TotalPrice,NEW.Checkout_Date,'Card
Payment');

```

```

elseif @amount < 1000

then

Update room

set status = 'N'

where NEW.Room_RoomID = RoomID;

insert into bill
values(NEW.Customer_CustomerID,NEW.TotalPrice,NEW.Checkout_Date,'Cash
Payment');

end if;

END *//;;

DELIMITER ;

/*!50003 SET sql_mode          = @saved_sql_mode */ ;

/*!50003 SET character_set_client = @saved_cs_client */ ;

/*!50003 SET character_set_results = @saved_cs_results */ ;

/*!50003 SET collation_connection = @saved_col_connection */ ;

--

-- Dumping data for table `customer_gets_roomservice`

--

LOCK TABLES `customer_gets_roomservice` WRITE;

/*!40000 ALTER TABLE `customer_gets_roomservice` DISABLE KEYS */;

```

```

INSERT INTO `customer_gets_roomservice` VALUES (1,3,5,'2017-12-12',76.5,'08:00:00'),(4,3,5,'2017-12-12',76.35,'08:00:00'),(8,3,5,'2017-12-12',76.5,'08:00:00');

/*!40000 ALTER TABLE `customer_gets_roomservice` ENABLE KEYS */;

UNLOCK TABLES;

/*!50003 SET @saved_cs_client      = @@character_set_client */;

/*!50003 SET @saved_cs_results    = @@character_set_results */;

/*!50003 SET @saved_col_connection = @@collation_connection */;

/*!50003 SET character_set_client = utf8 */;

/*!50003 SET character_set_results = utf8 */;

/*!50003 SET collation_connection = utf8_general_ci */;

/*!50003 SET @saved_sql_mode      = @@sql_mode */;

/*!50003 SET sql_mode            = 'NO_ENGINE_SUBSTITUTION' */;

DELIMITER ;;

/*!50003 CREATE*/ /*!50017 DEFINER=`root`@`localhost`*/ /*!50003 trigger
customer_gets_RS

after insert on customer_gets_roomservice

for each row

begin

update bill

set bill.totalamount = bill.totalamount + NEW.RoomServiceCost

where bill.customerID = (NEW.Customer_CustomerID);

```



```
END *//;
```

```
DELIMITER ;
```

```
/*!50003 SET sql_mode          = @saved_sql_mode */ ;
```

```
/*!50003 SET character_set_client = @saved_cs_client */ ;
```

```
/*!50003 SET character_set_results = @saved_cs_results */ ;
```

```
/*!50003 SET collation_connection = @saved_col_connection */ ;
```

```
--
```

```
-- Dumping data for table `customer_has_emergency`
```

```
--
```

```
LOCK TABLES `customer_has_emergency` WRITE;
```

```
/*!40000 ALTER TABLE `customer_has_emergency` DISABLE KEYS */;
```

```
INSERT INTO `customer_has_emergency` VALUES (1,1,'2017-08-08',500),(2,2,'2017-08-18',2500),(3,3,'2017-08-28',1500),(4,2,'2017-08-08',2500);
```

```
/*!40000 ALTER TABLE `customer_has_emergency` ENABLE KEYS */;
```

```
UNLOCK TABLES;
```

```
--
```

```
-- Dumping data for table `customer_orders_food`
```

```
--
```

```

LOCK TABLES `customer_orders_food` WRITE;

/*!40000 ALTER TABLE `customer_orders_food` DISABLE KEYS */;

INSERT INTO `customer_orders_food` VALUES (1,1,1,'2017-12-12','20:00:00',4),(2,2,2,'2017-12-12','21:00:00',4),(3,3,1,'2017-12-12','22:00:00',3),(4,4,4,'2017-12-12','20:00:00',3),(5,3,5,'2017-12-12','23:00:00',4),(6,6,6,'2017-12-12','20:00:00',4);

/*!40000 ALTER TABLE `customer_orders_food` ENABLE KEYS */;

UNLOCK TABLES;

/*!50003 SET @saved_cs_client      = @@character_set_client */;

/*!50003 SET @saved_cs_results    = @@character_set_results */;

/*!50003 SET @saved_col_connection = @@collation_connection */;

/*!50003 SET character_set_client = utf8 */;

/*!50003 SET character_set_results = utf8 */;

/*!50003 SET collation_connection = utf8_general_ci */;

/*!50003 SET @saved_sql_mode      = @@sql_mode */;

/*!50003 SET sql_mode            = 'NO_ENGINE_SUBSTITUTION' */;

DELIMITER ;;

/*!50003 CREATE*/ /*!50017 DEFINER=`root`@`localhost`*/ /*!50003 trigger
Customer_orders_food

after insert on customer_orders_food

for each row begin

set @ID := NEW.Food_ProductID;

set @price := (select productPrice from food where productID = @ID);

```

update bill

set bill.totalamount = bill.totalamount + NEW.Quantity*@price

where bill.customerID = (NEW.Customer_CustomerID);

END */;;

DELIMITER ;

/*!50003 SET sql_mode = @saved_sql_mode */ ;

/*!50003 SET character_set_client = @saved_cs_client */ ;

/*!50003 SET character_set_results = @saved_cs_results */ ;

/*!50003 SET collation_connection = @saved_col_connection */ ;

--

-- Dumping data for table `customer_uses_facility`

--

LOCK TABLES `customer_uses_facility` WRITE;

/*!40000 ALTER TABLE `customer_uses_facility` DISABLE KEYS */;

INSERT INTO `customer_uses_facility` VALUES (1,4,'2017-12-12','17:35:35'),(1,5,'2017-12-12','00:20:17'),(5,3,'2017-12-12','19:35:35'),(6,2,'2017-12-11','15:35:35'),(7,5,'2017-12-11','13:35:35'),(7,7,'2017-12-11','13:35:35');

/*!40000 ALTER TABLE `customer_uses_facility` ENABLE KEYS */;

UNLOCK TABLES;

```

/*!50003 SET @saved_cs_client      = @@character_set_client */;

/*!50003 SET @saved_cs_results     = @@character_set_results */;

/*!50003 SET @saved_col_connection = @@collation_connection */;

/*!50003 SET character_set_client  = utf8 */;

/*!50003 SET character_set_results = utf8 */;

/*!50003 SET collation_connection  = utf8_general_ci */;

/*!50003 SET @saved_sql_mode       = @@sql_mode */;

/*!50003 SET sql_mode              = 'NO_ENGINE_SUBSTITUTION' */;

DELIMITER ;;

/*!50003 CREATE*/ /*!50017 DEFINER=`root`@`localhost`*/ /*!50003 trigger
customer_uses_Facilityyy
after insert on customer_uses_facility
for each row
begin
set @ID := NEW.facility_facilityID;
set @price := (select rate from facility where FacilityID = @ID);
update bill
set bill.totalamount = bill.totalamount + @price
where bill.customerID = (NEW.Customer_CustomerID);
END */;;

DELIMITER ;

/*!50003 SET sql_mode              = @saved_sql_mode */;

```

```
/*!50003 SET character_set_client = @saved_cs_client */;

/*!50003 SET character_set_results = @saved_cs_results */;

/*!50003 SET collation_connection = @saved_col_connection */;

--

-- Dumping data for table `customer_visit_restaurant`

--

LOCK TABLES `customer_visit_restaurant` WRITE;

/*!40000 ALTER TABLE `customer_visit_restaurant` DISABLE KEYS */;

INSERT INTO `customer_visit_restaurant` VALUES (1,4,'2017-12-12',700,'5','15:00:00'),(3,2,'2017-12-12',530,'25','15:00:00'),(4,5,'2017-12-12',540,'52','15:00:00'),(5,3,'2017-12-12',550,'15','15:00:00');

/*!40000 ALTER TABLE `customer_visit_restaurant` ENABLE KEYS */;

UNLOCK TABLES;

/*!50003 SET @saved_cs_client = @@character_set_client */;

/*!50003 SET @saved_cs_results = @@character_set_results */;

/*!50003 SET @saved_col_connection = @@collation_connection */;

/*!50003 SET character_set_client = utf8 */;

/*!50003 SET character_set_results = utf8 */;

/*!50003 SET collation_connection = utf8_general_ci */;

/*!50003 SET @saved_sql_mode = @@sql_mode */;
```

```
/*!50003 SET sql_mode          = 'NO_ENGINE_SUBSTITUTION' */ ;

DELIMITER ;;

/*!50003 CREATE*/ /*!50017 DEFINER=`root`@`localhost`*/ /*!50003 trigger
on_visit_customer

after insert on customer_visit_restaurant

for each row begin

update bill

set bill.totalamount = bill.totalamount + NEW.OrderPrice

where bill.customerID = NEW.Customer_CustomerID;

END */;;

DELIMITER ;

/*!50003 SET sql_mode          = @saved_sql_mode */ ;

/*!50003 SET character_set_client = @saved_cs_client */ ;

/*!50003 SET character_set_results = @saved_cs_results */ ;

/*!50003 SET collation_connection = @saved_col_connection */ ;

--

-- Dumping data for table `facility`

--

LOCK TABLES `facility` WRITE;

/*!40000 ALTER TABLE `facility` DISABLE KEYS */;
```

```
INSERT INTO `facility` VALUES
(1,50,24,'Swimming'),(2,60,20,'WIFI'),(3,70,22,'Deluxe
Swimming'),(4,80,23,'Spa'),(5,76,20,'Private Lawn'),(6,56,17,'Conference
Room'),(7,77,18,'Conference Room 2'),(8,45,15,'Conference Room
3'),(9,77,12,'Driver'),(10,66,24,'Unlimited WIFI');
```

```
/*!40000 ALTER TABLE `facility` ENABLE KEYS */;
```

```
UNLOCK TABLES;
```

```
--
```

```
-- Dumping data for table `food`
```

```
--
```

```
LOCK TABLES `food` WRITE;
```

```
/*!40000 ALTER TABLE `food` DISABLE KEYS */;
```

```
INSERT INTO `food` VALUES (1,150,'Appetizer','Samosa
Chat'),(2,175,'Appetizer','MIXED Chat'),(3,125,'Appetizer','Bhel
Chat'),(4,180,'Appetizer','Dahi Chat'),(5,200,'Appetizer','Cutlet
Chat'),(6,225,'Appetizer','Sev Chat'),(7,245,'Appetizer','Pav Chat'),(8,245,'Main
Course','Biryani'),(9,145,'Main Course','Chicken Biryani'),(10,175,'Main
Course','MuttonBiryani'),(11,545,'Main Course','prawn Biryani'),(12,645,'Main
Course','Biryani'),(13,85,'Main Course','Naan'),(14,245,'Main
Course','Chapathi'),(15,115,'Main Course','Noodles'),(16,115,'Main Course','Egg
Noodles'),(17,105,'Main Course','Kurma'),(18,95,'Main Course','Fried
Rice'),(19,225,'Main Course','Egg FriedRice'),(20,235,'Main Course','Manchuriya');
```

```
/*!40000 ALTER TABLE `food` ENABLE KEYS */;
```

```
UNLOCK TABLES;
```

--

-- Dumping data for table `hasemergency`

--

LOCK TABLES `hasemergency` WRITE;

/*!40000 ALTER TABLE `hasemergency` DISABLE KEYS */;

/*!40000 ALTER TABLE `hasemergency` ENABLE KEYS */;

UNLOCK TABLES;

--

-- Dumping data for table `hoteldepartment`

--

LOCK TABLES `hoteldepartment` WRITE;

/*!40000 ALTER TABLE `hoteldepartment` DISABLE KEYS */;

INSERT INTO `hoteldepartment` VALUES

(1,'Cleaning'),(2,'Manager'),(3,'Reception'),(4,'Laundry'),(5,'Restaurant
Manager'),(6,'Waiter'),(7,'Pool Buy'),(8,'Waiter'),(9,'Receptionist'),(10,'Washer');

/*!40000 ALTER TABLE `hoteldepartment` ENABLE KEYS */;

UNLOCK TABLES;

--

-- Dumping data for table `hotelemmployee`

--

LOCK TABLES `hotelemmployee` WRITE;

/*!40000 ALTER TABLE `hotelemmployee` DISABLE KEYS */;

INSERT INTO `hotelemmployee` VALUES

(1,'Sameer123','Sameer','goel','sameer@gmail.com',1),(1,'pramod','asas','asa','asa',4),(2,'Varun123','varun','varma','varunvarma@gmail.com',2),(3,'Sathish','Sathish','prasad','sathish@gmail.com',1),(4,'Aravind','Mandapaka','Aravind','maravind@gmail.com',4),(5,'Rahul123','Rahul','Kamath','rahul@gmail.com',5),(6,'magesh123','magesh','sharma','magesh@gmail.com',6),(7,'deepan123','deepan','varma','deepan@gmail.com',7),(8,'sugan123','sugan','burma','sugan@gmail.com',8),(9,'pramod124','pramod','nagare','pramod@gmail.com',9),(10,'hrithik145','hrithik','roshan','hrithik@gmail.com',4),(11,'salman156','salman','salmankhan','salman@gmail.com',8);

/*!40000 ALTER TABLE `hotelemmployee` ENABLE KEYS */;

UNLOCK TABLES;

--

-- Dumping data for table `laundry`

--

LOCK TABLES `laundry` WRITE;

/*!40000 ALTER TABLE `laundry` DISABLE KEYS */;

```
INSERT INTO `laundry` VALUES (1,'4th Floor','2017-12-12',75,4,1),(2,'4th Floor','2017-12-12',75,4,2),(3,'3th Floor','2017-12-12',55,4,3),(4,'4th Floor','2017-12-12',65,4,4),(5,'5th Floor','2017-12-12',55,4,5),(6,'4th Floor','2017-12-12',65,4,6),(7,'1st Floor','2017-12-12',75,4,1),(9,'Chennai','2017-12-11',500,8,1);
```

```
/*!40000 ALTER TABLE `laundry` ENABLE KEYS */;
```

```
UNLOCK TABLES;
```

```
/*!50003 SET @saved_cs_client = @@character_set_client */;
```

```
/*!50003 SET @saved_cs_results = @@character_set_results */;
```

```
/*!50003 SET @saved_col_connection = @@collation_connection */;
```

```
/*!50003 SET character_set_client = utf8 */;
```

```
/*!50003 SET character_set_results = utf8 */;
```

```
/*!50003 SET collation_connection = utf8_general_ci */;
```

```
/*!50003 SET @saved_sql_mode = @@sql_mode */;
```

```
/*!50003 SET sql_mode = 'NO_ENGINE_SUBSTITUTION' */;
```

```
DELIMITER ;;
```

```
/*!50003 CREATE*/ /*!50017 DEFINER=`root`@`localhost`*/ /*!50003 trigger  
on_laundry_insertt
```

```
after insert on laundry
```

```
for each row begin
```

```
update bill
```

```
set bill.totalamount = bill.totalamount + NEW.laundryCost
```

```
where bill.customerID = NEW.customerID;
```

```
END */;;
```

DELIMITER ;

/*!50003 SET sql_mode = @saved_sql_mode */ ;

/*!50003 SET character_set_client = @saved_cs_client */ ;

/*!50003 SET character_set_results = @saved_cs_results */ ;

/*!50003 SET collation_connection = @saved_col_connection */ ;

--

-- Dumping data for table `medicalemergency`

--

LOCK TABLES `medicalemergency` WRITE;

/*!40000 ALTER TABLE `medicalemergency` DISABLE KEYS */;

INSERT INTO `medicalemergency` VALUES (1,'Fever',100),(2,'Heart Pain',101),(3,'Chest Pain',102),(4,'Breathing Issues',103),(5,'Stomach Pain',104),(6,'Vision Issues',105),(7,'Ear issues',106),(8,'ASthma',107),(9,'Heart Attack',108),(10,'Cold',109);

/*!40000 ALTER TABLE `medicalemergency` ENABLE KEYS */;

UNLOCK TABLES;

--

-- Dumping data for table `reservation`

--

```

LOCK TABLES `reservation` WRITE;

/*!40000 ALTER TABLE `reservation` DISABLE KEYS */;

INSERT INTO `reservation` VALUES (15,11,104,'2017-07-07','2018-09-09');

/*!40000 ALTER TABLE `reservation` ENABLE KEYS */;

UNLOCK TABLES;

/*!50003 SET @saved_cs_client      = @@character_set_client */;

/*!50003 SET @saved_cs_results     = @@character_set_results */;

/*!50003 SET @saved_col_connection = @@collation_connection */;

/*!50003 SET character_set_client  = utf8 */;

/*!50003 SET character_set_results = utf8 */;

/*!50003 SET collation_connection = utf8_general_ci */;

/*!50003 SET @saved_sql_mode       = @@sql_mode */;

/*!50003 SET sql_mode              = 'NO_ENGINE_SUBSTITUTION' */;

DELIMITER ;;

/*!50003 CREATE*/ /*!50017 DEFINER=`root`@`localhost`*/ /*!50003 trigger
on_reserved

after insert on reservation

for each row begin

set @roomID = new.Room_id;

set @status = (select status from room r inner join reservation rr on r.RoomID =
@roomID);

```

```

if @status = 'N'

then

signal sqlstate '02000' set message_text = 'Sorry room already occupied!!';

else

update room

set room.Status = 'N'

where room.RoomID = NEW.Room_ID;

end if;

END *//;;

DELIMITER ;

/*!50003 SET sql_mode          = @saved_sql_mode */ ;

/*!50003 SET character_set_client = @saved_cs_client */ ;

/*!50003 SET character_set_results = @saved_cs_results */ ;

/*!50003 SET collation_connection = @saved_col_connection */ ;

/*!50003 SET @saved_cs_client     = @@character_set_client */ ;

/*!50003 SET @saved_cs_results   = @@character_set_results */ ;

/*!50003 SET @saved_col_connection = @@collation_connection */ ;

/*!50003 SET character_set_client = utf8 */ ;

/*!50003 SET character_set_results = utf8 */ ;

/*!50003 SET collation_connection = utf8_general_ci */ ;

/*!50003 SET @saved_sql_mode     = @@sql_mode */ ;

```

```

/*!50003 SET sql_mode          = 'NO_ENGINE_SUBSTITUTION' */ ;

DELIMITER ;;

/*!50003 CREATE*/ /*!50017 DEFINER=`root`@`localhost`*/ /*!50003 trigger
on_unreservedd

after delete on reservation

for each row begin

insert into cancelledreservation values(OLD.ReservationID,OLD.CustomerID);

update room

set room.Status = 'Y'

where room.RoomID = OLD.Room_ID;

END */;;

DELIMITER ;

/*!50003 SET sql_mode          = @saved_sql_mode */ ;

/*!50003 SET character_set_client = @saved_cs_client */ ;

/*!50003 SET character_set_results = @saved_cs_results */ ;

/*!50003 SET collation_connection = @saved_col_connection */ ;

--

-- Dumping data for table `restaurant`

--

LOCK TABLES `restaurant` WRITE;

```

```
/*!40000 ALTER TABLE `restaurant` DISABLE KEYS */;
```

```
INSERT INTO `restaurant` VALUES
```

```
(1,'Paradise','OMR',4.5,'08:15:00','21:15:00',2),(2,'Shanus','T  
Nagar',3,'08:15:00','22:00:00',3),(3,'Bombay','Velachery',4,'08:15:00','22:00:00',2),  
(4,'Sangeetha','Adyar',5,'08:15:00','22:00:00',4),(5,'Chutney','Avadi',3,'08:00:00','2  
2:00:00',5),(6,'Absolute Barbecue','T  
Nagar',4,'09:00:00','23:00:00',6),(7,'RainTree','Velachery',3,'10:00:00','23:00:00',7)  
,(8,'Hot Chips','Nungambakkam',2,'10:00:00','23:00:00',8),(9,'Aapa  
Kadai','Kelambakkam',4,'10:00:00','23:00:00',5),(10,'Barbecue  
Nation','Vadapalani',5,'10:00:00','23:00:00',7);
```

```
/*!40000 ALTER TABLE `restaurant` ENABLE KEYS */;
```

```
UNLOCK TABLES;
```

```
--
```

```
-- Dumping data for table `restauranttype`
```

```
--
```

```
LOCK TABLES `restauranttype` WRITE;
```

```
/*!40000 ALTER TABLE `restauranttype` DISABLE KEYS */;
```

```
INSERT INTO `restauranttype` VALUES
```

```
(1,'Continental'),(2,'Italian'),(3,'Asian'),(4,'Chinese'),(5,'Thai'),(6,'Greek'),(7,'Seafoo  
d'),(8,'German'),(9,'Barbecue'),(10,'Cafetaria');
```

```
/*!40000 ALTER TABLE `restauranttype` ENABLE KEYS */;
```

```
UNLOCK TABLES;
```

--

-- Dumping data for table `room`

--

LOCK TABLES `room` WRITE;

/*!40000 ALTER TABLE `room` DISABLE KEYS */;

INSERT INTO `room` VALUES (101,'Single Bedroom',1200,'Ocean View','N'),(102,'Single Bedroom',1200,'Ocean View','N'),(103,'Single Bedroom',1200,'Ocean View','N'),(104,'Single Bedroom',1200,'Ocean View','N'),(105,'Single Bedroom',1200,'Ocean View','N'),(106,'Single Bedroom',1200,'Ocean View','N'),(107,'Single Bedroom',1200,'Ocean View','N'),(108,'Single Bedroom',1200,'Ocean View','N'),(109,'Single Bedroom',1200,'Ocean View','N'),(110,'Single Bedroom',1200,'Ocean View','N'),(111,'Single Bedroom',1200,'Ocean View','Y'),(112,'Single Bedroom',1200,'Ocean View','N'),(201,'Double Bedroom',1500,'Sun View','N'),(202,'Double Bedroom',1500,'Ocean View','N'),(203,'Double Bedroom',1500,'City View','N'),(204,'Double Bedroom',1500,'Ocean View','N'),(205,'Double Bedroom',1500,'Lake View','N'),(206,'Double Bedroom',1500,'Lake View','N'),(207,'Double Bedroom ',1850,'Suite','N');

/*!40000 ALTER TABLE `room` ENABLE KEYS */;

UNLOCK TABLES;

--

-- Dumping data for table `roomservice`

--


```
LOCK TABLES `roomservice` WRITE;
```

```
/*!40000 ALTER TABLE `roomservice` DISABLE KEYS */;
```

```
INSERT INTO `roomservice` VALUES (1,'BedSheet  
Cleaning','08:00:00','23:00:00'),(2,'Bathroom  
Cleaning','08:00:00','23:00:00'),(3,'Dusting','08:00:00','23:00:00'),(4,'Private  
Masseuse','08:00:00','23:00:00'),(5,'Carpet Cleaning','10:00:00','22:00:00');
```

```
/*!40000 ALTER TABLE `roomservice` ENABLE KEYS */;
```

```
UNLOCK TABLES;
```

```
--
```

```
-- Dumping events for database 'mydb'
```

```
--
```

```
--
```

```
-- Dumping routines for database 'mydb'
```

```
--
```

```
/*!50003 DROP PROCEDURE IF EXISTS `get_available_rooms` */;
```

```
/*!50003 SET @saved_cs_client = @@character_set_client */;
```

```
/*!50003 SET @saved_cs_results = @@character_set_results */;
```

```
/*!50003 SET @saved_col_connection = @@collation_connection */;
```

```
/*!50003 SET character_set_client = utf8 */;
```

```
/*!50003 SET character_set_results = utf8 */;
```

```

/*!50003 SET collation_connection = utf8_general_ci */;

/*!50003 SET @saved_sql_mode      = @@sql_mode */;

/*!50003 SET sql_mode            = 'NO_ENGINE_SUBSTITUTION' */;

DELIMITER ;;

CREATE DEFINER=`root`@`localhost` PROCEDURE `get_available_rooms`(IN
room_type VARCHAR(50))

BEGIN

(

select * from Room where room.RoomType = room_type

and room.Status = 'Y'

);

END ;;

DELIMITER ;

/*!50003 SET sql_mode            = @saved_sql_mode */;

/*!50003 SET character_set_client = @saved_cs_client */;

/*!50003 SET character_set_results = @saved_cs_results */;

/*!50003 SET collation_connection = @saved_col_connection */;

/*!50003 DROP PROCEDURE IF EXISTS `get_booked_rooms` */;

/*!50003 SET @saved_cs_client     = @@character_set_client */;

/*!50003 SET @saved_cs_results   = @@character_set_results */;

/*!50003 SET @saved_col_connection = @@collation_connection */;

```

```

/*!50003 SET character_set_client = utf8 */;

/*!50003 SET character_set_results = utf8 */;

/*!50003 SET collation_connection = utf8_general_ci */;

/*!50003 SET @saved_sql_mode      = @@sql_mode */;

/*!50003 SET sql_mode            = 'NO_ENGINE_SUBSTITUTION' */;

DELIMITER ;;

CREATE DEFINER=`root`@`localhost` PROCEDURE `get_booked_rooms`(IN
room_type VARCHAR(50))

BEGIN

(

select * from Room where room.RoomType = room_type

and room.Status = 'N'

);

END ;;

DELIMITER ;

/*!50003 SET sql_mode            = @saved_sql_mode */;

/*!50003 SET character_set_client = @saved_cs_client */;

/*!50003 SET character_set_results = @saved_cs_results */;

/*!50003 SET collation_connection = @saved_col_connection */;

/*!50003 DROP PROCEDURE IF EXISTS `get_customers` */;

/*!50003 SET @saved_cs_client    = @@character_set_client */;

```

```

/*!50003 SET @saved_cs_results    = @@character_set_results */;

/*!50003 SET @saved_col_connection = @@collation_connection */;

/*!50003 SET character_set_client  = utf8 */;

/*!50003 SET character_set_results = utf8 */;

/*!50003 SET collation_connection  = utf8_general_ci */;

/*!50003 SET @saved_sql_mode      = @@sql_mode */;

/*!50003 SET sql_mode              = 'NO_ENGINE_SUBSTITUTION' */;

DELIMITER ;;

CREATE DEFINER=`root`@`localhost` PROCEDURE `get_customers`(IN today_date
date)

BEGIN

SELECT x.Customer_CustomerID,x.Room_RoomID,c.customer_firstName

FROM customer_allotted_room x INNER JOIN Customer c

on x.customer_customerID = C.CustomerID

WHERE

x.checkout_date >= today_date AND x.checkin_date <= today_date ;

END ;;

DELIMITER ;

/*!50003 SET sql_mode              = @saved_sql_mode */;

/*!50003 SET character_set_client  = @saved_cs_client */;

/*!50003 SET character_set_results = @saved_cs_results */;

/*!50003 SET collation_connection = @saved_col_connection */;

```

```

/*!50003 DROP PROCEDURE IF EXISTS `get_roomService_revenue` */;

/*!50003 SET @saved_cs_client      = @@character_set_client */;

/*!50003 SET @saved_cs_results    = @@character_set_results */;

/*!50003 SET @saved_col_connection = @@collation_connection */;

/*!50003 SET character_set_client = utf8 */;

/*!50003 SET character_set_results = utf8 */;

/*!50003 SET collation_connection = utf8_general_ci */;

/*!50003 SET @saved_sql_mode      = @@sql_mode */;

/*!50003 SET sql_mode              = 'NO_ENGINE_SUBSTITUTION' */;

DELIMITER ;;

CREATE DEFINER=`root`@`localhost` PROCEDURE `get_roomService_revenue`(IN
today_date date)

BEGIN

(

select sum(RoomServiceCost) from customer_gets_roomservice

where customer_gets_roomservice.RoomServiceDate = today_date

);

END ;;

DELIMITER ;

/*!50003 SET sql_mode              = @saved_sql_mode */;

```

```

/*!50003 SET character_set_client = @saved_cs_client */;

/*!50003 SET character_set_results = @saved_cs_results */;

/*!50003 SET collation_connection = @saved_col_connection */;

/*!50003 DROP PROCEDURE IF EXISTS `moneyFromFacility` */;

/*!50003 SET @saved_cs_client      = @@character_set_client */;

/*!50003 SET @saved_cs_results    = @@character_set_results */;

/*!50003 SET @saved_col_connection = @@collation_connection */;

/*!50003 SET character_set_client = utf8 */;

/*!50003 SET character_set_results = utf8 */;

/*!50003 SET collation_connection = utf8_general_ci */;

/*!50003 SET @saved_sql_mode      = @@sql_mode */;

/*!50003 SET sql_mode              = 'NO_ENGINE_SUBSTITUTION' */;

```

```
DELIMITER ;;
```

```
CREATE DEFINER=`root`@`localhost` PROCEDURE `moneyFromFacility`(IN
today_date date)
```

```
begin
```

```
select sum(rate) as totalMoneyFromFacility from facility f
```

```
inner join customer_uses_facility c
```

```
on c.Facility_FacilityID = f.FacilityID
```

```
where c.UseDate = today_date;
```

```
end ;;
```

```
DELIMITER ;
```

```

/*!50003 SET sql_mode          = @saved_sql_mode */ ;

/*!50003 SET character_set_client = @saved_cs_client */ ;

/*!50003 SET character_set_results = @saved_cs_results */ ;

/*!50003 SET collation_connection = @saved_col_connection */ ;

/*!50003 DROP PROCEDURE IF EXISTS `RoomServiceDetails` */;

/*!50003 SET @saved_cs_client    = @@character_set_client */ ;

/*!50003 SET @saved_cs_results  = @@character_set_results */ ;

/*!50003 SET @saved_col_connection = @@collation_connection */ ;

/*!50003 SET character_set_client = utf8 */ ;

/*!50003 SET character_set_results = utf8 */ ;

/*!50003 SET collation_connection = utf8_general_ci */ ;

/*!50003 SET @saved_sql_mode     = @@sql_mode */ ;

/*!50003 SET sql_mode           = 'NO_ENGINE_SUBSTITUTION' */ ;

```

```

DELIMITER ;;

```

```

CREATE DEFINER=`root`@`localhost` PROCEDURE `RoomServiceDetails`(IN
today_date date)

```

```

begin

```

```

    select C.customer_FirstName,C.customer_lastName,E.EmployeeFirstName as
'Employee

```

```

Handled',CR.RoomServiceDate,CR.RoomServiceDate,RS.RoomServiceDetails

```

```

from customer_gets_roomservice CR

```

```

inner join Customer C

```

```

on C.CustomerID = CR.Customer_CustomerID

inner join hotelemmployee E

on CR.EmployeeID = E.EmployeeID

inner join roomservice RS

on CR.RoomService_RoomServiceID = RS.RoomServiceID

where CR.RoomServiceDate = today_date;

end ;;

DELIMITER ;

/*!50003 SET sql_mode          = @saved_sql_mode */ ;

/*!50003 SET character_set_client = @saved_cs_client */ ;

/*!50003 SET character_set_results = @saved_cs_results */ ;

/*!50003 SET collation_connection = @saved_col_connection */ ;

/*!50003 DROP PROCEDURE IF EXISTS `totalLaundryFromSingleDate` */;

/*!50003 SET @saved_cs_client    = @@character_set_client */ ;

/*!50003 SET @saved_cs_results   = @@character_set_results */ ;

/*!50003 SET @saved_col_connection = @@collation_connection */ ;

/*!50003 SET character_set_client = utf8 */ ;

/*!50003 SET character_set_results = utf8 */ ;

/*!50003 SET collation_connection = utf8_general_ci */ ;

/*!50003 SET @saved_sql_mode     = @@sql_mode */ ;

/*!50003 SET sql_mode            = 'NO_ENGINE_SUBSTITUTION' */ ;

```


DELIMITER ;;

CREATE DEFINER=`root`@`localhost` PROCEDURE
`totalLaundryFromSingleDate`(IN date_given Date)

BEGIN

(

select sum(LaundryCost) from laundry

where laundryDate = date_given

);

end ;;

DELIMITER ;

/*!50003 SET sql_mode = @saved_sql_mode */ ;

/*!50003 SET character_set_client = @saved_cs_client */ ;

/*!50003 SET character_set_results = @saved_cs_results */ ;

/*!50003 SET collation_connection = @saved_col_connection */ ;

/*!40103 SET TIME_ZONE=@OLD_TIME_ZONE */;

/*!40101 SET SQL_MODE=@OLD_SQL_MODE */;

/*!40014 SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS */;

/*!40014 SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS */;

/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;

/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;

/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;

```
/*!40111 SET SQL_NOTES=@OLD_SQL_NOTES */;
```

```
-- Dump completed on 2017-12-14 3:29:09
```