DBMS Assignment 11
ANZAL HUSAIN ABIDI (20BCS009)

1) **Write a SQL function and stored procedure for average of three numbers.**

Function: -

```
mysql> create function 20BCavg3no(a int,b int,c int) returns int
    ->     deterministic
    -> begin
    ->   declare sum,avg int;
    ->   set sum = a+b+c;
    ->   set avg = sum/3;
    ->   return avg;
    -> end]
Query OK, 0 rows affected (0.00 sec)

mysql> select 20BCavg3no(4,5,6)]
+-------------------+
| 20BCavg3no(4,5,6) |
+-------------------+
|                 5 |
+-------------------+
1 row in set (0.00 sec)
```

Stored Procedure: -

```
mysql> create procedure 20BCavg3no(a int,b int,c int,out t int)
    ->     deterministic
    -> begin
    ->   declare sum int;
    ->   set sum = a+b+c;
    ->   set t = sum/3;
    -> end]
Query OK, 0 rows affected (0.00 sec)

mysql> call 20BCavg3no(1,2,3,@avg)]
Query OK, 0 rows affected (0.00 sec)

mysql> select @avg]
+------+
| @avg |
+------+
|    2 |
+------+
1 row in set (0.00 sec)
```

2) **Write a SQL function and stored procedure to calculate factorial.**

Function: -

```
mysql> create function 20BCfactorial(n int) returns int
    -> deterministic
    -> begin
    ->     declare f,i int default 1;
    ->     myloop:loop
    ->         if i > n then
    ->             leave myloop;
    ->         else
    ->             set f = f * i;
    ->             set i = i + 1;
    ->             iterate myloop;
    ->         end if;
    ->     end loop;
    ->     return f;
    -> end]
Query OK, 0 rows affected (0.00 sec)

mysql> select 20BCfactorial(5)]
+------------------+
| 20BCfactorial(5) |
+------------------+
|              120 |
+------------------+
```

Stored Procedure: -

```
mysql> create procedure 20BCfactorial(in n int,out fact int)
    -> deterministic
    -> begin
    ->     declare f,i int default 1;
    ->     myloop:loop
    ->         if i > n then
    ->             leave myloop;
    ->         else
    ->             set f = f * i;
    ->             set i = i + 1;
    ->             iterate myloop;
    ->         end if;
    ->     end loop;
    ->     set fact = f;
    -> end]
Query OK, 0 rows affected (0.01 sec)

mysql> call 20BCfactorial(5,@factorial)]
Query OK, 0 rows affected (0.00 sec)

mysql> select @factorial]
+------------+
| @factorial |
+------------+
|        120 |
+------------+
```

3) Write a SQL function and stored procedure to print fibonacci series upto n terms and its sum.

Function: -

```
mysql> create function 20BCfibonacci(n int) returns varchar(1000)
    ->     deterministic
    -> begin
    ->     declare i int default 3;
    ->     declare a,temp int default 0;
    ->     declare b,sum int default 1;
    ->     declare str varchar(1000);
    ->     set str = cast(a as char(2));
    ->     set str = concat(str,' ');
    ->     myloop:loop
    ->         if i > n then
    ->             leave myloop;
    ->         else
    ->             set temp = a + b;
    ->             set a = b;
    ->             set b = temp;
    ->             set i = i + 1;
    ->             set sum = sum + temp;
    ->             set str = concat(str, cast(a as char(2)));
    ->             set str = concat(str, ' ');
    ->         end if;
    ->     end loop;
    ->     set str = concat(str, cast(b as char(2)));
    ->     set str = concat(str, ' and sum = ');
    ->     set str = concat(str, cast(sum as char(2)));
    ->     return str;
    -> end]
Query OK, 0 rows affected (0.00 sec)

mysql> select 20BCfibonacci(6)]
+--------------------------+
| 20BCfibonacci(6)         |
+--------------------------+
| 0 1 1 2 3 5 and sum = 12 |
+--------------------------+
1 row in set (0.00 sec)
```

Stored Procedure: -

```
mysql> create procedure 20BCfibonacci(in n int,out retStr varchar(1000))
    ->     deterministic
    -> begin
    ->     declare i int default 3;
    ->     declare a,temp int default 0;
    ->     declare b,sum int default 1;
    ->     declare str varchar(1000);
    ->     set str = cast(a as char(2));
    ->     set str = concat(str,' ');
    ->     myloop:loop
    ->         if i > n then
    ->             leave myloop;
    ->         else
    ->             set temp = a + b;
    ->             set a = b;
    ->             set b = temp;
    ->             set i = i + 1;
    ->             set sum = sum + temp;
    ->             set str = concat(str, cast(a as char(2)));
    ->             set str = concat(str, ' ');
    ->         end if;
    ->     end loop;
    ->     set str = concat(str, cast(b as char(2)));
    ->     set str = concat(str, ' and sum = ');
    ->     set str = concat(str, cast(sum as char(2)));
    ->     set retStr = str;
    -> end]
Query OK, 0 rows affected (0.00 sec)

mysql> call  20BCfibonacci(6,@str)]
Query OK, 0 rows affected (0.00 sec)

mysql> select @str]
+--------------------------+
| @str                     |
+--------------------------+
| 0 1 1 2 3 5 and sum = 12 |
+--------------------------+
1 row in set (0.00 sec)
```

4) **Write a SQL function and stored procedure to calculate age.**
Function: -

```
mysql> create function 20BCcalcAge(dat date) returns varchar(25)
    ->     deterministic
    -> begin
    ->     declare curDate date default CURRENT_DATE();
    ->     declare tempDate date;
    ->     declare year, month, date int default 0;
    ->     declare str varchar(25) default '';
    ->     set year = TIMESTAMPDIFF(YEAR, dat, curDate);
    ->     set month = TIMESTAMPDIFF(MONTH, dat, curDate);
    ->     set month = month - (year * 12);
    ->     set tempDate = DATE_ADD(dat, INTERVAL year YEAR);
    ->     set tempDate = DATE_ADD(tempDate, INTERVAL month MONTH);
    ->     set date = DATEDIFF(curDate, tempDate) + 1;
    ->     set str = CONCAT(str,cast(year as char(2)));
    ->     set str = CONCAT(str,'Y ');
    ->     set str = CONCAT(str,cast(month as char(2)));
    ->     set str = CONCAT(str,'M ');
    ->     set str = CONCAT(str,cast(date as char(2)));
    ->     set str = CONCAT(str,'D');
    ->     return str;
    -> end]
Query OK, 0 rows affected (0.00 sec)

mysql> select 20BCcalcAge('2001-03-10')]
+---------------------------+
| 20BCcalcAge('2001-03-10') |
+---------------------------+
| 21Y 1M 19D                |
+---------------------------+
1 row in set (0.00 sec)
```

Stored Procedure:

```
mysql> create procedure 20BCcalcAge(in dat date,out retStr varchar(25))
    ->     deterministic
    -> begin
    ->     declare curDate date default CURRENT_DATE();
    ->     declare tempDate date;
    ->     declare year, month, date int default 0;
    ->     declare str varchar(25) default '';
    ->     set year = TIMESTAMPDIFF(YEAR, dat, curDate);
    ->     set month = TIMESTAMPDIFF(MONTH, dat, curDate);
    ->     set month = month - (year * 12);
    ->     set tempDate = DATE_ADD(dat, INTERVAL year YEAR);
    ->     set tempDate = DATE_ADD(tempDate, INTERVAL month MONTH);
    ->     set date = DATEDIFF(curDate, tempDate) + 1;
    ->     set str = CONCAT(str,cast(year as char(2)));
    ->     set str = CONCAT(str,'Y ');
    ->     set str = CONCAT(str,cast(month as char(2)));
    ->     set str = CONCAT(str,'M ');
    ->     set str = CONCAT(str,cast(date as char(2)));
    ->     set str = CONCAT(str,'D');
    ->     set retStr = str;
    -> end]
Query OK, 0 rows affected (0.01 sec)

mysql> call  20BCcalcAge('2001-03-10',@age)]

mysql> select @age]
+------------+
| @age       |
+------------+
| 21Y 1M 19D |
+------------+
1 row in set (0.00 sec)
```

5) **Write a SQL function and stored procedure to count the total no. of employees present in the employee table.**

Function: -

```
mysql> create function 20BCtotalNoEmployees() returns int
    -> deterministic
    -> begin
    ->   declare s int;
    ->   select count(*)  from employee into s;
    ->   return s;
    -> end]
Query OK, 0 rows affected (0.03 sec)

mysql> select 20BCtotalNoEmployees()]
+------------------------+
| 20BCtotalNoEmployees() |
+------------------------+
|                      8 |
+------------------------+
1 row in set (0.01 sec)
```

Stored Procedure: -

```
mysql> create procedure 20BCtotalNoEmployees(out count int)
    -> deterministic
    -> begin
    ->   declare s int;
    ->   select count(*)  from employee into s;
    ->   set count = s;
    -> end]
Query OK, 0 rows affected (0.00 sec)

mysql> call 20BCtotalNoEmployees(@res)]
Query OK, 1 row affected (0.00 sec)

mysql> select @res]
+------+
| @res |
+------+
|    8 |
+------+
1 row in set (0.00 sec)
```

6) Write a SQL function and stored procedure to calculate the budget of the department.

Function: -

```
mysql> create function 20BCcalcBudget(dept varchar(30)) returns int
    -> deterministic
    -> begin
    -> declare deptnumber varchar(5);
    -> declare budget int default 0;
    -> select Dno from department where Dept_name = dept into deptnumber;
    -> select sum(Salary) from employee where Dno = deptnumber into budget;
    -> return budget;
    -> end]
Query OK, 0 rows affected (0.01 sec)

mysql> select 20BCcalcBudget('Marketing')]
+----------------------------+
| 20BCcalcBudget('Marketing') |
+----------------------------+
|                      40000 |
+----------------------------+
1 row in set (0.00 sec)
```

Stored Procedure: -

```
mysql> create procedure 20BCcalcBudget(dept varchar(30), out budget int)
    ->     deterministic
    -> begin
    ->     declare deptnumber varchar(5);
    ->     declare sumSal int default 0;
    ->     select Dno from department where Dept_name = dept into deptnumber;
    ->     select sum(Salary) from employee where Dno = deptnumber into sumSal;
    ->     set budget = sumSal;
    -> end]
Query OK, 0 rows affected (0.00 sec)

mysql> call 20BCcalcBudget('Marketing',@res)]
Query OK, 1 row affected (0.00 sec)

mysql> select @res]
+-------+
| @res  |
+-------+
| 40000 |
+-------+
1 row in set (0.00 sec)
```

7) Write a SQL function and stored procedure to print the following message:

Function: -

```
mysql> create function 20BCprintMsg(name varchar(50)) returns varchar(100)
    ->     deterministic
    -> begin
    ->     declare msg varchar(100) default 'Hello ';
    ->     set msg = concat(msg, name);
    ->     set msg = concat(msg, ' How are you?');
    ->     return msg;
    -> end]
Query OK, 0 rows affected (0.00 sec)

mysql> select 20BCprintMsg('Ijlal')]
+--------------------------+
| 20BCprintMsg('Ijlal')    |
+--------------------------+
| Hello Ijlal How are you? |
+--------------------------+
1 row in set (0.00 sec)
```

Stored Procedure: -

```
mysql> create procedure 20BCprintMsg(name varchar(50), out message varchar(100))
    ->     deterministic
    -> begin
    ->     declare msg varchar(100) default 'Hello ';
    ->     set msg = concat(msg, name);
    ->     set msg = concat(msg, ' How are you?');
    ->     set message = msg;
    -> end]
Query OK, 0 rows affected (0.00 sec)

mysql> call 20BCprintMsg('Ijlal',@message)]
Query OK, 0 rows affected (0.00 sec)

mysql> select @message]
+--------------------------+
| @message                 |
+--------------------------+
| Hello Ijlal How are you? |
+--------------------------+
1 row in set (0.00 sec)
```

Triggers: -

LogTable

```
mysql> create table LogTable (
    ->     User      varchar(50),
    ->     Operation varchar(20),
    ->     Time      varchar(20),
    ->     Peid      varchar(5),
    ->     Pename    varchar(50),
    ->     Pesal     varchar(6),
    ->     Neid      varchar(5),
    ->     Nename    varchar(50),
    ->     Nesal     varchar(6)
    -> )]
Query OK, 0 rows affected (0.03 sec)
```

1) Insert Trigger

```
mysql> create trigger insertTrig after insert on employee for each row
    -> begin
    -> insert into logtable values (user(), 'Insert', now(), '-', '-', '-', new.Emp_id, new.Emp_name, new.Salary);
    -> end]
Query OK, 0 rows affected (0.01 sec)

mysql> insert into employee values (109, 'Suresh', 10000,'D1003')]
Query OK, 1 row affected (0.01 sec)

mysql> select * from LogTable]
+----------------+-----------+---------------------+------+--------+-------+------+--------+-------+
| User           | Operation | Time                | Peid | Pename | Pesal | Neid | Nename | Nesal |
+----------------+-----------+---------------------+------+--------+-------+------+--------+-------+
| root@localhost | Insert    | 2022-04-29 18:46:18 | -    | -      | -     | 109  | Suresh | 10000 |
+----------------+-----------+---------------------+------+--------+-------+------+--------+-------+
1 row in set (0.00 sec)
```

2) Update trigger

```
mysql> create trigger updateTrig after update on employee for each row
    -> begin
    ->     insert into logtable values (user(), 'Update', now(), old.Emp_id, old.Emp_name, old.Salary,
new.Emp_id, new.Emp_name, new.Salary);
    -> end]
Query OK, 0 rows affected (0.01 sec)

mysql> update employee set Salary = 20000 where Emp_id = 109]
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> select * from LogTable]
+----------------+-----------+---------------------+------+--------+-------+------+--------+-------+
| User           | Operation | Time                | Peid | Pename | Pesal | Neid | Nename | Nesal |
+----------------+-----------+---------------------+------+--------+-------+------+--------+-------+
| root@localhost | Insert    | 2022-04-29 18:46:18 | -    | -      | -     | 109  | Suresh | 10000 |
| root@localhost | Update    | 2022-04-29 18:51:31 | 109  | Suresh | 10000 | 109  | Suresh | 20000 |
+----------------+-----------+---------------------+------+--------+-------+------+--------+-------+
2 rows in set (0.00 sec)
```

3) Delete Trigger

```
mysql> create trigger deleteTrig after delete on employee for each row
    -> begin
    ->     insert into logtable values (user(), 'Delete', now(), old.Emp_id, old.Emp_name, old.Salary, '-', '-
','-');
    -> end]
Query OK, 0 rows affected (0.01 sec)

mysql> delete from employee where Emp_id = '109']
Query OK, 1 row affected (0.01 sec)

mysql> select * from LogTable]
+----------------+-----------+---------------------+------+--------+-------+------+--------+-------+
| User           | Operation | Time                | Peid | Pename | Pesal | Neid | Nename | Nesal |
+----------------+-----------+---------------------+------+--------+-------+------+--------+-------+
| root@localhost | Insert    | 2022-04-29 18:46:18 | -    | -      | -     | 109  | Suresh | 10000 |
| root@localhost | Update    | 2022-04-29 18:51:31 | 109  | Suresh | 10000 | 109  | Suresh | 20000 |
| root@localhost | Delete    | 2022-04-29 22:20:25 | 109  | Suresh | 20000 | -    | -      | -     |
+----------------+-----------+---------------------+------+--------+-------+------+--------+-------+
3 rows in set (0.00 sec)
```

4) Cursor: -

Write a cursor to output salary of all employees in a string.

```
mysql> select * from employee]
+--------+----------+--------+-------+
| Emp_id | Emp_name | Salary | Dno   |
+--------+----------+--------+-------+
|    101 | Amit     |  25000 | D1001 |
|    102 | Sunil    |  20000 | D1002 |
|    103 | Rakesh   |  18000 | D1003 |
|    104 | Ajay     |  16000 | D1001 |
|    105 | Suhail   |  20000 | D1002 |
|    106 | Arif     |  18000 | D1004 |
|    107 | Suresh   |  24000 | D1002 |
|    108 | Vijay    |  22000 | D1003 |
+--------+----------+--------+-------+
8 rows in set (0.00 sec)

mysql> create procedure mypro(out s varchar(6))
    ->     deterministic
    -> begin
    ->     declare f int default 1;
    ->     declare str longtext default '';
    ->     declare cur cursor for select Salary from employee;
    ->     declare continue handler for not found set f = 0;
    ->     open cur;
    ->     myloop:
    ->     loop
    ->         fetch cur into s;
    ->         if f = 0 then
    ->             leave myloop;
    ->         else
    ->             set str = concat(str, ' ', s);
    ->         end if;
    ->     end loop;
    ->     close cur;
    ->     select str;
    -> end]
Query OK, 0 rows affected (0.01 sec)
```

```
mysql> call mypro(@s)]
+-------------------------------------------------+
| str                                             |
+-------------------------------------------------+
|  25000 20000 18000 16000 20000 18000 24000 22000 |
+-------------------------------------------------+
1 row in set (0.00 sec)

Query OK, 0 rows affected (0.01 sec)
```