DATABASE MANAGEMENT AND SYSTEMS

NAME – RAHUL KATTINI

ROLL NO - S20200010091

QUESTION-1

```
mysql> delimiter //
mysql> create trigger trig
    -> BEFORE INSERT
    -> ON Distance FOR EACH ROW
    -> BEGIN
    -> declare fromCity1, toCity1 varchar(30);
    -> declare dist, test1, test2 INT;
    -> set fromCity1 = new.fromCity;
    -> set tocity1 = new.tocity;
    -> set dist = new.di;
    -> select count(*) into test1 from Distance where fromCity=fromCity1 and toCity=toCity1;
    -> select count(*) into test2 from Distance where fromCity=tocity1 and toCity=fromCity1;
    -> if test1>0 or test2> 0 then
    -> SIGNAL SQLSTATE '02000' SET MESSAGE_TEXT = "Distance between these
    "> cities already exist";
    -> END IF;
    -> END //
Query OK, 0 rows affected (0.06 sec)

mysql> DELIMITER ;
mysql> insert into distance values('chennai','Hyderabad', 470);
Query OK, 1 row affected (0.04 sec)

mysql> insert into distance values('gujarat','Hyderabad', 1332);
Query OK, 1 row affected (0.02 sec)
```

QUESTION-2

```
mysql> create table BankCustomers(accnum varchar(20), name varchar(20), loan int);
Query OK, 0 rows affected (0.06 sec)

mysql> delimiter //
mysql> CREATE PROCEDURE customer_insert(
    ->  in accno varchar(20),  in name varchar(20),  in loan int
    -> )
    -> deterministic
    -> begin
    -> if loan > 1000000 then
    -> select 'OOPS! Max limit as 10 Lakhs only' as error;
    -> else
    -> insert into bankcustomers values(iaccno,name,loan);
    -> end if;
    -> end //
Query OK, 0 rows affected (0.03 sec)

mysql> delimiter ;
mysql> insert into bankcustomers values('12', 'jhon', 1000001);
Query OK, 1 row affected (0.03 sec)

mysql> call customer_insert('12','jhon', 1000001);
+----------------------------------+
| error                            |
+----------------------------------+
| OOPS! Max limit as 10 Lakhs only |
+----------------------------------+
1 row in set (0.01 sec)

Query OK, 0 rows affected (0.02 sec)
```

QUESTION-3

```
mysql> delimiter //
mysql> create procedure merge_table()
    -> begin
    -> declare done int default 0;
    -> declare nt_id int;
    -> declare nt_name varchar(10);
    -> declare curmerge cursor for
    -> select id,name from table2;
    -> declare continue handler for not found set done = 1;
    -> open curmerge;
    -> merging: loop
    -> fetch curmerge into nt_id,nt_name;
    -> if done =1 then
    -> leave merging;
    -> end if;
    -> insert into table1 values(nt_id,nt_name);
    -> end loop merging;
    -> close curmerge;
    -> end //
Query OK, 0 rows affected (0.01 sec)
```

QUESTION-4

```
mysql> create table employees(name varchar(30), experience int, salary int(10));
Query OK, 0 rows affected, 1 warning (0.07 sec)

mysql> insert into employees values('sham', 11, 10000);
Query OK, 1 row affected (0.02 sec)

mysql> insert into employees values('siv', 45, 10040);
Query OK, 1 row affected (0.01 sec)

mysql> insert into employees values('sri', 1, 600000);
Query OK, 1 row affected (0.01 sec)

mysql> delimiter //
mysql> create procedure FindEmployee(in input INT)
    -> begin
    -> declare done INT DEFAULT FALSE;
    -> declare name1 varchar(40);
    -> declare salary1 INT;
    -> DECLARE cur1 CURSOR FOR SELECT name,
    -> salary FROM employees;
    -> DECLARE CONTINUE HANDLER FOR NOT FOUND SET done=TRUE;
    -> OPEN cur1;
    -> read_loop: LOOP
    -> FETCH cur1 INTO name1, salary1;
    -> IF done THEN
    -> LEAVE read_loop;
    -> END IF;
    -> if(salary1 < input) THEN
    -> SELECT name1 as Name, salary1 as Salary;
    -> END IF;
    -> END LOOP;
    -> CLOSE cur1;
    -> END //
Query OK, 0 rows affected (0.03 sec)

mysql> delimiter ;
mysql> call FindEmployee(10000);
Query OK, 0 rows affected (0.02 sec)

mysql> call FindEmployee(20000);
+------+--------+
| Name | Salary |
+------+--------+
| sham |  10000 |
+------+--------+
1 row in set (0.00 sec)

+------+--------+
| Name | Salary |
+------+--------+
| siv  |  10040 |
+------+--------+
1 row in set (0.01 sec)
```

QUESTION-5

```
mysql> delimiter //
mysql> create procedure
    -> UpdateSalaryofEmployees()
    -> begin
    -> declare done INT DEFAULT FALSE;
    -> declare temp decimal(15,2);
    -> declare name1 varchar(40);
    -> declare salary1, exp1 INT;
    -> DECLARE cur1 CURSOR FOR SELECT name, experience, salary FROM employees;
    -> DECLARE CONTINUE HANDLER FOR NOT FOUND SET done=TRUE;
    -> OPEN cur1;
    -> read_loop: LOOP
    -> FETCH cur1 INTO name1, exp1, salary1;
    -> IF done then
    -> leave read_loop;
    -> END IF;
    -> set temp=salary1;
    -> IF exp1>30 then
    -> set temp = (1.3) *salary1;
    -> elseif exp1>20 and exp1<=30 then set
    -> temp=(1.2)*salary1;
    -> elseif exp1>10 and exp1<=20 then set
    -> temp=(1.1)*salary1;
    -> END IF;
    -> SELECT name1 as name, exp1 as experience, salary1 as salary, temp as incrementedSalary;
    -> END LOOP;
    -> CLOSE cur1;
    -> END //
Query OK, 0 rows affected (0.02 sec)

mysql> delimiter ;
mysql> call UpdateSalaryofEmployees();
+------+------------+--------+-------------------+
| name | experience | salary | incrementedSalary |
+------+------------+--------+-------------------+
| sham |         11 |  10000 |          11000.00 |
+------+------------+--------+-------------------+
```

THANK YOU