**Query 1**

Which is the most booked car ?

**SQL**

SELECT \* FROM (SELECT NAME,COUNT(\*) AS COUNT FROM VEHICLE NATURAL JOIN BOOKING GROUP BY NAME)AS Q4 NATURAL JOIN (SELECT MAX(COUNT) AS COUNT FROM (SELECT NAME,COUNT(\*) AS COUNT FROM VEHICLE NATURAL JOIN BOOKING GROUP BY NAME) AS Q1) AS Q2

**Query 2**

List customer name and amount who have booked swift desire & billing amount >=20000

**SQL**

SELECT CUSTOMER.NAME,B.TOTAL\_AMOUNT FROM ((BOOKING JOIN BILLING ON BOOKING.BOOK\_ID=BILLING.BOOK\_ID) AS A JOIN VEHICLE ON A.VEHICLE\_ID=VEHICLE.VEHICLE\_ID ) AS B JOIN CUSTOMER ON B.CUST\_ID=CUSTOMER.CUST\_ID WHERE TOTAL\_AMOUNT>=20000 AND B.NAME='Swift Desire'

**Query 3**

List all available vehicle for given date.

**SQL**

SELECT VEHICLE\_ID,NAME FROM VEHICLE NATURAL JOIN (SELECT VEHICLE\_ID FROM VEHICLE EXCEPT SELECT VEHICLE\_ID FROM BOOKING WHERE BOOKING.START\_TIME <= '09-15-2017' AND BOOKING.END\_TIME>='09-15-2017') AS V1

**Query 4**

List all available driver for given date

**SQL**

SELECT D\_ID,NAME FROM DRIVER NATURAL JOIN (SELECT D\_ID FROM DRIVER EXCEPT SELECT D\_ID FROM BOOKING WHERE BOOKING.START\_TIME <= '09-15-2017' AND BOOKING.END\_TIME>='09-15-2017') AS D1

**Query 5**

Avg driver rating of all drivers.(View)

**SQL**

CREATE VIEW AVG\_DRIVER\_RATING AS SELECT NAME,D\_ID,AVG(DRIVER\_RATING) FROM DRIVER NATURAL JOIN (BILLING JOIN BOOKING ON BILLING.BOOK\_ID=BOOKING.BOOK\_ID )GROUP BY D\_ID ORDER BY AVG DESC

**Query 6**

Avg vehicle rating for all vehicle(View)

**SQL**

CREATE VIEW AVG\_VEHICLE\_RATING AS SELECT NAME,VEHICLE\_ID,AVG(VEH\_RATING) FROM VEHICLE NATURAL JOIN(BILLING JOIN BOOKING ON BILLING.BOOK\_ID=BOOKING.BOOK\_ID )GROUP BY VEHICLE\_ID ORDER BY AVG DESC

**Query 7**

Give data for trip where driver has male gender

**SQL**

SELECT DISTINCT NAME,D\_ID FROM (BOOKING NATURAL JOIN DRIVER) WHERE GENDER='M' ORDER BY D\_ID;

**Query 8**

Give customer’s name and vehicle’s name of all female customers who have travelled more than 400 miles.

**SQL**

SELECT CUSTOMER.NAME AS CUST\_NAME,B.NAME AS CAR\_NAME,BOOK\_ID,F\_READING-INIT\_READING AS TOTAL\_TRAVEL FROM (((BILLING NATURAL JOIN BOOKING) AS A NATURAL JOIN VEHICLE ) AS B JOIN CUSTOMER ON B.CUST\_ID=CUSTOMER.CUST\_ID ) WHERE CUSTOMER.GENDER='F' AND F\_READING-INIT\_READING>400;

**Query 9**

Calculate average distance travelled by each vehicle.

**SQL**

SELECT VEHICLE\_ID,NAME,AVG FROM (SELECT VEHICLE\_ID,SUM/COUNT AS AVG FROM (SELECT VEHICLE\_ID,SUM(BILLING.F\_READING-BILLING.INIT\_READING),COUNT(BILLING.BOOK\_ID) FROM BILLING JOIN BOOKING ON BILLING.BOOK\_ID=BOOKING.BOOK\_ID GROUP BY VEHICLE\_ID)AS Q )AS Q2 NATURAL JOIN VEHICLE

**Query 10**

Give name of customer who have done intrastate journey.

**SQL**

SELECT NAME,B1.FROM\_CITY,B2.TO\_CITY FROM ((BOOKING JOIN CITY ON BOOKING.FROM\_CITY=CITY.C\_NAME) AS B1 JOIN (BOOKING JOIN CITY ON BOOKING.TO\_CITY=CITY.C\_NAME) AS B2 ON B1.BOOK\_ID=B2.BOOK\_ID) JOIN CUSTOMER ON B1.CUST\_ID=CUSTOMER.CUST\_ID WHERE B1.STATE=B2.STATE

**Query 11**

Calculate total miles driven by all drivers.

**SQL**

SELECT SUM(BILLING.F\_READING-BILLING.INIT\_READING) AS TOTAL\_MILES\_DRIVEN, DRIVER.NAME FROM BILLING JOIN BOOKING ON BILLING.BOOK\_ID=BOOKING.BOOK\_ID NATURAL JOIN DRIVER WHERE DRIVER\_REQUIREMENT = 'YES' GROUP BY NAME ORDER BY TOTAL\_MILES\_DRIVEN DESC;

**Query 12**

List top 5 customers with highest bill.

**SQL**

SELECT NAME, SUM(TOTAL\_AMOUNT) AS TOTAL\_BILL FROM BILLING JOIN BOOKING ON BILLING.BOOK\_ID=BOOKING.BOOK\_ID NATURAL JOIN CUSTOMER GROUP BY NAME ORDER BY TOTAL\_BILL DESC LIMIT 5

**Query 13**

Give customer name, vehicle name and book id for the scenario where total amount is equal to minimum amount for that booking period (where driver is not taken).

**SQL**

SELECT C.BOOK\_ID,C.NAME,C.VEHICLE\_NAME,MIN\_AMOUNT,REAL\_AMOUNT FROM(SELECT B.BOOK\_ID,B.NAME,B.VEHICLE\_NAME,REAL\_AMOUNT,(TOTAL\_DAYS\*COST\_PER\_DAY) AS MIN\_AMOUNT FROM (SELECT BOOKING.BOOK\_ID,(TOTAL\_AMOUNT-TAX\_AMOUNT) AS REAL\_AMOUNT,(END\_TIME-START\_TIME+1) AS TOTAL\_DAYS,VEHICLE\_ID,COST\_PER\_DAY,CUSTOMER.NAME,VEHICLE.NAME AS VEHICLE\_NAME FROM (BILLING JOIN BOOKING ON BILLING.BOOK\_ID=BOOKING.BOOK\_ID NATURAL JOIN VEHICLE ) JOIN CUSTOMER ON BOOKING.CUST\_ID=CUSTOMER.CUST\_ID)AS B ) AS C WHERE C.MIN\_AMOUNT=C.REAL\_AMOUNT

**Query 14**

Make a view of final amount that is calculated by considering advance amount (for all the bills).

**SQL**

CREATE VIEW FINAL\_BILL\_RECEIPT AS SELECT NAME, BILL\_ID, (TOTAL\_AMOUNT - ADVANCE\_AMOUNT)AS FINAL\_PAYABLE\_AMOUNT FROM BILLING JOIN BOOKING ON BILLING.BOOK\_ID=BOOKING.BOOK\_ID NATURAL JOIN CUSTOMER