

# DBMS – LAB -07

**NAME** : RAHUL VARMA

**ROLL NO**: S20200010212

**SECTION**: C

**TASK**: (LAB EXERCISES)

SQL COMMANDS:

Topic:

SQL procedures

## 1. Create a procedure to insert a tuple into any table of bank schema

### Procedure:

```
mysql> delimiter /
mysql> create procedure insert_tuple()
  -> begin
  -> insert into branch values("south town","sye town","400000");
  -> end /
Query OK, 0 rows affected (0.02 sec)

mysql> delimiter ;
```

### Contents in table before calling procedure:

```
mysql> select * from branch;
+-----+-----+-----+
| branch_name | branch_city | assets      |
+-----+-----+-----+
| Brighton    | Brooklyn    | 7100000.00  |
| Downtown    | Brooklyn    | 9000000.00  |
| Mianus       | Horseneck   | 400000.00   |
| North Town  | Rye         | 3700000.00  |
| Perryridge  | Horseneck   | 1700000.00  |
| Pownal      | Bennington  | 300000.00   |
| Redwood     | Palo Alto   | 2100000.00  |
| Round Hill  | Horseneck   | 8000000.00  |
+-----+-----+-----+
8 rows in set (0.04 sec)
```

Contents in table after calling procedure:  
(One tuple is added by procedure "south town", "sye town", 400000.00)

```
mysql> call insert_tuple();
Query OK, 1 row affected (0.01 sec)

mysql> select * from branch;
+-----+-----+-----+
| branch_name | branch_city | assets |
+-----+-----+-----+
| Brighton    | Brooklyn   | 7100000.00 |
| Downtown    | Brooklyn   | 9000000.00 |
| Mianus       | Horseneck  | 400000.00 |
| North Town  | Rye        | 3700000.00 |
| Perryridge  | Horseneck  | 1700000.00 |
| Pownal      | Bennington | 300000.00 |
| Redwood     | Palo Alto  | 2100000.00 |
| Round Hill  | Horseneck  | 8000000.00 |
| south town  | sye town   | 400000.00 |
+-----+-----+-----+
9 rows in set (0.00 sec)
```

## 2. Drop the created procedure

Procedure for dropping table:

```
mysql> drop procedure insert_tuple;  
Query OK, 0 rows affected (0.02 sec)  
  
mysql>
```

After dropping table again calling to check whether the procedure is present or not (it should not be present)

So we got error

So Successfully table dropped

```
mysql> call insert_tuple;  
ERROR 1305 (42000): PROCEDURE lab5.insert_tuple does not exist  
mysql>
```

3. Create a procedure that takes any 2 numbers and returns their sum and multiplication.

Procedure and calling procedure:

```
mysql> delimiter /
mysql> create procedure sum_and_product(in x int,in y int, out sum int, out product int)
  -> begin
  -> set sum = x+y;
  -> set product = x*y;
  -> end /
Query OK, 0 rows affected (0.02 sec)

mysql> delimiter ;
```

## Calling procedure and printing the final answers:

```
mysql> call sum_and_product(5,10,@sum,@product);  
Query OK, 0 rows affected (0.01 sec)
```

```
mysql> select @sum;
```

```
+-----+  
| @sum |  
+-----+  
|    15 |  
+-----+  
1 row in set (0.00 sec)
```

```
mysql> select @product;
```

```
+-----+  
| @product |  
+-----+  
|        50 |  
+-----+  
1 row in set (0.00 sec)
```

```
mysql>
```

4. Write a procedure with only one parameter such that it returns the square root of any given number

Procedure:

```
mysql> delimiter /
mysql> create procedure Square_root(inout x int)
  -> begin
  -> set x = sqrt(x);
  -> end /
Query OK, 0 rows affected (0.02 sec)

mysql> delimiter ;
```

Calling procedure:

```
mysql> set @root = 49;
Query OK, 0 rows affected (0.00 sec)

mysql> call Square_root(@root);
Query OK, 0 rows affected (0.00 sec)

mysql> select @root;
+-----+
| @root |
+-----+
|      7 |
+-----+
1 row in set (0.00 sec)

mysql> _
```

## 5. Write a Procedure that returns no.of characters in any given string

### Procedure:

```
mysql> delimiter /
mysql> create procedure string_length(in str varchar(30), out length int)
-> begin
-> set length = length(str);
-> end /
Query OK, 0 rows affected (0.01 sec)
```

### Calling procedure:

```
mysql> delimiter ;
mysql> call string_length("helloworld",@length);
Query OK, 0 rows affected (0.00 sec)

mysql> select @length;
+-----+
| @length |
+-----+
|      10 |
+-----+
1 row in set (0.00 sec)

mysql>
```



6. Write a procedure with only one parameter such that it displays the factorial of the given number

Procedure and calling procedure:

```
mysql> delimiter /
mysql> create procedure factorial(in n int)
  -> begin
  -> declare i int default 1;
  -> declare facto bigint default 1;
  -> while i<=n do
  -> set facto = facto*i;
  -> set i=i+1;
  -> end while;
  -> select n as input_number , facto as factorial;
  -> end /
Query OK, 0 rows affected (0.02 sec)
```

```
mysql> delimiter ;
mysql> call factorial(6);
+-----+-----+
| input_number | factorial |
+-----+-----+
|          6 |        720 |
+-----+-----+
1 row in set (0.00 sec)
```

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

7. Write a procedure(that accepts 2 arguments,one argument has old name the other holds new name) to update name of the existing customer to a new name

Procedure:

```
mysql> delimiter /
mysql> create procedure update_name(in old varchar(30), in new varchar(30))
-> begin
-> update branch set branch_name=new where branch_name=old;
-> end /
Query OK, 0 rows affected (0.01 sec)

mysql> delimiter ;
```

Content in branch before calling procedure:

```
mysql> select * from branch;
+-----+-----+-----+
| branch_name | branch_city | assets |
+-----+-----+-----+
| Brighton   | Brooklyn   | 7100000.00 |
| Downtown   | Brooklyn   | 9000000.00 |
| Mianus      | Horseneck  | 400000.00 |
| North Town | Rye        | 3700000.00 |
| Perryridge | Horseneck  | 1700000.00 |
| Pownal     | Bennington | 300000.00 |
| Redwood    | Palo Alto  | 2100000.00 |
| Round Hill | Horseneck  | 8000000.00 |
| south town | sye town   | 400000.00 |
+-----+-----+-----+
9 rows in set (0.01 sec)
```

## Content in branch after calling procedure:

```
mysql> call update_name("south town","east town");
```

```
Query OK, 1 row affected (0.02 sec)
```

```
mysql> select * from branch;
```

branch_name	branch_city	assets
Brighton	Brooklyn	7100000.00
Downtown	Brooklyn	9000000.00
east town	sydney	400000.00
Mianus	Horseneck	400000.00
North Town	Rye	3700000.00
Perryridge	Horseneck	1700000.00
Pownal	Bennington	300000.00
Redwood	Palo Alto	2100000.00
Round Hill	Horseneck	8000000.00

```
9 rows in set (0.00 sec)
```

```
mysql>
```

## 8. Procedure that accepts customer id and displays whether he has loan or Not

### Procedure:

```
mysql> delimiter /
mysql> create procedure loan_exists_or_not(in cust_name varchar(30))
-> begin
-> declare present varchar(30);
-> declare absent varchar(30);
-> declare i int default 0;
-> set present = "customer has loan";
-> set absent = "customer doesn't has loan";
-> select count(loan_number) into i from borrower where customer_name=cust_name;
-> if i>0 then
-> select present;
-> else select absent;
-> end if;
-> end /
Query OK, 0 rows affected (0.02 sec)

mysql> delimiter ;
```

### Content in borrower:

```
mysql> select * from borrower;
```

customer_name	loan_number
Smith	L-11
Hayes	L-15
Adams	L-16
Jones	L-17
Williams	L-17
Smith	L-23
Curry	L-93

7 rows in set (0.01 sec)

## Calling procedure :

```
mysql> call loan_exists_or_not("Smith");
```

present
customer has loan

1 row in set (0.01 sec)

Query OK, 0 rows affected (0.01 sec)

```
mysql> call loan_exists_or_not("Rahul");
```

absent
customer doesn't has loan

1 row in set (0.00 sec)

Query OK, 0 rows affected (0.01 sec)

9. Display city of given customer and If he/she is not an existing customer create a new entry in customer table

Procedure:

```
mysql> delimiter /
mysql> create procedure que_9(in name varchar(30))
-> begin
-> declare i int default 0;
-> select count(*) into i from customer where customer_name=name;
-> if i<1 then
-> insert into customer values(name,"albert","slimfield");
-> end if;
-> select * from customer where customer_name=name;
-> end /
Query OK, 0 rows affected (0.02 sec)
```

## Content in customer table :

```
mysql> select * from customer/
+-----+-----+-----+
| customer_name | customer_street | customer_city |
+-----+-----+-----+
| Adams        | Spring          | Pittsfield    |
| Brooks       | Senator         | Brooklyn      |
| Curry        | North           | Rye            |
| Glenn        | Sand Hill       | Woodside      |
| Green        | Walnut          | Stamford      |
| Hayes        | Main            | Harrison       |
| Johnson      | Alma            | Palo Alto     |
| Jones        | Main            | Harrison       |
| Lindsay      | Park            | Pittsfield     |
| Smith        | North           | Rye            |
| Turner       | Putnam          | Stamford       |
| Williams     | Nassau          | Princeton     |
+-----+-----+-----+
12 rows in set (0.04 sec)

mysql> delimiter ;
```

## Procedure calling:

```
mysql> call que_9("Brooks");
+-----+-----+-----+
| customer_name | customer_street | customer_city |
+-----+-----+-----+
| Brooks       | Senator        | Brooklyn      |
+-----+-----+-----+
1 row in set (0.01 sec)

Query OK, 0 rows affected (0.02 sec)

mysql> call que_9("Rahul");
+-----+-----+-----+
| customer_name | customer_street | customer_city |
+-----+-----+-----+
| Rahul        | albert         | slimfield     |
+-----+-----+-----+
1 row in set (0.01 sec)

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



## 10. Display account numbers of customers whose balance is above the given Amount

### Procedure:

```
mysql> delimiter /
mysql> create procedure que_10(in amount int)
  -> begin
  -> select account_number from account where balance > amount;
  -> end /
Query OK, 0 rows affected (0.02 sec)
```

### Procedure calling: (input with 700)

```
mysql> delimiter ;
mysql> call que_10("700");
+-----+
| account_number |
+-----+
| A-201          |
| A-217          |
+-----+
2 rows in set (0.02 sec)

Query OK, 0 rows affected (0.04 sec)
```

## Procedure calling with input 500

```
mysql> call que_10("500");  
+-----+  
| account_number |  
+-----+  
| A-201          |  
| A-215          |  
| A-217          |  
| A-222          |  
+-----+  
4 rows in set (0.00 sec)
```

THANK YOU