PROJECT REPORT
CAR RENTAL SYSTEM

SUBMITTED TO:-

Mr. Deepak Kumar Verma

M.Tech. (Computer Science and Engineering)
Assistant Professor, Department of Computer Science and Engineering and Technology, C S J M University Kanpur

TEAM MEMBERS:-

- **RAHUL SHARMA (ROLL NO: - 25)**
- **AVINASH TRIPATHI (ROLL NO: - 45)**

UIET CSJMU KANPUR

ABOUT

1) INTRODUCTION:-

We have selected CAR RENTAL SYSTEM as our DBMS Project. Customers can select the cars for rent on the basis of car-type (i.e. Economy, Standard, Premium, and SUV). Our system provides facility for customers to book the car that is available in a particular Car Rental Agency by just providing their details. In our project we have considered only the booking date and returning date, hence we have not included the concept of "Late Fees". The customers have to register on the web page to book any car; the registration includes their License details, the car which they want to book and many more. The customers would get an automatic generated bill on returning the car with the amount charged for car and 10% tax included.

2) REQUIREMENTS:-

a)
Each car present in Car Agency must have an individual   Car Id and must belong to any one of the car type.

b)   Customers have choice to select any car they are interested in, if it is available.

c) At any instance of time numbers of cars must not be equal to ZERO.

d) Bill should be automatically generated as soon as customer submits the car.

e) Auto decrement and increment in the stock of cars must be done according to the booking and returning of any car.

f) Details submitted by customer must be valid and age should be checked as priority (greater than 18).
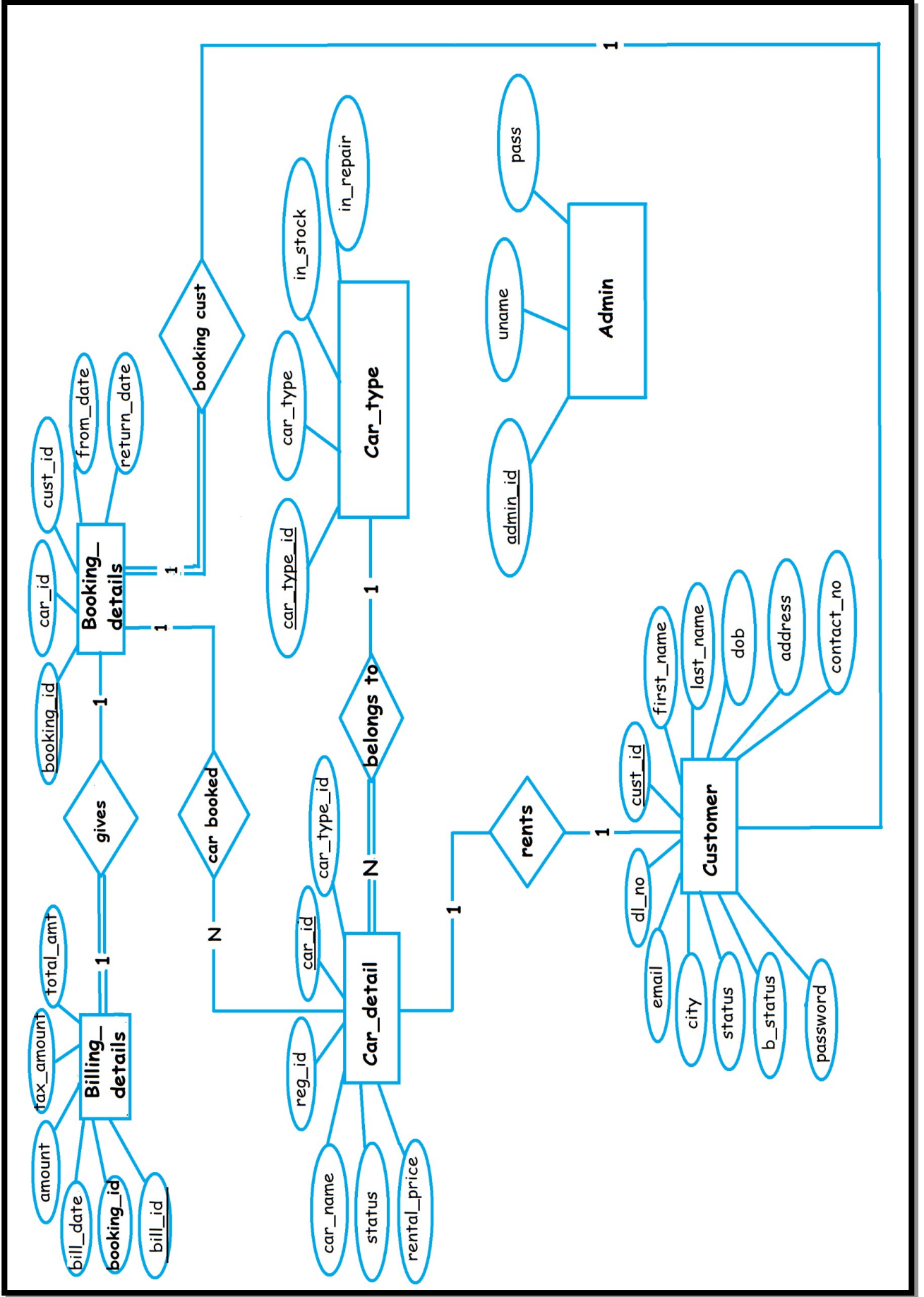
An Entity-Relationship (ER) diagram for a car rental system showing the following entities and relationships:

Entities and attributes:

- **Booking_details**: booking_id, car_id, cust_id, from_date, return_date
- **Billing_details**: bill_id, booking_id, bill_date, amount, tax_amount, total_amt
- **Car_type**: car_type_id, car_type, in_stock, in_repair
- **Car_detail**: reg_id, car_id, car_type_id, car_name, status, rental_price
- **Admin**: admin_id, uname, pass
- **Customer**: cust_id, first_name, last_name, dob, address, contact_no, dl_no, email, city, status, b_status, password

Relationships:

- Booking_details — gives (1:1) — Billing_details
- Booking_details — car booked (1:N) — Car_detail
- Booking_details — booking cust (1:1) — Customer
- Car_detail — belongs to (N:1) — Car_type
- Customer — rents (1:1) — Car_detail

Table Schemas

Our project consists of database with 6 tables. Each table with details of various fields like customer details, car type details, booking details, billing details etc.

Below we give the schema of individual table:-

1) billing_details:-

#	Name	Type	Attributes	Null	Default	Extra
1	Bill_id	int(10)		no	none	auto_increment
2	booking_id	int(20)		no	none	
3	bill_date	date		no	none	
4	amount	decimal(10,2)		no	none	
5	tax_amount	decimal(10,2)		no	none	
6	total_amount	decimal(10,2)		no	none	

DDL:-

```
create table Billing_details
(
bill_id integer (10) auto_increment primary key,
```

booking_id integer (20) not null references Booking_details (Booking_id),
Bill_date date not null,
amount decimal (10, 2) not null,
tax_amount decimal (10, 2) not null,
total_amount decimal(10,2) not null
);

2) booking_details:-

#	Name	Type	Attributes	Null	Default	Extra
1	booking_id	int(20)		no	none	auto_increment
2	car_id	int(11)		no	none	
3	cust_id	int(11)		no	none	
4	from_date	date		no	none	
5	return_date	date		no	none	

DDL:-

create table booking_details
(
booking_id integer(20) auto_increment primary key,
car_id integer(11) not null references car_detail(car_id),
cust_id integer(11) not null references customer(cust_id),
from_date date not null,
return_date date not null check(return_date>from_date)
);

3) car_detail:-

#	Name	Type	Attributes	Null	Default	Extra
1	car_id	int(11)		no	none	auto_increment
2	car_type_id	int(11)		no	none	
3	reg_id	varchar(255)		no	none	
4	car_name	varchar(255)		no	none	
5	image	text		no	none	
6	rental_price	int(11)		no	none	
7	status	varchar(255)				

DDL:-

```
create table car_detail
(
car_id integer(11) auto_increment primary key,
reg_id varchar(255) not null,
car_name varchar(255) not null,
image text not null,
rental_price integer(11) not null,
status varchar(255) not null
);
```

4) car_type:-

#	Name	Type	Attributes	Null	Default	Extra
1	car_type_id	int(11)		no	none	auto_increment

#	Name	Type	Attributes	Null	Default	Extra
2	car_type_name	varchar(50)		no	none	
3	in_stock	integer(11)		no	none	
4	in_repair	integer(11)		no	none	

DDL:-

```
create table car_type
( car_type_id integer(11) auto_increment primary key,
car_type_name varchar(50) not null,
in_stock integer(11) not null,
in_repair integer(11) not null
);
```

5)   customer:-

#	Name	Type	Attributes	Null	Default	Extra
1	cust_id	int(11)		no	none	auto_increment
2	dl_no	varchar(10)		no	none	
3	first_name	varchar(10)		no	none	
4	last_name	varchar(10)		no	none	
5	dob	date		no	none	
6	address	varchar(255)		no	none	
7	city	varchar(255)		no	none	
8	state	varchar(255)		no	none	
9	contact_no	char(10)		no	none	
10	password	varchar(255)		no	none	
11	email_id	varchar(50)		no	none	
12	b_status	int(1)		no	none	

DDL:-

```sql
create table customer
( cust_id integer(11) auto_increment primary key,
  dl_no varchar(10) not null,
  first_name varchar(10) not null,
  last_name varchar(10) not null,
  dob date not null,
  address varchar(255) not null,
  city varchar(255) not null,
  state varchar(255) not null,
  contact_no char(10) not null,
  password varchar(255) not null,
  email_id varchar(255) not null,
  b_status integer(1)
);
```

6) admin:-

#	Name	Type	Attributes	Null	Default	Extra
1	admin_id	int(11)		no	none	auto_increment
2	uname	varchar(50)		no	none	
3	pass	varchar(50)		no	none	

DDL:-

```sql
create table admin
( admin_id integer(11) auto_increment primary key,
  uname varchar(50) not null,
  pass varchar (50) not null
);
```

Triggers implemented:-

Name	Table	Action			Time	Event
auto_bill_generate	booking_details	✎ Edit	🗐 Export	⊘ Drop	AFTER	UPDATE
decrement_in_stock	booking_details	✎ Edit	🗐 Export	⊘ Drop	AFTER	INSERT
increment_in_stock	billing_details	✎ Edit	🗐 Export	⊘ Drop	AFTER	INSERT
set_unavailable	booking_details	✎ Edit	🗐 Export	⊘ Drop	AFTER	INSERT

We have implemented 4 triggers in our database. These triggers were very beneficial for making our rental system more efficient and effective.

1) auto_bill_generate trigger:-

This trigger generates the bill for taking the car on rent automatically as soon as customer enters the return_date on which he/she is returning the car to the agency. Customer initially fills only the from_date and other details while booking.

Details

Trigger name	auto_bill_generate
Table	booking_details
Time	AFTER
Event	UPDATE

Definition

```
1  INSERT INTO billing_details
2  SET booking_id=old.booking_id,
3  bill_date=new.return_date,
4  amount=(datediff(new.return_date, old.from_date))*(SELECT rental_price
   FROM car_detail WHERE car_id=old.car_id),
5  tax_amount=(amount * 0.1),
6  total_amount=amount+tax_amount
```

Definer   root@localhost

2) **decrement_in_stock:-**
This trigger decrements the no of cars in current stock by 1 as soon as the customer books any one the car. In the car_type table the in_stock attribute automatically gets decremented as the car is booked, showing that now the car is unavailable in agency.

Details

Trigger name	decrement_in_stock
Table	booking_details
Time	AFTER
Event	INSERT

Definition

```
1  UPDATE car_type
2  SET in_stock=(in_stock)-1
3  WHERE car_type_id=(SELECT car_type_id FROM car_detail, booking_details
   WHERE car_detail.car_id=booking_details.car_id AND
   booking_id=new.booking_id)
```

Definer	root@localhost

Go   Close

3) set_unavailable:-
This trigger sets the status of the car to be unavailable to be booked. This works after the car has been booked by any customer and hence is currently not in the agency. The car would now not be displayed on the car list page.

Edit trigger

Details

Trigger name	set_unavailable
Table	booking_details
Time	AFTER
Event	INSERT

Definition

```
1 UPDATE car_detail
2 SET car_detail.status= 'Unavailable'
3 WHERE car_id=(SELECT car_id FROM booking_details WHERE
  booking_id=new.booking_id)
```

Definer root@localhost

Go   Close

4) auto_increment:-

This trigger is just opposite of auto_decrement trigger. It incerements the stock by 1 when the customer returns the car in agency. The in_stock attribute is incremented as soon as car is returned back i.e. now the car is available for another booking and can be seen in the car list page.

Edit trigger

Details

Trigger name	increment_in_stock
Table	billing_details
Time	AFTER
Event	INSERT

Definition

```sql
1  UPDATE car_type
2  SET in_stock=(in_stock)+1
3  WHERE car_type_id=(SELECT car_type_id FROM car_detail, booking_details
   WHERE car_detail.car_id=booking_details.car_id AND
   booking_id=new.booking_id)
```

Definer root@localhost

Go   Close

- <u>Purpose and Functionality:-</u>

The home page consists of option to REGISTER or USER LOGIN for either creating a new account or logining into an old one. The top of the page contains links to Home Page, Vehicle information and about us. The page also consists of list of available cars according to their types.

On clicking on Register button, a new page opens with various fields for data to be entered by a new user. The user has to enter his complete information in order to take any car for rent. On clicking Submit button, a unique Cust_id is generated which he has to use further to book any car. He is requested to remember the generated cust_id.

Vehicle Rental System	User Registration Page	

Driving licence Number

First Name

Last Name

Date of Birth(yyyy-mm-dd)

Address

City

State

Contact Number

Choose Your Password

Enter email

SUBMIT

Whereas the Login button asks customer for his/her mobile number and password to log in.

After registering/logining only the customer would be able to book any car.

Vehicle Rental System	User Login Page	

contact number

password

login

The Vehicle and Pricing info button opens up a page where a search button is provided through which customer could search for the type of car he wants to book. And on

selecting the type he/she would get the list of all cars present in the stock currently from which he can choose any one of them.

Vehicle Rental System   Vehicles & Pricing Info

Select Car type :   Economy ⌄

Search

List of Available cars (click for booking.)

₹ 6000

₹ 5000

Car Name : Scorpio

Car ID: 5

Car Type: SUV

Car Name : Renault duster

Car ID: 6

Car Type: SUV

Admin login page asks for the username and password for log in. Admin can enter by giving details and could make changes to database.

Vehicle Rental System   Admin login page

username

password

login

B.Tech Project ©2017 DBMS

Customer profile page displays the list of car he has booked on the home page. He can choose the options from left navigation bar to update his information book a new car or return the car that he has booked.

Hello rahul

Welcome rahul

Please select options from left navigation Panel.

Home

Update Info

Book a car

Return car

Booking history

Booking ID	Car ID	CUST ID	FROM DATE	Return DATE
6	2	8	2017-11-20	2017-11-23

B.Tech Project ©2017 DBMS

Hello User

Update profile info...

Home

Update Info

Rent a car

Return car

Enter cust ID:

Edit Last name:

Edit address:

Edit city:

Edit state:

Edit contact no:

Edit password:

Update   please enter details and click submit.

Error:
You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near ' address=, city=, state=, contact_no=, password= WHERE cust_id='" at line 1

B.Tech Project ©2017 DBMS

localhost/project/profileupdate.php

Return car...

Enter booking ID :

Enter car ID :

Enter Return date:

Return Car

Bill ID	Booking ID	Bill date	Amount	Tax Amount	Total Amount

Home

Update Info

Rent a car

Return car

B.Tech Project ©2017 DBMS

Admin panel has the full control over the database. The navigation bar is given on the left side from which user can select the various options. He can add a new car, update the car info. Also he can view various reports that can be generated by the database. The reports are given in the dropdown list of the heading REPORTS.

Welcome Admin...

Please select options from left navigation Panel.

B.Tech Project ©2017 DBMS

Admin Panel

Home

Admin Home

Add Car

Update car info

Reports

View All Customers

View All Cars

View Booking details

Revenue details (by Car ID)

Revenue details (by Car types)

Stock details

For adding a new car, admin has to give the Registeration_id, name, type, image, price and status of the car. The car would be added and displayed as available on the car list page.

Admin Panel

Welcome admin...

Registration ID

Car Name

CAr type

full name of img file

rental price

Status

SUBMIT   please enter details and click submit.

Error:
Unknown column 'car_type' in 'field list'

Home

Admin Home

Add Car

Update car info

Reports

View All
Customers

View All Cars

View Booking
details

Revenue
details (by
Car ID)

Revenue
details (by

B.Tech Project ©2017 DBMS

For updating renting price of any car info, admin has to select the car type, enter the car_id and new price.

Admin Panel

Welcome admin...

Select Car type to update: Economy ⌄

Enter car ID:

New Daily Rate:

Update   please enter details and click submit.

Error:
You have an error in your SQL syntax; check the manual that corresponds to your
MySQL server version for the right syntax to use near 'WHERE car_id='' at line 1

Home

Admin Home

Add Car

Update car info

Reports

View All
Customers

View All Cars

View Booking
details

Revenue
details (by
Car ID)

Revenue
details (by

B.Tech Project ©2017 DBMS

•Reports

1) View all customers:-

This report generates records of all customers that are registered in the agency.

<u>SQL:-</u>

select * from customers;

2) View all cars:-
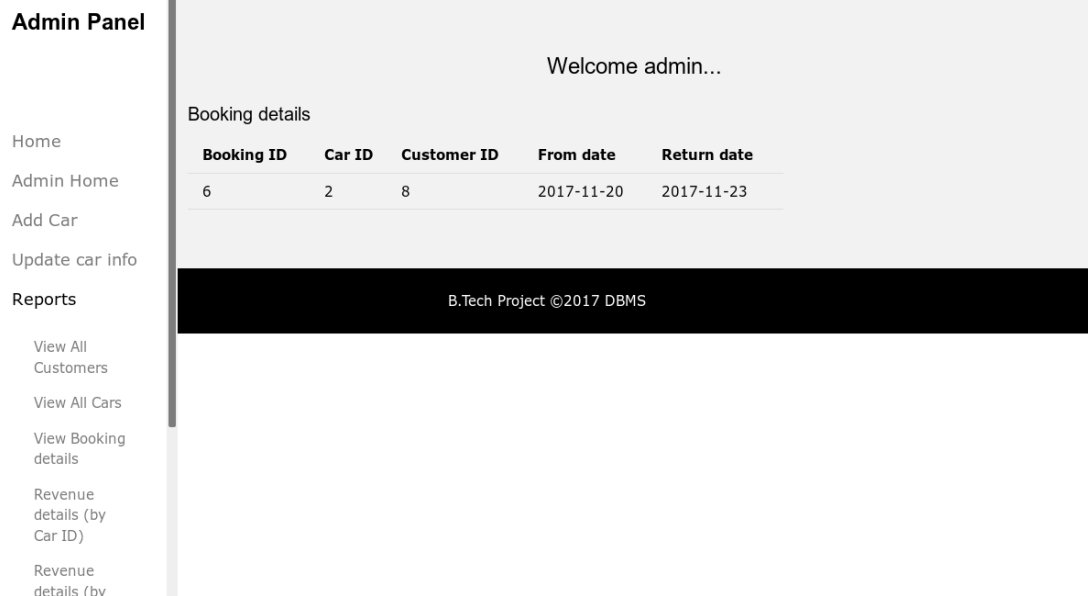
Admin can view all the cars in the agency, their details and status of availability.

SQL:-

select * from car_detail;

3) View booking details:-

Admin can see the list of cars that are booked at any instance. The cust_id gives the customer details and car_id provides the car details that have been booked.

SQL:-

select * from booking_details;

4) View bills report(by car_id):-

SQL:-

select * from billing_details,booking_details where billing_details.booking_id=booking_details.booking_id and car_id=2;

Admin can view the bills generated at the time when customer has returned the car. The bill is generated with the car_id, cust_id, booking_id, and the total amount calculated after adding the 10% tax on car.

5) **View bill reports(by car_type):-**

SQL:-
select * from
billing_details,booking_details,car_detail,car_type
where

billing_details.booking_id=booking_details.booking_id
and booking_details.car_id=car_detail.car_id and
car_detail.car_id=car_type.car_id and
car_type.car_type_id=1;

Admin has the option to view the bill reports on the basis of car_type i.e. which car type is booked.

6) **Stock details report:-**

Admin can view the current stock of cars. He can get the complete knowledge of the number of cars present in the agency and the number of cars went for repair. Each car is listed in this list either in stock or in repair. SQL:-

select * from car_type;

7) **Customer booked any car Report:-**

Admin Panel

Home

Admin Home

Add Car

Update car info

Reports

   View All
Customers

   View All Cars

   View Booking
details

   Revenue
details (by
Car ID)

   Revenue
details (by

Welcome admin...

List of all those customers who have booked any Car

Cust ID	DL NO.	First name	Last name	Booking ID	Car ID
8	UPIN12345	rahul	sharma	6	2

B.Tech Project ©2017 DBMS

Admin can list out the customers that have booked any car in present time. All those who have registered but not booked are not present in this list.

SQL:-

select * from booking_details,customer where booking_details.cust_id=customer.cust_id;

8) Customers not booked any car Report:-

Admin Panel

Home

Admin Home

Add Car

Update car info

Reports

   View All
Customers

   View All Cars

   View Booking
details

   Revenue
details (by
Car ID)

   Revenue
details (by

Welcome admin...

List of all those customers who have not booked any Car

Cust ID	DL NO.	First name	Last name	Contact no
9	UPIN11111	avinash	tripathi	8765432100
10	58323265	Amitabh	kumar	8532826584

B.Tech Project ©2017 DBMS

The customers who have registered in the agency but not have booked any car are listed in this report. Admin must have record of these customers also.

SQL:-

```
select * from booking_details,customer where
booking_details.cust_id<>customer.cust_id;
```

9) Revenue by dates Reports:-

Revenue collected by the rents of the cars can be viewed from one date to another. Admin can give the dates from which he wants to check for the revenue up to the dates he is interested in. Total revenue could be judged by this report.

SQL:-

```
select * from billing_details where bill_date between
"2017-10-06" and "2017-11-25";
```

10) Booking between dates Report:-

Admin can have a look upon the car that is booked between given dates. The details of the cars not present in the agency for the particular time can also be got from the report.

Welcome admin...

from date

to date

SUBMIT

Booking ID	Car ID	Cust ID	From date
6	2	8	2017-11-20

B.Tech Project ©2017 DBMS

SQL:-

select * from booking_details where from_date
between "2017-10-20" and "2017-11-25";

11) Car been booked the most report:-

Welcome admin...

Car ID	Booking Count
2	1

B.Tech Project ©2017 DBMS

Admin can view the car that has been booked the
maximum no. of times. Also the decreasing order of the
booking frequency of the particular car.

SQL:-

select car_id,count(*) from booking_details group by car_id order by 2 desc;

REFERENECES

- **Database System Concepts by Silberschatz Korth Sudarshan.**
- **W3Schools website(www.w3schools.com)**
- **internet**