DBMS

LAB – 8

NAME : SANMAT SANJAYAKUMAR PAYAGOUDAR

SRN : PES1UG20CS385

1. **Write a function to find the number of tickets booked by a customer. If no of tickets is more than 3 for the current month then display error message as “cannot purchase tickets current limit is over”.**

Query :

```
DELIMITER $$
CREATE FUNCTION count_tickets(ticket_385 int)
RETURNS VARCHAR(50)
DETERMINISTIC
BEGIN
DECLARE VALUE varchar(50);
IF ticket_385 > 3 then
set VALUE="Cannot purchase tickets Current limit is over";
    ELSE
set VALUE ="Can Purchase ticket_385";
    end if;
    return value;
END$$

DELIMITER ;
with t as (Select user_id, count(pnr) as count from ticket_385 group by
user_id )
select user_id, count_tickets(count) as Validate, count as ticket_purchased
from t;
```

OUTPUT :

```
MariaDB [railway_reservation]> DELIMITER $$
MariaDB [railway_reservation]> CREATE FUNCTION count_tickets(ticket_385 int)
    -> RETURNS VARCHAR(50)
    -> DETERMINISTIC
    -> BEGIN
    -> DECLARE VALUE varchar(50);
    -> IF ticket_385 > 3 then
    -> set VALUE="Cannot purchase tickets Current limit is over";
    ->     ELSE
    -> set VALUE ="Can Purchase ticket_385";
    ->     end if;
    ->     return value;
    -> END$$
Query OK, 0 rows affected (0.005 sec)

MariaDB [railway_reservation]> DELIMITER ;
MariaDB [railway_reservation]> with t as (Select user_id, count(pnr) as count from ticket_385 group by user_id )
    -> select user_id, count_tickets(count) as Validate, count as ticket_purchased from t;
+---------+-----------------------+------------------+
| user_id | Validate              | ticket_purchased |
+---------+-----------------------+------------------+
| ADM_001 | Can Purchase ticket_385 |                2 |
| ADM_002 | Can Purchase ticket_385 |                2 |
| ADM_003 | Can Purchase ticket_385 |                2 |
| USR_002 | Can Purchase ticket_385 |                2 |
| USR_007 | Can Purchase ticket_385 |                1 |
| USR_008 | Can Purchase ticket_385 |                2 |
| USR_009 | Can Purchase ticket_385 |                1 |
| USR_010 | Can Purchase ticket_385 |                1 |
| USR_012 | Can Purchase ticket_385 |                1 |
+---------+-----------------------+------------------+
9 rows in set (0.002 sec)
```

2. **Write a stored procedure to calculate the age of the customer when the date of birth is given. Update the column named age in the customer table.**

Query :

```
DELIMITER $$
CREATE procedure age_updation(
IN UID varchar(30),IN DB date, OUT msg varchar(30))
BEGIN
DECLARE age int;
IF DB>sysdate() THEN
    set msg= 'Invalid DoB';
ELSE
    update user_385
    set Age=(datediff(sysdate(),DB))/365
    where User_ID= UID;
    update user_385
    set DOB=DB
    where User_ID=UID;
    set msg=' Age updated Successfully';
END IF;
END$$

DELIMITER ;
CALL age_updation('ADM_003','1991-04-08',@A);
SELECT @A;
select * from user_385 where User_ID='ADM_003';
```

OUTPUT :

```
MariaDB [railway_reservation]> DELIMITER $$
MariaDB [railway_reservation]> CREATE procedure age_updation(
    -> IN UID varchar(30),IN DB date, OUT msg varchar(30))
    -> BEGIN
    -> DECLARE age int;
    -> IF DB>sysdate() THEN
    ->     set msg= 'Invalid DoB';
    -> ELSE
    ->     update user_385
    ->     set Age=(datediff(sysdate(),DB))/365
    ->     where User_ID= UID;
    ->     update user_385
    ->     set DOB=DB
    ->     where User_ID=UID;
    ->     set msg=' Age updated Successfully';
    -> END IF;
    -> END$$
Query OK, 0 rows affected (0.009 sec)

MariaDB [railway_reservation]>
MariaDB [railway_reservation]> DELIMITER ;
MariaDB [railway_reservation]> CALL age_updation('ADM_003','1991-04-08',@A);
Query OK, 1 row affected (0.017 sec)

MariaDB [railway_reservation]> SELECT @A;
+---------------------------+
| @A                        |
+---------------------------+
|  Age updated Successfully |
+---------------------------+
1 row in set (0.000 sec)

MariaDB [railway_reservation]> select * from user_385 where User_ID='ADM_003';
+---------+-----------+-----------+-------+------+------------+---------+----------------+
| User_ID | User_Type | FName     | LName | Age  | DOB        | PinCode | Street         |
+---------+-----------+-----------+-------+------+------------+---------+----------------+
| ADM_003 | ADMIN     | Manmohan  | Singh |   32 | 1991-04-08 |  575003 | Pumpwell Road  |
+---------+-----------+-----------+-------+------+------------+---------+----------------+
1 row in set (0.001 sec)
```