

-----CREATE & DESC TABLES-----

```
mysql> CREATE TABLE bank_customer (  
-> cust_id INT PRIMARY KEY AUTO_INCREMENT,  
-> cust_name VARCHAR(30), email VARCHAR(30));  
Query OK, 0 rows affected (0.04 sec)
```

```
mysql> DESC bank_customer;
```

Field	Type	Null	Key	Default	Extra
cust_id	int	NO	PRI	NULL	auto_increment
cust_name	varchar(30)	YES		NULL	
email	varchar(30)	YES		NULL	

3 rows in set (0.02 sec)

```
mysql> INSERT INTO bank_customer(cust_name, email) VALUES ('Hari','hari@gmail.com'),  
('Sam','sam@gmail.com'), ('Sree','sree@lbscek.ac.in'), ('Ali','ali@lbscek.ac.in'), ('John','john@yahoo.co.in');  
Query OK, 5 rows affected (0.01 sec)  
Records: 5 Duplicates: 0 Warnings: 0
```

```
mysql> SELECT * FROM bank_customer;
```

	cust_id	cust_name	email
1	Hari	hari@gmail.com	
2	Sam	sam@gmail.com	
3	Sree	sree@lbscek.ac.in	
4	Ali	ali@lbscek.ac.in	
5	John	john@yahoo.co.in	

```
mysql> CREATE TABLE account (  
-> acc_no INT PRIMARY KEY, acc_type ENUM('saving', 'current'),  
-> cust_id INT, balance INT,  
-> FOREIGN KEY (cust_id) REFERENCES bank_customer(cust_id));  
Query OK, 0 rows affected (0.05 sec)
```

```
mysql> DESC account;
```

Field	Type	Null	Key	Default	Extra
acc_no	int	NO	PRI	NULL	
acc_type	enum('saving','current')	YES		NULL	
cust_id	int	YES	MUL	NULL	
balance	int	YES		NULL	

4 rows in set (0.00 sec)

-----INSERT VALUES FROM FILE-----

**Q1: Import DATA of account table from a csv file**

```
mysql> SHOW GLOBAL VARIABLES LIKE 'local_infile';
```

```
+-----+-----+
| Variable_name | Value |
+-----+-----+
| local_infile | OFF   |
+-----+-----+
```

```
mysql> SET GLOBAL local_infile=1;
```

```
mysql> SHOW GLOBAL VARIABLES LIKE 'local_infile';
```

```
+-----+-----+
| Variable_name | Value |
+-----+-----+
| local_infile | ON    |
+-----+-----+
```

```
malar@dell:~$ sudo mysql --local-infile=1
```

```
mysql> use s5lbscek;
```

```
mysql> SELECT * FROM account;
```

Empty set (0.00 sec)

```
mysql> LOAD DATA LOCAL INFILE '/home/malar/Documents/DBMSLABRECORD/account.csv'
```

```
-> INTO TABLE account
```

```
-> FIELDS TERMINATED BY ','
```

```
-> ENCLOSED BY '"'
```

```
-> LINES TERMINATED BY '\n'
```

```
-> IGNORE 1 LINES;
```

Query OK, 5 rows affected (0.00 sec)

Records: 5 Deleted: 0 Skipped: 0 Warnings: 0

```
mysql> SELECT * FROM account;
```

```
+-----+-----+-----+-----+
| acc_no | acc_type | cust_id | balance |
+-----+-----+-----+-----+
| 1001 | saving | 1 | 10000 |
| 1002 | saving | 2 | 20000 |
| 1003 | saving | 3 | 30000 |
| 2001 | current | 4 | 10000 |
| 2002 | current | 5 | 20000 |
+-----+-----+-----+-----+
```

**Q2. create a function getBalance(acc\_no) to get the balance details of a bank account.**

```
mysql> DELIMITER ;
```

```
mysql> DELIMITER //
```

```
mysql> CREATE FUNCTION getBalance(accno INT) RETURNS INT DETERMINISTIC
```

```
-> BEGIN
```

```

-> DECLARE bal INT;
-> SELECT balance INTO bal FROM account WHERE acc_no=accno;
-> RETURN bal;
-> END //

```

Query OK, 0 rows affected (0.00 sec)

mysql> DELIMITER ;

Q3. create two procedures - credit(acc\_no,amount), debit(acc\_no,amount).  
Display appropriate messages of every operations and errors.

mysql> DELIMITER //

mysql> CREATE PROCEDURE credit(IN accno INT, IN amount INT)

```

-> BEGIN
-> DECLARE crebal INT;
-> SELECT balance+amount INTO crebal FROM account WHERE acc_no=accno;
-> IF (SELECT COUNT(*) FROM account WHERE acc_no=accno)=0 THEN
->   SELECT 'Invalid Account Number' AS ERROR;
-> ELSE
->   UPDATE account SET balance=crebal WHERE acc_no=accno;
->   SELECT CONCAT(amount, ' Deposited To ', accno) AS SUCCESS, crebal AS current_balance;
-> END IF;
-> END //

```

Query OK, 0 rows affected (0.00 sec)

mysql> DELIMITER ;

mysql> DELIMITER //

mysql> CREATE PROCEDURE debit(IN accno INT, IN amount INT)

```

-> BEGIN
-> DECLARE dedbal INT;
-> SELECT balance-amount INTO dedbal FROM account WHERE acc_no=accno;
-> IF (SELECT COUNT(*) FROM account WHERE acc_no=accno)=0 THEN
->   SELECT 'Invalid Account Number' AS ERROR;
-> ELSEIF dedbal<0 THEN
->   SELECT 'Insufficient Balance' AS ERROR;
-> ELSEIF (SELECT acc_type FROM account WHERE acc_no=accno)='saving'
->   AND dedbal<1000 THEN
->   SELECT 'Insufficient Minimum Balance' AS ERROR;
-> ELSE
->   UPDATE account SET balance=dedbal WHERE acc_no=accno;
->   SELECT CONCAT(amount, ' Withdrawn From ', accno) AS SUCCESS, dedbal AS current_balance;
-> END IF;
-> END //

```

Query OK, 0 rows affected (0.01 sec)

mysql> DELIMITER ;

**Q4. Use curosr and create a procedure to list all email ids of customer separated by semicolon;**

```
mysql> DELIMITER //
mysql> CREATE PROCEDURE getemail(INOUT emails TEXT)
-> BEGIN
-> DECLARE done BOOL DEFAULT FALSE;
-> DECLARE email_id VARCHAR(30) DEFAULT "";
-> DECLARE cur CURSOR FOR SELECT email FROM bank_customer;
-> DECLARE CONTINUE HANDLER FOR NOT FOUND
-> SET done=TRUE;
-> OPEN cur;
-> SET emails="";
-> process_email: LOOP
-> FETCH cur INTO email_id;
-> IF done=TRUE THEN
-> LEAVE process_email;
-> END IF;
-> SET emails=CONCAT(email_id, '; ', emails);
-> END LOOP;
-> CLOSE cur;
-> END //
```

Query OK, 0 rows affected (0.01 sec)

mysql> DELIMITER ;

-----OUTPUT-----USE-----

mysql> SELECT \* FROM account;

```
+-----+-----+-----+
| acc_no | acc_type | cust_id | balance |
+-----+-----+-----+
| 1001 | saving | 1 | 10000 |
| 1002 | saving | 2 | 20000 |
| 1003 | saving | 3 | 30000 |
| 2001 | current | 4 | 10000 |
| 2002 | current | 5 | 20000 |
+-----+-----+-----+
```

5 rows in set (0.00 sec)

mysql> SELECT getBalance(1001) AS currentBalance;

```
+-----+
| currentBalance |
+-----+
| 10000 |
+-----+
```

1 row in set (0.00 sec)

mysql> CALL credit(1001, 2000);

```
+-----+-----+-----+
| acc_no | acc_type | cust_id | balance |
+-----+-----+-----+
```

SUCCESS	current_balance
2000 Deposited To 1001	12000

1 row in set (0.00 sec)

Query OK, 0 rows affected (0.00 sec)

mysql> CALL credit(1004, 1000);

ERROR
Invalid Account Number

1 row in set (0.00 sec)

mysql> CALL debit(1001, 1000);

SUCCESS	current_balance
1000 Withdrawn From 1001	11000

1 row in set (0.01 sec)

Query OK, 0 rows affected (0.01 sec)

mysql> CALL debit(1004, 1000);

ERROR
Invalid Account Number

1 row in set (0.00 sec)

Query OK, 0 rows affected (0.00 sec)

mysql> CALL debit(1001, 10500);

ERROR
Insufficient Minimum Balance

1 row in set (0.00 sec)

Query OK, 0 rows affected (0.00 sec)

mysql> CALL debit(1001, 10000);

SUCCESS	current_balance
10000 Withdrawn From 1001	1000

```
+-----+
1 row in set (0.01 sec)
```

Query OK, 0 rows affected (0.01 sec)

```
mysql> CALL debit(2001, 10000);
```

```
+-----+
| SUCCESS          | current_balance |
+-----+
| 10000 Withdrawn From 2001 |          0 |
+-----+
1 row in set (0.01 sec)
```

Query OK, 0 rows affected (0.01 sec)

```
mysql> SELECT getBalance(2001) AS currentBalance;
```

```
+-----+
| currentBalance |
+-----+
|          0 |
+-----+
1 row in set (0.00 sec)
```

```
mysql> SELECT * FROM account;
```

```
+-----+-----+-----+
| acc_no | acc_type | cust_id | balance |
+-----+-----+-----+
| 1001 | saving | 1 | 1000 |
| 1002 | saving | 2 | 20000 |
| 1003 | saving | 3 | 30000 |
| 2001 | current | 4 | 0 |
| 2002 | current | 5 | 20000 |
+-----+-----+-----+
5 rows in set (0.00 sec)
```

```
mysql> CALL getemail(@email_list);
```

Query OK, 0 rows affected (0.00 sec)

```
mysql> SELECT @email_list;
```

```
+-----+
| @email_list |
+-----+
| john@yahoo.co.in; ali@lbscek.ac.in; sree@lbscek.ac.in; sam@gmail.com; hari@gmail.com; |
+-----+
1 row in set (0.00 sec)
```