DBMS ASSIGNMENT 3

car rental management

Submitted by : Gaurav J PES1UG19CS599
Gokul Karthik PES1UG19CS170
Gurukiran S PES1UG19CS173

1) Complete Working model of the Database application

Simple Queries

```
-- CUSTOMER VIEW

-- 1) list of all outlets - select * from outlet;s1

select outlet_name,outlet_location
from outlet;
```

```
   outlet_name    | outlet_location
------------------+-----------------
 Vinayaka outlets | bangalore
 narayan outlets  | chitradurga
(2 rows)
```

```
-- 2) list of available vehicles - ASSUME checking in outlet_id=2
-- LATER can be passed parameter in function
select * from vehicle
where vehicleStatus='not-taken' and outlet_id=2;
```

```
car_rental_management=# select * from vehicle where vehicleStatus='not-taken' and outlet_id=2;
     plate_number     | outlet_id | model | number_of_seats | ac | vehiclestatus | emp_id
----------------------+-----------+-------+-----------------+----+---------------+--------
 KA-18-1221           |         2 | swift |               5 | t  | not-taken     |      2
(1 row)
```

```
-- 3) list of reservations made by customer - ASSUME customer_id=1
-- LATER can be passed as a parameter
select * from reservation where customer_id=1;
```

```
 reservation_id | reservation_date | vehicle_taken_date | expected_return_dat
e | customer_id | outlet_id | advance | emp_id |       plt_num        |  rese
rvation_status
----------------+------------------+--------------------+--------------------
--+-------------+-----------+---------+--------+----------------------+------
----------------
              1 | 2021-01-11       | 2021-01-12         | 2021-01-19
  |           1 |         1 |    1000 |      1 | DA-12-2192           | inpro
gress
              2 | 2021-01-23       | 2021-01-25         | 2021-02-01
  |           1 |         2 |    1000 |      2 | KA-18-1221           | cance
l
(2 rows)
```

```
-- 4) when the customer selects the vehicle then the status of the vehicle is set to to selected

-- to show list of not-taken
select * from vehicle where vehicleStatus='not-taken';
```

```
car_rental_management=# select * from vehicle where vehicleStatus='not-taken' and outlet_id=2;
     plate_number     | outlet_id | model | number_of_seats | ac | vehiclestatus | emp_id
----------------------+-----------+-------+-----------------+----+---------------+--------
 KA-18-1221           |         2 | swift |               5 | t  | not-taken     |      2
(1 row)
```

```
update vehicle
set vehicleStatus='selected'
where plate_number='KA-18-1221' and vehicleStatus='not-taken';
```

```
car_rental_management=# update vehicle set vehicleStatus='selected'
where plate_number='KA-18-1221' and vehicleStatus='not-taken';
UPDATE 0
car_rental_management=#
```

```
-- to show update
select * from vehicle where plate_number='KA-18-1221';
```

```
     plate_number     | outlet_id | model | number_of_seats | ac | vehiclestatus | emp_id
----------------------+-----------+-------+-----------------+----+---------------+--------
 KA-18-1221           |         2 | swift |               5 | t  | selected      |      2
(1 row)
```

```
-- CAN DO concurrency control here

-- 5) getting outlet contact

select outlet_mail,outlet_phone from outlet_contact where outlet_id=1
```

```
car_rental_management=# select outlet_mail,outlet_phone from outlet_contact
car_rental_management-# where outlet_id=1;
 outlet_mail | outlet_phone
-------------+--------------
             | 2190301923
(1 row)
```

-- EMPLOYEE VIEW

```
-- 1) list of all vehicles in that outlet -
select * from vehicle where outlet_id=1;
```

```
    plate_number     | outlet_id | model | number_of_seats | ac | vehiclestatus | emp_id
---------------------+-----------+-------+-----------------+----+---------------+--------
 DA-12-2192          |         1 | omni  |               6 | t  | taken         |      1
 DW-12-2192          |         1 | omni  |               6 | t  | selected      |      1
 DA-12-2193          |         1 | benz  |               3 | t  | not-taken     |      1
(3 rows)
```

```
-- 2) list of all reservations made my all customers in that outlet,
-- to see if to approve a customer coming to outlet on pickup date
select * from reservation where outlet_id=1 and reservation_status='inprogress';
```

```
 reservation_id | reservation_date | vehicle_taken_date | expected_return_date
| customer_id | outlet_id | advance | emp_id |       plt_num        |  reservat
ion_status
----------------+------------------+--------------------+----------------------
+-------------+-----------+---------+--------+----------------------+----------
------------
              1 | 2021-01-11       | 2021-01-12         | 2021-01-19
|           1 |         1 |    1000 |      1 | DA-12-2192           | inprogres
s
(1 row)
```

```
-- 3) update rent table, where reservations whose approves is inprogress
-- set the reservation status to returned, and update the rent

-- first set return_date(REAL) in rent, when customer comes back
update rent
set return_date='20-01-2021',number_of_days=return_date-
taken_date,total_amount=1000*number_of_days+tax_amount
where reservation_id=1;
```

```
car_rental_management=# update rent
car_rental_management-# set return_date='20-01-2021',number_of_days=return_date
-taken_date,total_amount=1000*number_of_days+tax_amount
car_rental_management-# where reservation_id=1;
UPDATE 1
```

```
select * from rent where reservation_id=1;
```

```
 bill_id | taken_date | return_date | number_of_days | tax_amount | total_amoun
t | customer_id | reservation_id | refund |       plt_num
---------+------------+-------------+----------------+------------+------------
--+-------------+----------------+--------+----------------------
       1 | 2021-01-12 | 2021-01-20  |              7 |        100 |         710
0 |           1 |              1 |      0 | DA-12-2192
(1 row)
```

```
-- now in reservation go and set status to returned + vehicle status to 'not-taken'

update reservation
set reservation_status='completed'
where reservation_id=1;
```

```
car_rental_management=# update reservation
car_rental_management-# set reservation_status='completed'
car_rental_management-# where reservation_id=1;
UPDATE 1
car_rental_management=#
```

```
select * from reservation where reservation_id=1;
```

```
 reservation_id | reservation_date | vehicle_taken_date | expected_return_date
| customer_id | outlet_id | advance | emp_id |       plt_num        |  reservat
ion_status
----------------+------------------+--------------------+----------------------
+-------------+-----------+---------+--------+----------------------+----------
------------
              1 | 2021-01-11       | 2021-01-12         | 2021-01-19
|           1 |         1 |    1000 |      1 | DA-12-2192           | completed

(1 row)
```

```
update Vehicle
set vehicleStatus='not-taken'
where plate_number = (SELECT plt_num from reservation where reservation_id=1);
```

```
car_rental_management=# update Vehicle
car_rental_management-# set vehicleStatus='not-taken'
car_rental_management-# where plate_number = (SELECT plt_num from reservation w
here reservation_id=1);
UPDATE 1
car_rental_management=#
```

```
SELECT * from vehicle where plate_number = (SELECT plt_num from reservation where
reservation_id=1);
```

```
     plate_number     | outlet_id | model | number_of_seats | ac | vehiclestatu
s | emp_id
----------------------+-----------+-------+-----------------+----+------------
--+--------
 DA-12-2192           |         1 | omni  |               6 | t  | not-taken
  |      1
(1 row)
```

```
-- 4) IF customer shows usp on pickup date in reservation
-- insert into rent table

-- dummy to show
INSERT into reservation values(3, '21-04-2021', '28-04-2021','01-05-2021',2,1,1000,1,'DW-12-2192','inprogress');
```

```
car_rental_management=# INSERT into reservation values(3, '21-04-2021', '28-04-2021','01-05-2021',2,1,1000,1,'DW-12-2192','inprogress');
INSERT 0 1
car_rental_management=#
```

```
insert into rent(taken_date, return_date ,number_of_days ,total_amount,
customer_id ,reservation_id ,refund ,plt_num )
select vehicle_taken_date,expected_return_date,expected_return_date-vehicle_taken_date,1000(expected_return_date-vehicle_taken_date),
customer_id,reservation_id ,advance*0.1 , plt_num
from reservation
where reservation_id=3;
select * from rent where reservation_id=3;
```

```
INSERT 0 1
 bill_id | taken_date | return_date | number_of_days | tax_amount | total_amount | customer_id | reservation_id | refund |       plt_num
---------+------------+-------------+----------------+------------+--------------+-------------+----------------+--------+---------------------
       3 | 2021-04-28 | 2021-05-01  |              3 |         10 |         3000 |           2 |              3 |    100 | DW-12-2192
(1 row)
```

```
-- as he showed up update vehicle status to taken

update Vehicle
set vehicleStatus='taken'
where plate_number = (SELECT plt_num from reservation where reservation_id=3);

SELECT * from vehicle where plate_number = (SELECT plt_num from reservation where reservation_id=3);
```

```
UPDATE 1
     plate_number     | outlet_id | model | number_of_seats | ac | vehiclestatus
----------------------+-----------+-------+-----------------+----+---------------
 DW-12-2192           |         1 | omni  |               6 | t  | taken
(1 row)
```

-- MANAGER/ADMIN VIEW

--1) view current savings of an outlet (after adding base + income of any rent(complex query)) -
```
select outlet_savings from outlet where outlet_id=1;
```

```
 outlet_savings
----------------
          10000
(1 row)
```

--2) list of all vehicles in that outlet -
```
select * from vehicle where outlet_id=1;
```

```
     plate_number     | outlet_id | model | number_of_seats | ac | vehiclestatus
----------------------+-----------+-------+-----------------+----+---------------
 DA-12-2192           |         1 | omni  |               6 | t  | not-taken
 DW-12-2192           |         1 | omni  |               6 | t  | taken
(2 rows)
```

--3) orderby(number of seats, outlet) all vehicles in db

```
select number_of_seats, plate_number, model,outlet_id
from vehicle
order by number_of_seats;
```

```
 number_of_seats |     plate_number     | model | outlet_id
-----------------+----------------------+-------+-----------
               5 | KA-18-1221           | swift |         2
               6 | DA-12-2192           | omni  |         1
               6 | DW-12-2192           | omni  |         1
(3 rows)
```

--4) update outlet contact

```
update outlet_contact
set outlet_mail='ilikeWeed@cubahavana',outlet_phone='0212112123'
where outlet_id=1;

select * from outlet_contact where outlet_id=1
```

```
UPDATE 1
 outlet_id | outlet_phone |     outlet_mail
-----------+--------------+---------------------
         1 | 0212112123   | ilikeWeed@cubahavana
(1 row)
```

Complex and Nested Queries

```
\c car_rental_management

--1) update income of outlet from reservation completed ;

-- Assume outlet id is passed as parameter


update outlet
set outlet_savings = outlet_savings+(
select sum(total_amount) from rent
where outlet_id=1 and reservation_id in (
select reservation_id
from reservation
where reservation_status='completed'
)
)
where outlet_id=1;


select * from outlet;
```

```
 outlet_id |   outlet_name    | outlet_location | outlet_savings
-----------+------------------+-----------------+----------------
         1 | Vinayaka outlets | bangalore       |          10000
         2 | narayan outlets  | chitradurga     |          20000
(2 rows)
```

```
--2) getting final payment amount bill by joining the rent and got_discount;

-- DID NOT UPDATE THE TOTAL AMOUNT INSTEAD JUST RETURNED THE AMOUNT

-- for bill_id=1


insert into got_discount values('JollySeason',1);
SELECT * from got_discount;
```

```
INSERT 0 1
           disc_id            | bill_id
------------------------------+---------
 AugustSeason                 |       1
 JollySeason                  |       2
 JollySeason                  |       1
(3 rows)
```

```sql
select total_amount+tax_amount - (
select sum(discount_amount)
from discount
where promo_id in (
select disc_id
from got_discount
where bill_id=1
)
) as payable
from rent
where bill_id = 1;
```

```
 payable
---------
    2601
(1 row)
```

```sql
--3) finding the customer with maximum rent amount using max aggrigation group by outlet;

-- reservation is based on outlet wise, using resererervation fk, get the max amount through rent table

select * from rent;
select * from reservation;

select foo.outlet_id, rent.customer_id, foo.max
from (
select outlet_id,max(total_amount)
from reservation join rent on rent.reservation_id=reservation.reservation_id
group by outlet_id
) as foo join rent on rent.total_amount=foo.max;
```

```
 bill_id | taken_date | return_date | number_of_days | tax_amount | total_amount | customer_id | reservation_id | refund |       plt_num
---------+------------+-------------+----------------+------------+--------------+-------------+----------------+--------+---------------------
       1 | 2021-01-12 | 2021-01-19  |              7 |        100 |         3000 |           1 |              1 |      0 | DA-12-2192
       2 | 2021-01-12 | 2021-01-19  |              7 |        100 |         3000 |           1 |              2 |      0 | KA-18-1221
(2 rows)
```

```
 outlet_id | customer_id | max
-----------+-------------+------
         1 |           1 | 3000
         2 |           1 | 3000
         1 |           1 | 3000
         2 |           1 | 3000
(4 rows)
```

```
-- 4)finding the customer with most number of reservations;

select customer_id,count(*)
from reservation
group by customer_id;
```

```
 customer_id | count
-------------+-------
           1 |     2
(1 row)
```

```
-- 5)selecting customer who has made reservation for all cars of an outlets;
-- for all , using double negation
-- assume outlet 2
-- to check
select plate_number,outlet_id
from vehicle;
```

```
     plate_number     | outlet_id
----------------------+-----------
 DA-12-2192           |         1
 KA-18-1221           |         2
 DW-12-2192           |         1
(3 rows)
```

```
select customer_id,plt_num,outlet_id
from reservation;
```

```
 customer_id |       plt_num        | outlet_id
-------------+----------------------+-----------
           1 | DA-12-2192           |         1
           1 | KA-18-1221           |         2
(2 rows)
```

```
select customer_id, licence_number
from customer
where NOT EXISTS (
(
Select plate_number
from vehicle
where outlet_id=2
)
EXCEPT
(
SELECT plt_num
from reservation
where customer.customer_id=reservation.customer_id
)
);
```

```
 customer_id | licence_number
-------------+----------------
           1 | DL1234
(1 row)
```

Multiple users with different access privilege levels for different parts of the database should be created.

1. customer view

creating views for customer role

```
\c car_rental_management

--creating views for customer role
create view customer_view_outlet as select outlet_id,outlet_name,outlet_location from outlet;
create view customer_view_vehicle as select plate_number,outlet_id,model,number_of_seats,ac from vehicle;
create view customer_view_reservation as select reservation_id,reservation_date,vehicle_taken_date,expected_return_date,customer_id,outlet_id,advance,plt_num,reservation_status from reservation;
create view customer_view_rent as select * from rent;
create view customer_view_outlet_contact as select * from outlet_contact;
```

```
engineer@engineer:~/Desktop/dbms/assignment/car-rental-management$ sudo -u postgres psql -f multiusers.sql
You are now connected to database "car_rental_management" as user "postgres".
CREATE VIEW
CREATE VIEW
CREATE VIEW
CREATE VIEW
CREATE VIEW
```

customer role creation and granting views to customer role

```
--customer role creation and granting the views to customer role

DROP ROLE IF EXISTS customer_role;

create role customer_role with password 'qwerty';

grant select on customer_view_outlet to customer_role;

grant select on customer_view_vehicle to customer_role;

grant select,insert on customer_view_reservation to customer_role;

grant select on customer_view_rent to customer_role;

grant select on customer_view_outlet_contact to customer_role;
```

```
DROP ROLE
CREATE ROLE
GRANT
GRANT
GRANT
GRANT
GRANT
```

2. Employee view

Creating views of employee role

```
--creating views for employee role

create view employee_view_customer as select * from customer;

create view employee_view_outlet as select outlet_id,outlet_name,outlet_location from outlet;

create view employee_view_vehicle as select * from vehicle;

create view employee_view_reservation as select * from reservation;

create view employee_view_outlet_contact as select * from outlet_contact;
```

```
CREATE VIEW
CREATE VIEW
CREATE VIEW
CREATE VIEW
CREATE VIEW
```

employee role creation and granting views to employee role

```
--employee role creation and granting view to employee role

DROP ROLE IF EXISTS employee_role;

create role employee_role with password 'qwerty';

grant select on employee_view_customer to employee_role;

grant select on employee_view_outlet to employee_role;

grant select,insert,update,delete on employee_view_vehicle to employee_role;

grant select,update,delete on employee_view_reservation to employee_role;

grant select on employee_view_outlet_contact to employee_role;
```

```
DROP ROLE
CREATE ROLE
GRANT
GRANT
GRANT
GRANT
GRANT
```

3. Admin view

Creating views of admin role

```
--creating views of admin role

create view admin_view_customer as select * from customer;

create view admin_view_outlet as select * from outlet;

create view admin_view_employee as select * from employee;

create view admin_view_vehicle as select * from vehicle;

create view admin_view_reservation as select * from reservation;

create view admin_view_rent as select * from rent;

create view admin_view_discount as select * from discount;

create view admin_view_outlet_contact as select * from outlet_contact;

create view admin_view_got_discount as select * from got_discount;
```

```
CREATE VIEW
CREATE VIEW
CREATE VIEW
CREATE VIEW
CREATE VIEW
CREATE VIEW
CREATE VIEW
CREATE VIEW
CREATE VIEW
```

admin role creation and granting views to admin role

```
--admin role creation and granting views to admin role

DROP ROLE IF EXISTS admin_role;

create role admin_role with password 'qwerty';

grant select on admin_view_customer to admin_role;

grant select on admin_view_outlet to admin_role;

grant select,insert,update,delete on admin_view_employee to admin_role;

grant select,insert,update,delete on admin_view_vehicle to admin_role;
```

```
grant select on admin_view_reservation to admin_role;

grant select on admin_view_rent to admin_role;

grant select,insert,update,delete on admin_view_discount to admin_role;

grant select,update on admin_view_outlet_contact to admin_role;

grant select on admin_view_got_discount to admin_role;
```

```
DROP ROLE
CREATE ROLE
GRANT
GRANT
GRANT
GRANT
GRANT
GRANT
GRANT
GRANT
GRANT
```

Creating users

Creating customer, employee, admin users

```
--creating customer, employee, admin users
DROP USER IF EXISTS customer_user1;

create user customer_user1 with password '123';

DROP USER IF EXISTS employee_user1;

create user employee_user1 with password '123';

DROP USER IF EXISTS admin_user1;

create user admin_user1 with password '123';
```

```
DROP ROLE
CREATE ROLE
DROP ROLE
CREATE ROLE
DROP ROLE
CREATE ROLE
```

granting roles to each users.

```
--granting roles to each users

grant customer_role to customer_user1;

grant employee_role to employee_user1;

grant admin_role to admin_user1;
```

```
GRANT ROLE
GRANT ROLE
GRANT ROLE
```

Contribution of team member :

We divided the work equally, searched on internet for problems we faced, made many meetings on google meet.