## SQL HANDS ON

### SQL Basics with DML and DDL Statements

SCHEMA

## CAR

## Car rental system - Create Table

Refer to the given schema. Write a query to create the Owners table with the specified columns and constraints.

Note: Letters in bold represents the table name

NOTE: Maintain the same sequence of column order, as specified in the question description

**create table OWNERS(owner\_id varchar(20) primary key,**

**owner\_name varchar(20),**

**address varchar(20),**

**phone\_no bigint,**

**email\_id varchar(20));**

## Car rental system - add new column

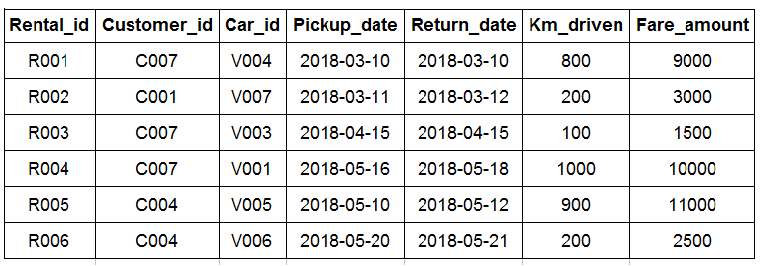
Refer to the given schema. Assume, CARS table has been already created. Write an appropriate query for the given requirement.

Requirement 1: Add a new column Car\_Regno VARCHAR(10)  to the Cars table.  
  
Note: Letters in the bold represents the attribute name.

**alter table CARS ADD COLUMN car\_regno varchar(10);**

## Car rental system - Insert values

Refer to the given schema diagram. Insert the below records into Rentals Table. Assume the rentals table has been already created.



Note: Letters in bold represent the attributes.

NOTE: Maintain the same sequence of column order, as specified in the question description

**insert into rentals values('R001','C007','V004','2018-03-10','2018-03-10',800,9000); insert into rentals values('R002','C001','V007','2018-03-11','2018-03-12',200,3000); insert into rentals values('R003','C007','V003','2018-04-15','2018-04-15',100,1500); insert into rentals values('R004','C007','V001','2018-05-16','2018-05-18',1000,10000); insert into rentals values('R005','C004','V005','2018-05-10','2018-05-12',900,11000); insert into rentals values('R006','C004','V006','2018-05-20','2018-05-21',200,2500);****select \* from rentals order by 1;**

## Car & owner details based on car type

Write a query to display car id, car name and owner id of all the cars whose car type is 'Hatchback' or 'SUV'.  Sort the result based on car id.

(Hint: Use  CARS tables to retrieve records. Data is case-sensitive. E.g: Car\_type='Hatchback'. Use IN operator)

NOTE: Maintain the same sequence of column order, as specified in the question description

**select car\_id,car\_name,owner\_id**

**from cars where car\_type='SUV' or car\_type='Hatchback'**

**order by car\_id;**

## Car details based on type and name

Write a query to display car id, car name and car type of Maruthi company 'Sedan' type cars.  Sort the result based on car id.

(HINT : Use Cars table to retrieve records.car name='Maruthi Swift'.car type='Sedan'.Data is case sensitive.)

NOTE: Maintain the same sequence of column order, as specified in the question description

**Select car\_id,car\_name,car\_type from cars where car\_name like 'Maruthi%' and car\_type='Sedan' order by 1;**

## Customers having gmail id

Write a query to display customer id, customer name, address, and phone number of customers having Gmail id.  Sort the result based on customer id.

(HINT: Use Customers table to retrieve records. Email id='xxxxx@gmail.com'.Data is case sensitive.)

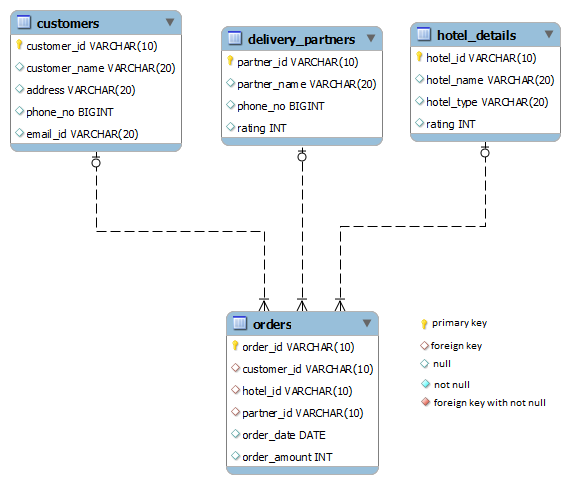
NOTE: Maintain the same sequence of column order, as specified in the question description

**select customer\_id,customer\_name,address,phone\_no**

**from customers**

**where email\_id like '%gmail.com'**

**order by 1**;

SCHEMA

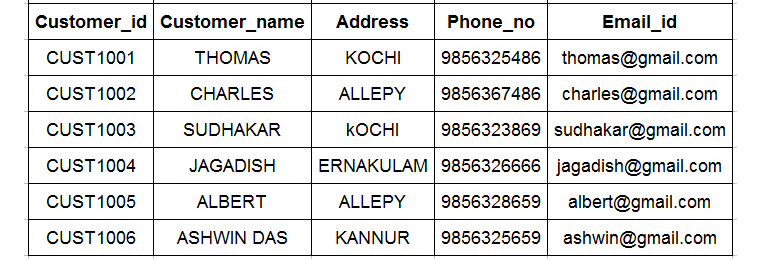
## Hunger eats - change datatype

Refer to the given schema. Assume that the 'Customers' table has been already created.  
Write a query to change the data type of the field customer\_id in Customers table to int.  
  
Note: Letters in bold represents the attribute name

**ALTER TABLE customers MODIFY customer\_id int;**

## Hunger eats - update table

Refer to the given schema diagram and sample records inserted into the Customers table



**UPDATE CUSTOMERS**

**SET Phone\_no = 9876543210**

**WHERE customer\_id = 'CUST1004';**

1. **Delivery Partner details based on rating**

Write a query to display partner id, partner name, phone number of delivery partners whose rating is between 3 to 5, sort the result based on partner id.

(Hint: Use Delivery\_partners table to retrieve records.)

NOTE: Maintain the same sequence of column order, as specified in the question description

**select partner\_id,partner\_name,phone\_no**

**from delivery\_partners**

**where rating between 3 and 5 order by 1**

## Hunger eats - Change the field name

Write appropriate query/queries for the given requirement. Assume, Hotel\_Details table has been already created.  
  
Requirement 1: Change the name of the existing field Rating to Hotel\_Rating in the  Hotel\_Details table.  
  
Note: Letters in the bold represents the attributes

**ALTER TABLE HOTEL\_DETAILS CHANGE rating hotel\_rating int;**

### Querying Database - Operators,  Aggregate, String, Date Functions

SCHEMA



## Rental details based on date

Write a query to display rental id, car id, customer id and km driven of rentals taken during 'AUGUST 2019'.  Sort the result based on rental id.

(HINT : Use Rentals table to retrieve records. Eg: return date: 2019-08-12 )

NOTE: Maintain the same sequence of column order, as specified in the question description

**select rental\_id,car\_id,customer\_id,km\_driven from rentals where extract(month from return\_date)=08 and extract(year from return\_date)=2019 order by 1;**

## Customer contact details

Write a query to display the customer id, customer name and contact details of customers. If address is missing, display the email id. If both address and email is missing then display ‘NA’. Give an alias name as CONTACT\_DETAILS.Sort the results based on customer id in ascending order.

(HINT : Use Customers table to retrieve records.)

**select customer\_id,customer\_name,coalesce(address,email\_id,'NA') as Contact\_Details from customers**

**order by 1;**

## No of time rented by each car

Write a query to display car id and number of times car taken for rental. Give an alias name to the number of times car taken for rental as 'NO\_OF\_TRIPS'. Sort the records based on car id in ascending order.

(HINT: Use Rentals table to retrieve records.)

**select car\_id,count(rental\_id) as No\_Of\_Trips from rentals group by car\_id order by 1;**

## Credential details

Refer to the schema. Write a query to display the username and password of all owners. Give an alias name as USERNAME and PASSWORD. Sort the result based on the username in ascending order.

Username and password is generated as mentioned below

USERNAME: concatenate the owner's name with owner id.

PASSWORD: concatenate first 3 character of owner name with owner id.

For Example:

Owner\_id - 01

Owner\_name - jeeva

**Sample Output:**

|  |  |
| --- | --- |
| USERNAME | PASSWORD |
| jeeva01 | jee01 |

(HINT: Use Owners table to retrieve records.)

NOTE: Maintain the same sequence of column order, as specified in the question description

**select concat(owner\_name,owner\_id) as Username,concat(substr(owner\_name,1,3),owner\_id) as Password**

**from Owners order by username;**

## Maruthi car owner details

Write a query to display distinct owner id, owner name, address, and phone no of owners who owns 'Maruthi' company car.  Sort the result based on owner id.

Note: If car\_name contains a string 'Maruthi' it is a Maruthi company car.

Example: 'Maruthi swift','Maruthi 800'

(HINT: Use Owners and Cars tables to retrieve records.)

NOTE: Maintain the same sequence of column order, as specified in the question description

**select distinct o.owner\_id,owner\_name,address,phone\_no**

**from owners o join cars c on o.owner\_id=c.owner\_id**

**where car\_name like 'Maruthi%' order by 1;**

## Cars not taken for rent

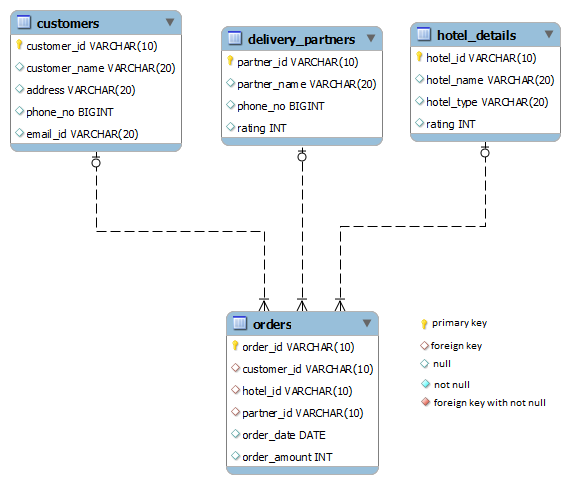
Write a query to display car id, car name, car type of cars which was not taken for rent.  Sort the result based on car id.

(HINT: Use Cars and Rentals tables to retrieve records.)

NOTE: Maintain the same sequence of column order, as specified in the question description

**select car\_id,car\_name,car\_type from cars where car\_id not in(select car\_id from rentals) order by 1;**

SCHEMA



## Hotels that took order based on month

Write a query to display distinct hotel id, hotel name, and rating of hotels that have taken order in the month of July. Sort the result based on hotel id in ascending order.

(HINT: Use Hotel\_details and  Orders tables to retrieve records.Order date='2019-07-14')

NOTE: Maintain the same sequence of column order, as specified in the question description

**select hotel\_id,hotel\_name,rating from hotel\_details**

**where hotel\_id in(select hotel\_id from orders**

**where extract(month from order\_date)=07) order by 1;**

## Review of delivery partner based on rating

Write a query to display partner id,partner name and review of the delivery partner, give alias name for partner review as 'REVIEW', sort the result based on partner id in ascending order.

Note: Review is based on the following condition

IF rating>=4 ----> 'GOOD'

IF rating between >=2 and <4 -----> 'AVERAGE'

IF rating <2 ------------>'WORST'

(HINT : Use Delivery\_partners table to retrieve records.)

**select partner\_id,partner\_name,**

**CASE when rating>=4 then 'GOOD'**

**when rating between 2 and 4 then 'AVERAGE'**

**when rating<2 then 'WORST'**

**end**

**as REVIEW from delivery\_partners order by 1;**

## Total sale daywise

Write a query to display order\_date, total order amount in each day. Give an alias name for total order amount as ‘TOTAL\_SALE’. Sort the result based on order\_date.

(HINT : Use Orders table to retrieve records.)

NOTE: Maintain the same sequence of column order, as specified in the question description

**select order\_date,sum(order\_amount) as Total\_Sale from orders**

**group by order\_date order by 1;**

## Hotels not taken orders in a specific month

Write a query to display hotel id, hotel name and hotel type of hotels which has not taken any orders in the month of 'MAY 19'. Sort the result based on hotel id in ascending order.

(HINT: Use Hotel\_details and Orders tables to retrieve records. Eg: order\_date= 2019-05-12)

NOTE: Maintain the same sequence of column order, as specified in the question description

**select hotel\_id,hotel\_name,hotel\_type from hotel\_details**

**where hotel\_id not in(select**

**hotel\_id from orders**

**where extract(month from order\_date)=05 and extract(year from order\_date)=2019) order by 1;**

## Hotels that took order more than five times

Write a query to display hotel id, hotel name, and number of orders taken by hotels that have taken orders more than 5 times. Give an alias name for number of orders as 'NO\_OF\_ORDERS'.sort the result based on hotel id in ascending order.

(HINT: Use Hotel\_details and Orders tables to retrieve records.)

NOTE: Maintain the same sequence of column order, as specified in the question description

**select hotel\_id,hotel\_name,count(\*) no\_of\_orders**

**from hotel\_details natural join orders**

**group by hotel\_id,hotel\_name**

**having count(\*)>5 order by 1;**

## Hotel\_info

Refer to the schema. Write a query to display the hotel name along with the type. Display the details in the below format.

Give an alias name as HOTEL\_INFO. Sort the result in descending order.

For Example:

          Hotel\_name - 'A2B'

          Hotel\_type - 'VEG'

**Sample Output:**

|  |
| --- |
| HOTEL\_INFO |
| A2B is a VEG hotel |

(HINT: Use Hotel\_details table to retrieve records.)

NOTE: Maintain the same sequence of column order, as specified in the question description

**select concat(hotel\_name,' is a ',hotel\_type,' hotel') as Hotel\_info**

**from hotel\_details**

**order by hotel\_info desc;**

## Customer mail details

Refer to the schema. Write a query to display the customer mail details. Display the details in the below format.

Give an alias name as CUSTOMER\_MAIL\_INFO. Sort the result in ascending order.

For Example:

          Customer\_id - 'CUST001'

          Email\_id - 'mano@hotmail.com'

**Sample Output:**

|  |
| --- |
| CUSTOMER\_MAIL\_INFO |
| CUST001 mail id is mano@hotmail.com |

(HINT: Use Customers table to retrieve records.)

NOTE: Maintain the same sequence of column order, as specified in the question description

**select concat(customer\_id,' mail id is ',email\_id) as CUSTOMER\_MAIL\_INFO**

**from customers**

**order by customer\_mail\_info;**

## Order details

Write a query to display order id, customer name, hotel name, and order amount of all orders. Sort the result based on order id in ascending order.

(HINT: Use Customers, Hotel\_details and Orders tables to retrieve records.)

NOTE: Maintain the same sequence of column order, as specified in the question description

**select order\_id,customer\_name,hotel\_name,order\_amount**

**from customers**

**join orders on customers.customer\_id=orders.customer\_id**

**join hotel\_details on hotel\_details.hotel\_id=orders.hotel\_id**

**order by 1;**