Car Rental ManagementSystem

1. **Understand the given problem statement and apply W3H to analyze it.**

What?	How?
1.Admin	**1.Admin**
a.Add/read/update/delete car details	**a.Add/delete/update/view car details**
1.Admin can add,delete,update,view details?	Method 1:Using the car name
Ans:Yes,Admin can handle those things.	Method 2:Using the car brand and model
2.Who is responsible for checking the car availability?	Method 3:Using the car specific id
Ans:Admin is responsible for checking the availability	**b.Add/delete/update/view customer details**
3.Who is managing the rental car details and customer details?	Method 1:Using the customer id
Ans:Admin is managed those things	Method 2:Using the customer name
4.How the admin can identify the car details?	Method 3:Using the customer Email and mobile number
Ans:using specific car name and id	**c.Add/delete/update/view rental details**
5.How can the admin login into the application?	Method 1:Using rental database
Ans:using Email id and Mobile number	Method 2:Using rental details
b.Add/read/update/delete customer details	Method 3:Both
1.Admin has to add,delete,update,view the customer details?	**d.Register/login**
Ans:Yes,they managed those things.	Method 1:Using id
2.How the admin has contacted the customer?	Method 2 :Using id ,email and mobile number
Ans:Using customer mobile number or Email id	Method 3:Using name
c.Add/read/update/delete rental details	**2.Customer**
1.Who will manage the rental?	**a.Add/delete/update/view**
Ans:Admin	Method 1:Using the customer id
	Method 2:Using the Email and mobile number
2.Customer	Method 3:Both
a.Add/delete/update/view	**b.Register/login**
	Method 1:Using customer id

1.Customer has to add,delete,update,view their details? Ans:Yes,Customer has only add,update,delete,view their own details only **2.Admin has viewed the customer details?** Ans:Yes,they using customer specific id **3.How customer contact admin?** Ans:Using Email id or Mobile number **b.Register/login** **1.How do customers register/login into the application?** Ans:Using their customer id , email id and mobile number **c.Car Rental** **1.What is the procedure for a customer to rent a car?** Ans: using the application for booking a car for rental 2.How much is the duration to return a car? Ans: It's based on a specific company,handled by only admin. **d.) Payment** 1. What details are required for payment? Ans: In order to show accurate rental amount to customer	Method 2 :Using customer id ,Email id and mobile number Method 3:Using name **c.Car Rental** Method 1:Using customer id and car id Method 2 :Using Email id and mobile number Method 3:Using name **d.) Payment** Method 1:Using customer id ,car id,rental id and transaction id Method 2 :Using customer id and car id Method 3:Using customer id and rental id

Why?	Why Not?
1.Admin **a.Add/delete/update/view car details** Method 3:Using the car specific id (Using specific for car it will easily find the car details) **b.Add/delete/update/view customer details** Method 1:Using the customer id (Using customer id unique and find the customer details easily) **c.Add/delete/update/view rental details** Method 1:Using rental database Method 2:Using rental details Method 3:Both (Using both method is efficiently handle the rental details) **d.Register/login** Method 2 :Using id and mobile number (Using id is easy to login and mobile number is easy to receive the otp) **2.Customer** **a.Add/delete/update/view** Method 1:Using the customer id (Its unique and avoid ambiguity) **b.Register/login** Method 2 :Using customer id and mobile number (Using customer id is easy to login and mobile number is easy to receive the otp)	**1.Admin** a.Using brand name and car name is not unique because,same brand name car is more. b.Using employee names is not an efficient manner,it confuses more than one employee's name. c.Using customer name and id is take much time to find a employee details e.Register/login into portal is lessely consider the username and their unique id **2.Customer** a.Using Email id mobile number take much to gathering a details b.Register/login into portal is lessely consider the username and their unique id c.Car rental is also based on car and customer id ,less possibility to their personal information

c.Car Rental Method 1:Using customer id and car id (Using this method,it is easy to find the availability of car details and maintain the customer details) **d.) Payment** Method 1:Using customer id ,car id,rental id,and transaction id . (Using this method ,it is easy to show the accurate amount to pay)	

2. **Bring out the list of tables and attributes required for the database design.**

User table:
(**Userid** (**primary key**),
Uname,
Gender,
Address,
Phn_no,
email)

Admin table:
(**Adminid** (**primary key**),
Adminname,
Adminemail,
Adminaddress,
Adminphn,
User id(**foreign key from user table**))

Rental table:
(**rentid** (**primary key**),
Veh id(foreign key),
Start date,
End date,
paymentid(**foreign key**),
Total cost,
Userid (**foreign key**))

Vehicle table:
(**veh id** (**primary key**),
Model,
Type,

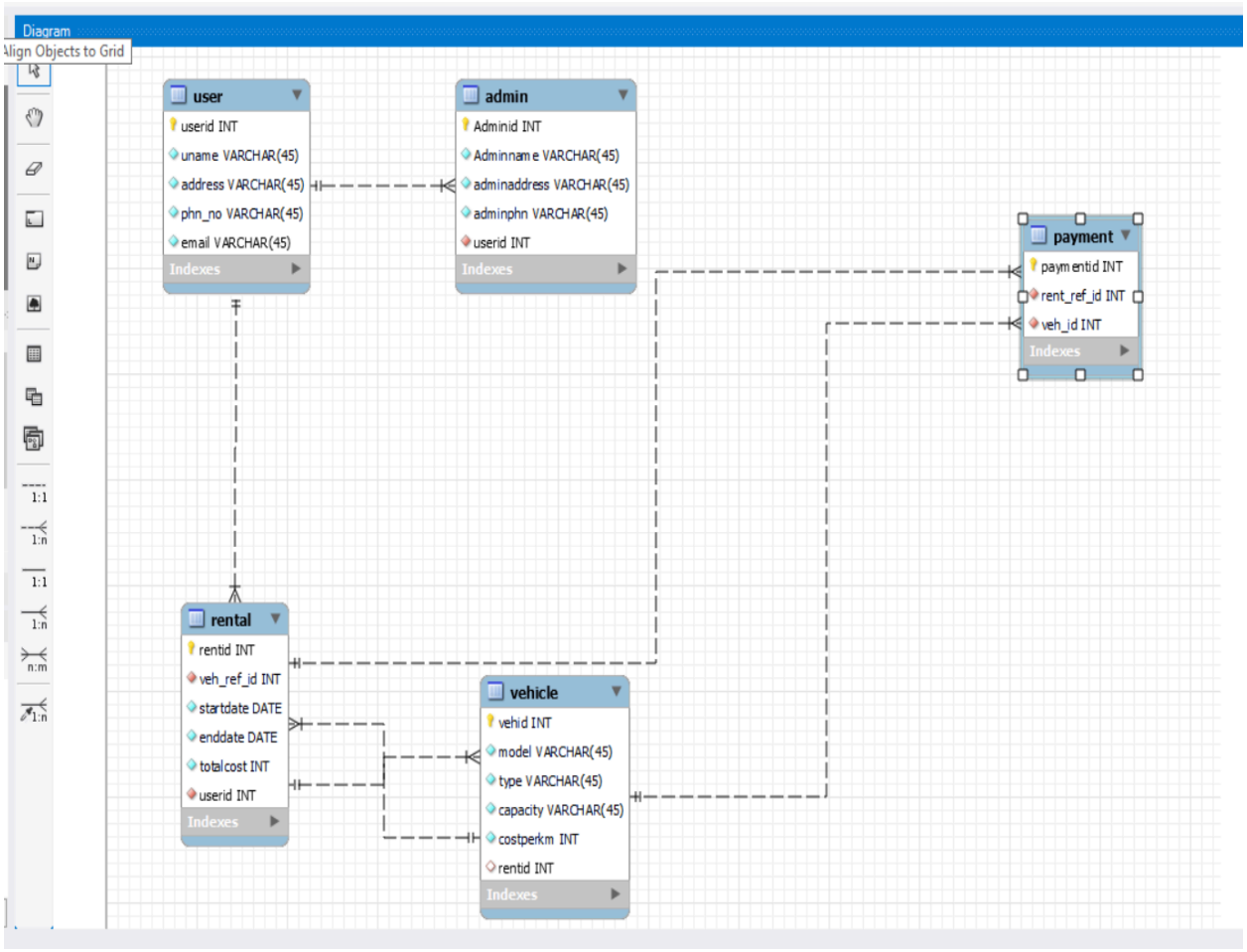
Capacity,
Brand,
Cost per km,
userid(**foreign key**))

Payment table:
(**payment id**(**primary key**),
rent_ref_id(**foreign key**),
Veh_id (**foreign key**))

3.Apply Normalization:

Table is normalized while creating.

4.Draw an ER Diagram.



5.Perform the CRUD operations.

Read:

```
mysql> select * from user;
+--------+-----------+--------+----------------+------------+---------------------+
| userid | uname     | gender | address        | phn_no     | email               |
+--------+-----------+--------+----------------+------------+---------------------+
|   1001 | Srini     | male   | Thiruvanmiyur  | 9489274502 | srini@gmail.com     |
|   1002 | Kaviya    | female | Sozhinganallur | 9812784503 | kaviya@gmail.com    |
|   1003 | Sundar    | male   | Avadi          | 7496128530 | sundar@gmailcom     |
|   1004 | Prashanth | male   | Vandalur       | 8796451320 | prashanth@gmail.com |
|   1005 | Ranjitha  | female | Anna nagar     | 8574961320 | ranjitha@gmail.com  |
+--------+-----------+--------+----------------+------------+---------------------+
5 rows in set (0.00 sec)
```

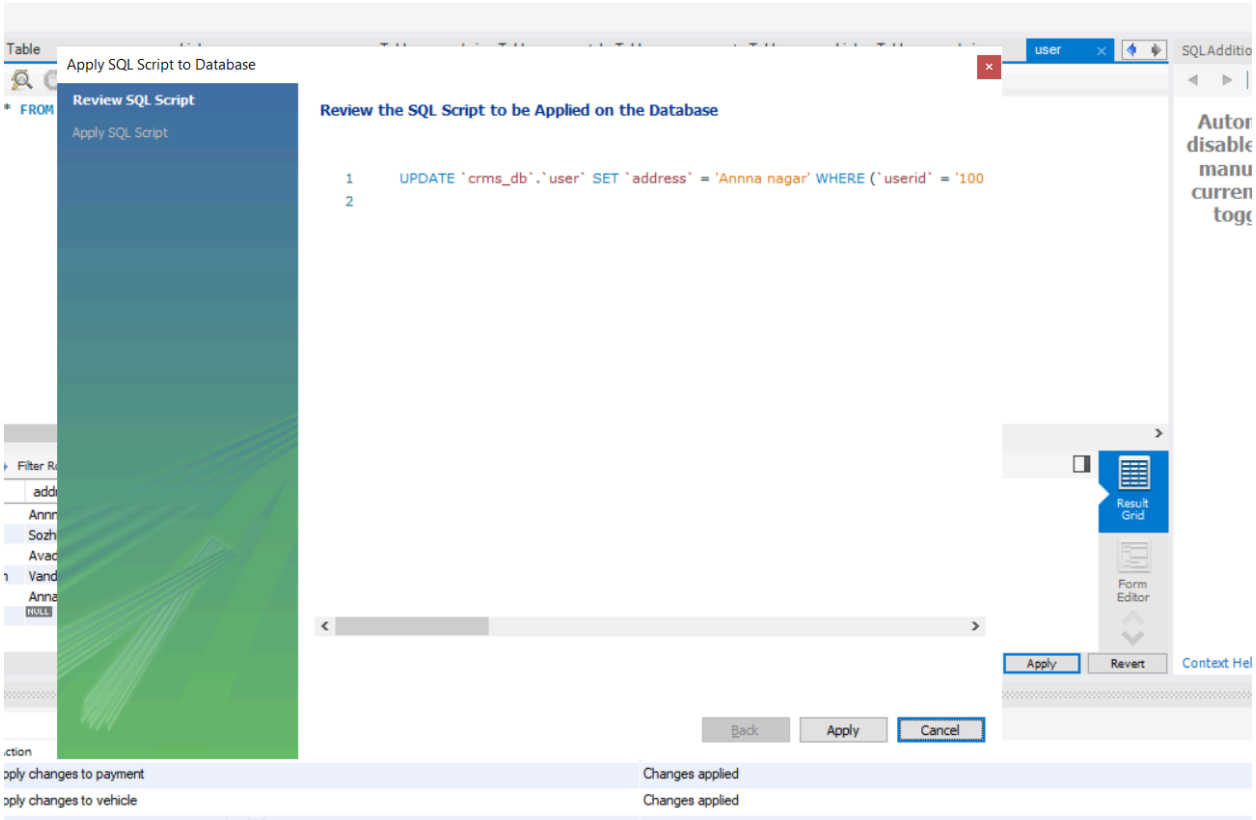
```
mysql> select * from admin;
+---------+-----------+---------------------+--------------+------------+--------+
| Adminid | Adminname | adminemail          | adminaddress | adminphn   | userid |
+---------+-----------+---------------------+--------------+------------+--------+
|    1101 | Surya     | surya@gmail.com     | Thoraipakam  | 7410852096 |   1001 |
|    1102 | Vino      | vino@gmail.com      | Siruseri     | 8520741096 |   1002 |
|    1103 | Sathyan   | sathyan@gmail.com   | Maduravoyal  | 9630741258 |   1003 |
|    1104 | Jothi     | jothi@gmail.com     | Poonamallee  | 8574961230 |   1004 |
|    1105 | Varshinee | varshinee@gmail.com | Thiruvanmiyur| 9685741232 |   1005 |
+---------+-----------+---------------------+--------------+------------+--------+
5 rows in set (0.00 sec)
```

```
ysql> select * from rental;
--------+------------+------------+------------+------------+-----------+--------+
 rentid | veh_ref_id | startdate  | enddate    | payment_id | totalcost | userid |
--------+------------+------------+------------+------------+-----------+--------+
   2001 |       3001 | 2023-10-11 | 2023-10-12 |        501 |      5600 |   1001 |
   2002 |       3002 | 2023-10-13 | 2023-10-14 |        502 |      5500 |   1002 |
   2003 |       3003 | 2023-10-18 | 2023-10-19 |        503 |      4680 |   1003 |
   2004 |       3004 | 2023-10-20 | 2023-10-21 |        504 |      1500 |   1004 |
   2005 |       3005 | 2023-10-22 | 2023-10-25 |        505 |      4000 |   1005 |
--------+------------+------------+------------+------------+-----------+--------+
 rows in set (0.00 sec)
```

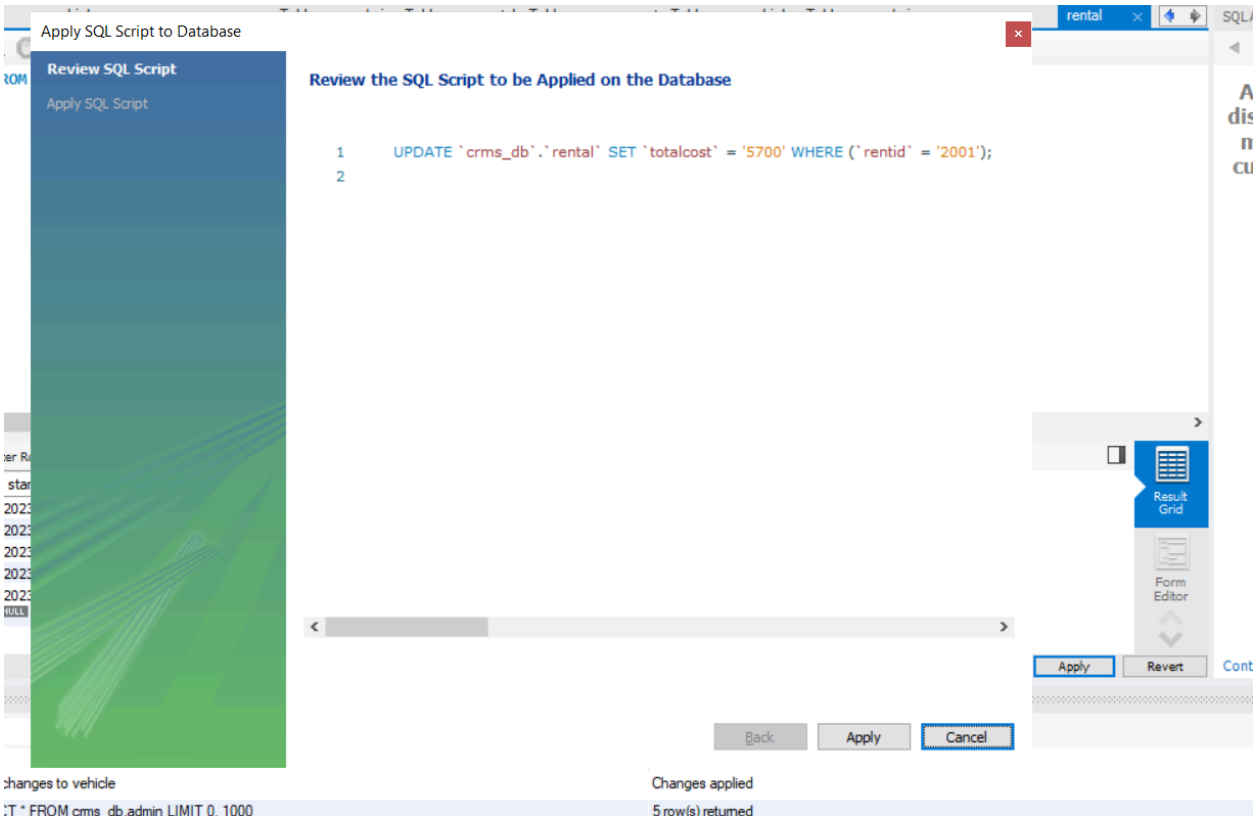
```
mysql> select * from vehicle
    -> ;
+-------+---------------+-----------+----------+-----------+-----------+--------+
| vehid | model         | type      | capacity | brand     | costperkm | rentid |
+-------+---------------+-----------+----------+-----------+-----------+--------+
|  3001 | Innova Crysta | xuv       | 8        | Toyota    |        25 |   2001 |
|  3002 | Safari        | xuv       | 7        | Tata      |        24 |   2002 |
|  3003 | Seltos        | mid xuv   | 5        | Kia       |        20 |   2003 |
|  3004 | Virtus        | mid sedon | 5        | Volswagon |        20 |   2004 |
|  3005 | I20           | micro     | 5        | Hyundai   |        15 |   2005 |
+-------+---------------+-----------+----------+-----------+-----------+--------+
5 rows in set (0.00 sec)
```

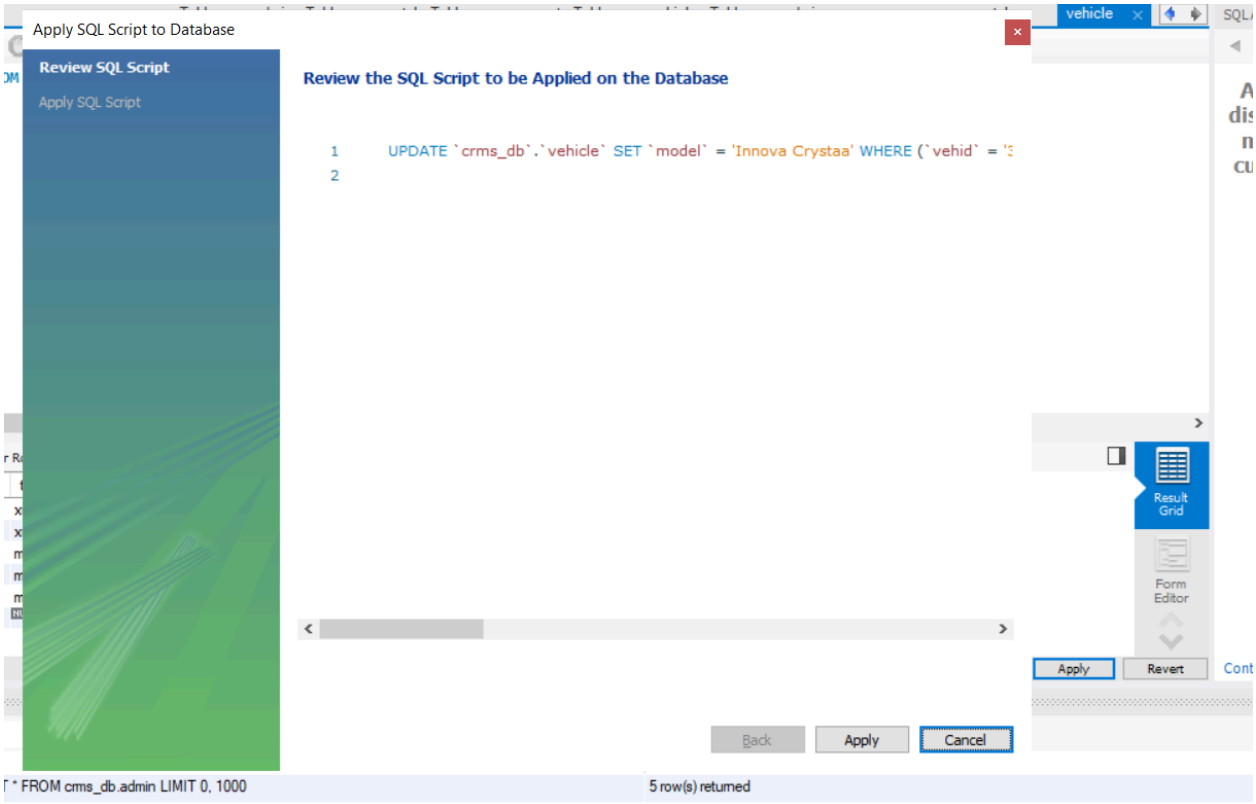
```
mysql> select * from payment;
+-----------+-------------+--------+
| paymentid | rent_ref_id | veh_id |
+-----------+-------------+--------+
|       501 |        2001 |   3001 |
|       502 |        2002 |   3002 |
|       503 |        2003 |   3003 |
|       504 |        2004 |   3004 |
|       505 |        2005 |   3005 |
+-----------+-------------+--------+
5 rows in set (0.00 sec)
```

Update:

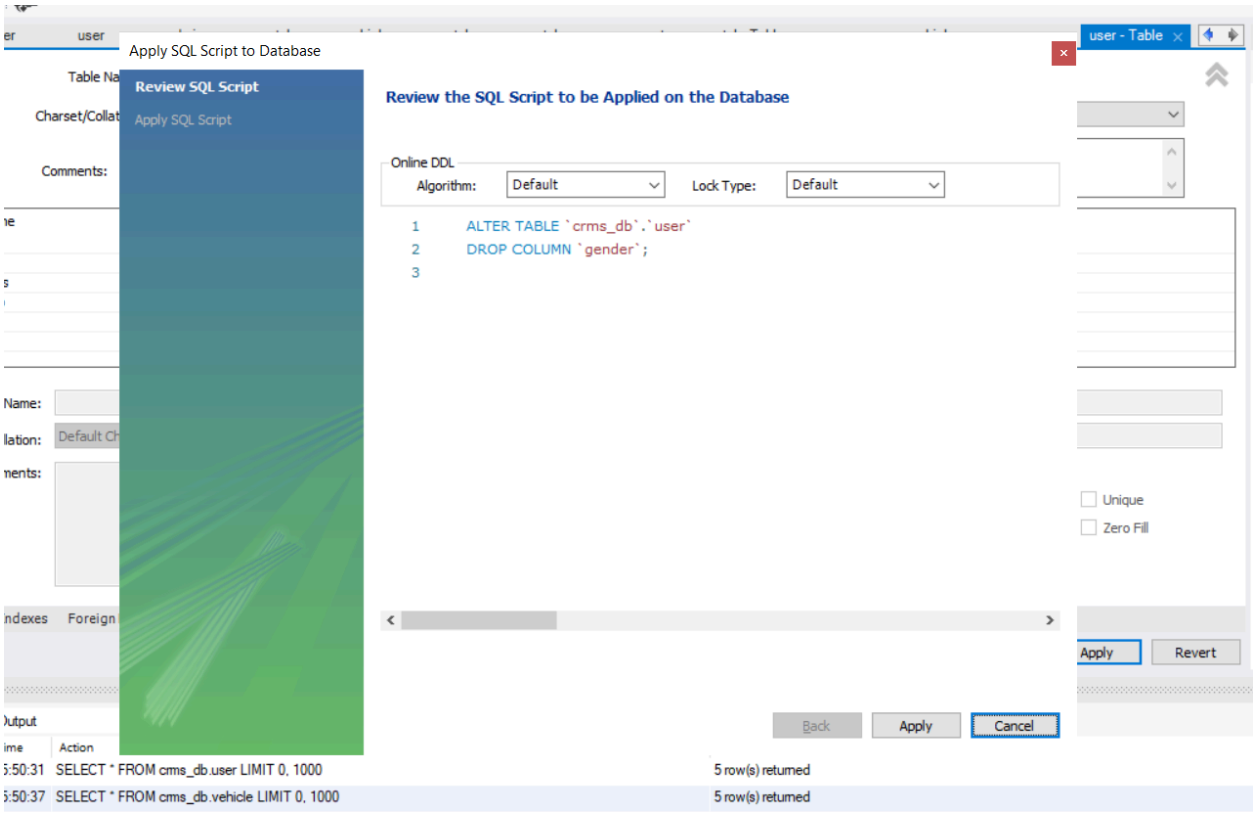


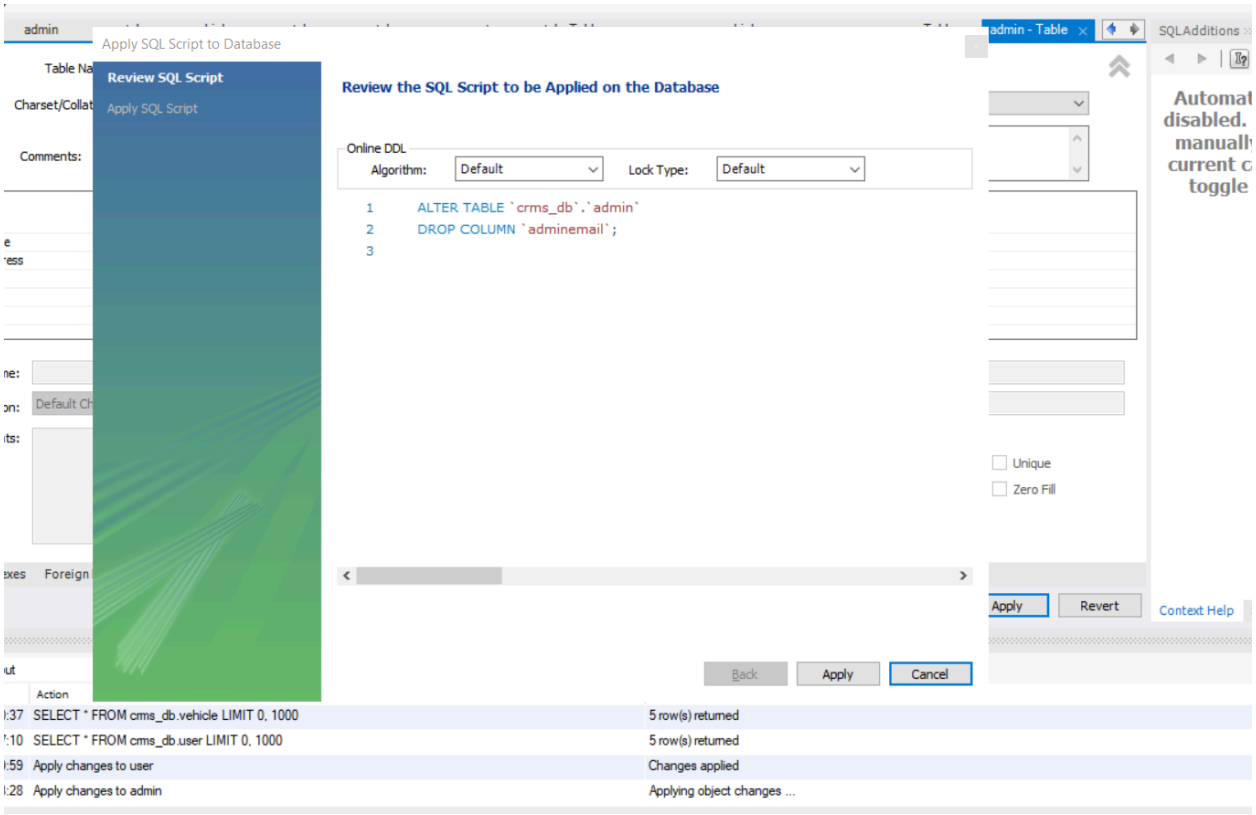

Apply SQL Script to Database
Review SQL Script
Apply SQL Script
Review the SQL Script to be Applied on the Database
1 UPDATE `crms_db`.`admin` SET `adminaddress` = 'Vaanagaram' WHERE (`Admini
2
admin
SQLAddit
Result Grid
Form Editor
Apply
Revert
Context H

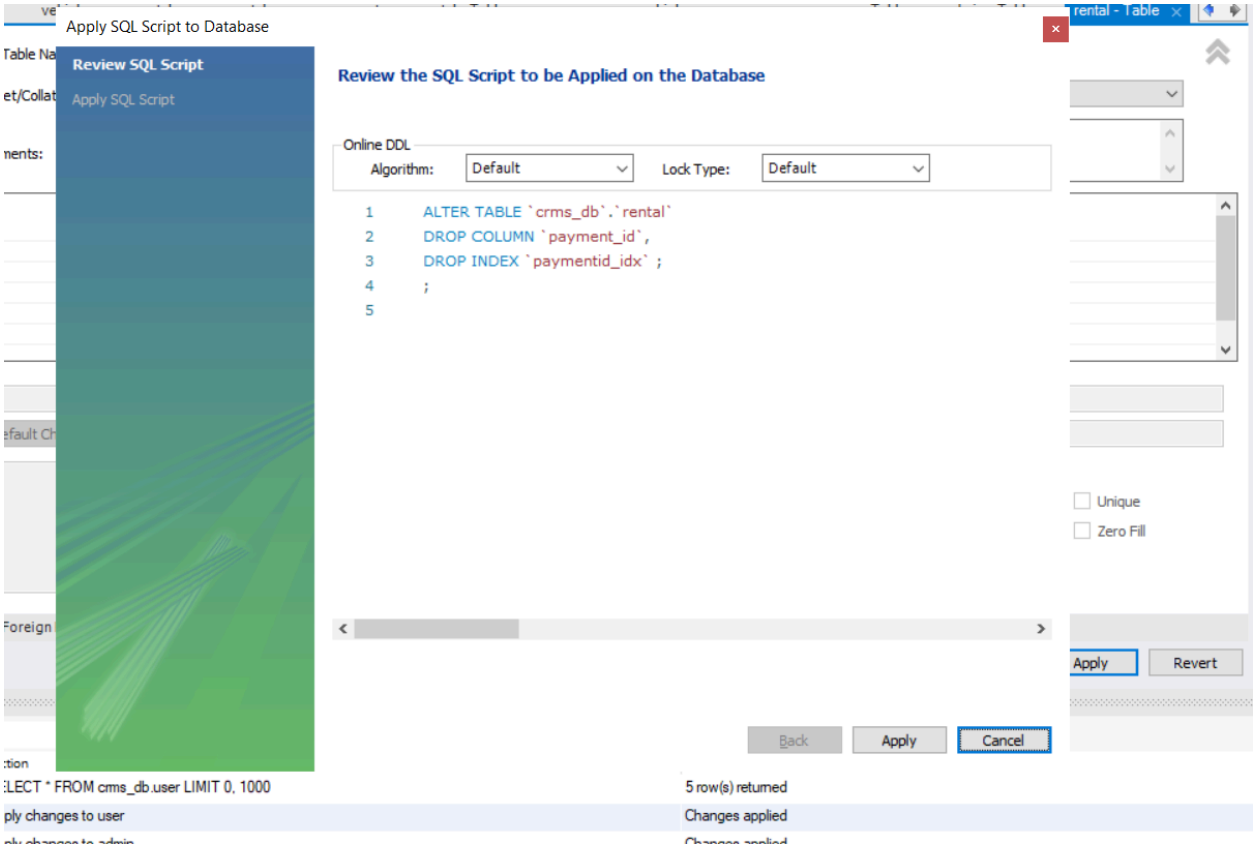

Apply SQL Script to Database
Review SQL Script
Apply SQL Script
Review the SQL Script to be Applied on the Database
1 UPDATE `crms_db`.`rental` SET `totalcost` = '5700' WHERE (`rentid` = '2001');
2
rental
Result Grid
Form Editor
Apply
Revert
Cont
Back
Apply
Cancel
changes to vehicle
Changes applied
T * FROM crms_db.admin LIMIT 0, 1000
5 row(s) returned

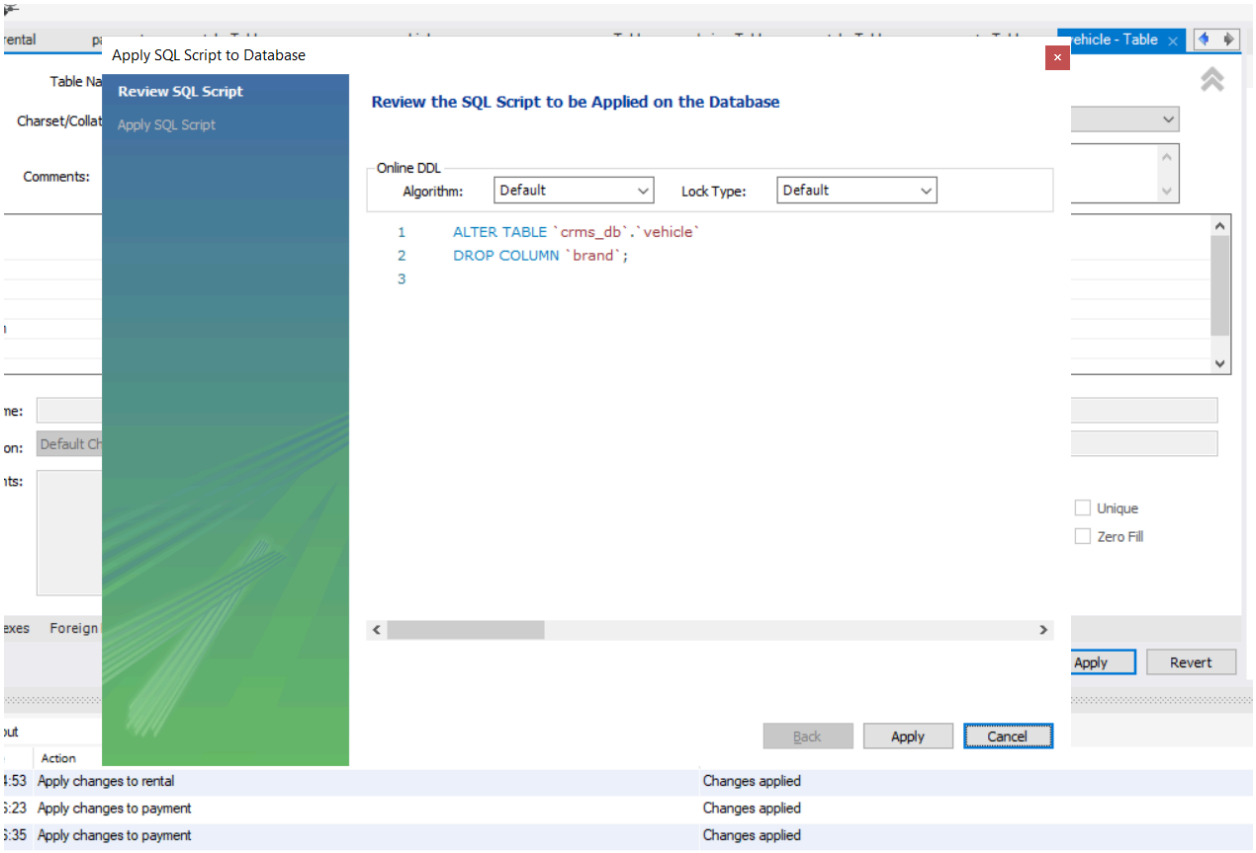

Apply SQL Script to Database
Review SQL Script
Apply SQL Script
Review the SQL Script to be Applied on the Database
1 UPDATE `crms_db`.`vehicle` SET `model` = 'Innova Crystaa' WHERE (`vehid` = '
2
vehicle
Result Grid
Form Editor
Apply
Revert
Cont
Back
Apply
Cancel
T * FROM crms_db.admin LIMIT 0, 1000
5 row(s) returned

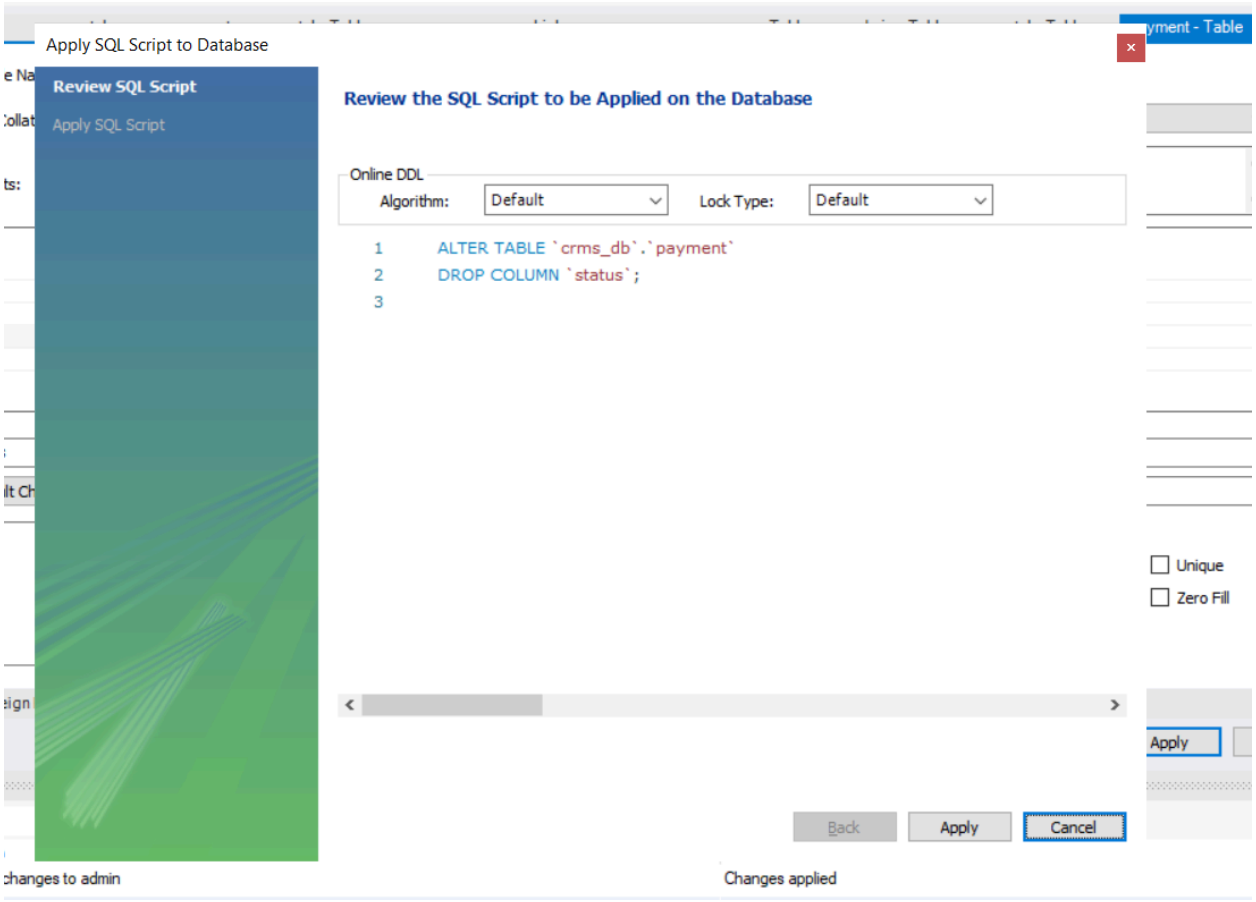
Delete:


Apply SQL Script to Database
Review SQL Script
Apply SQL Script
Review the SQL Script to be Applied on the Database
Online DDL
Algorithm:
Default
Lock Type:
Default
1 ALTER TABLE `crms_db`.`user`
2 DROP COLUMN `gender`;
3
Back
Apply
Cancel
user - Table
Unique
Zero Fill
Apply
Revert
Action
SELECT * FROM crms_db.user LIMIT 0, 1000
5 row(s) returned
SELECT * FROM crms_db.vehicle LIMIT 0, 1000
5 row(s) returned


admin
admin - Table
SQLAdditions
Apply SQL Script to Database
Review SQL Script
Apply SQL Script
Review the SQL Script to be Applied on the Database
Online DDL
Algorithm:
Default
Lock Type:
Default
1 ALTER TABLE `crms_db`.`admin`
2 DROP COLUMN `adminemail`;
3
Table Na
Charset/Collat
Comments:
Unique
Zero Fill
Foreign
Apply
Revert
Context Help
Back
Apply
Cancel
Action
:37 SELECT * FROM crms_db.vehicle LIMIT 0, 1000
5 row(s) returned
:10 SELECT * FROM crms_db.user LIMIT 0, 1000
5 row(s) returned
:59 Apply changes to user
Changes applied
:28 Apply changes to admin
Applying object changes ...


Apply SQL Script to Database
Review SQL Script
Apply SQL Script
Review the SQL Script to be Applied on the Database
Online DDL
Algorithm:
Default
Lock Type:
Default
1 ALTER TABLE `crms_db`.`rental`
2 DROP COLUMN `payment_id`,
3 DROP INDEX `paymentid_idx` ;
4 ;
5
Unique
Zero Fill
Apply
Revert
Back
Apply
Cancel
rental - Table
LECT * FROM crms_db.user LIMIT 0, 1000
5 row(s) returned
ply changes to user
Changes applied


Apply SQL Script to Database
Review SQL Script
Apply SQL Script
Review the SQL Script to be Applied on the Database
Online DDL
Algorithm:
Default
Lock Type:
Default
1 ALTER TABLE `crms_db`.`vehicle`
2 DROP COLUMN `brand`;
3
Unique
Zero Fill
Apply
Revert
Back
Apply
Cancel
Action
Apply changes to rental
Changes applied
Apply changes to payment
Changes applied
Apply changes to payment
Changes applied


Apply SQL Script to Database
Review SQL Script
Apply SQL Script
Review the SQL Script to be Applied on the Database
Online DDL
Algorithm:
Default
Lock Type:
Default
1 ALTER TABLE `crms_db`.`payment`
2 DROP COLUMN `status`;
3
Unique
Zero Fill
Apply
Back
Apply
Cancel
changes to admin
Changes applied