Q)Write a Java program to connect to MySQL DBMS(Database Server).

```
//ConnectionApplication.java
import java.sql.DriverManager;
import java.sql.Connection;
import java.sql.SQLException;
class  ConnectionApplication
{
      public static void main(String[] args) throws SQLException
      {
      Connection
connection=DriverManager.getConnection("jdbc:mysql://localhost:3306","roo
t","ashokit");
      System.out.println("JDBC client connected to MySQL server");
      connection.close();
      System.out.println("After performing database operations,
connection is closed");
      }
}
```

Q)Write a Java program to store an account's data(accno,name,balance) into the database.

Note:- Before writing and executing this program, the following things should have been done in MySQL DBMS.

Step 1:- Login to MySQL DBMS.

Step 2:- create a database

```
mysql>CREATE DATABASE hdfcdb;
```

Step 3:- enter into the database.

```
mysql>USE  hdfcdb;
```

STEP 4:- create the table

```
mysql>CREATE TABLE ACCOUNT(ACCNO INT(11),NAME VARCHAR(20),BALANCE
DOUBLE(10,2));
```

```
//StoreAccount.java
//StoreAccount.java
import java.sql.*;//making JDBC API available
class StoreAccount
{
      public static void main(String[] args) throws SQLException
      {
      Connection
connection=DriverManager.getConnection("jdbc:mysql://localhost:3306/hdfcd
b","root","ashokit");
      Statement statement=connection.createStatement();
        int re=statement.executeUpdate("INSERT INTO ACCOUNT
VALUES(10001,'Rama',50000)");
      System.out.println(re+" account stored into the database
successfully");
      statement.close();
      connection.close();
```

```
      }
}
```
Q)Write a Java program to increase the balance of all the accounts by Rs.1000.

```
//DepositApplication.java
//StoreAccount.java
import java.sql.*;//making JDBC API available
class DepositApplication
{
      public static void main(String[] args) throws SQLException
      {
      Connection
connection=DriverManager.getConnection("jdbc:mysql://localhost:3306/hdfcd
b","root","ashokit");
      Statement statement=connection.createStatement();
        int re=statement.executeUpdate("UPDATE ACCOUNT SET
BALANCE=BALANCE+1000");
      System.out.println(re+" accounts updated");
      statement.close();
      connection.close();
      }
}
```