**Server**

import java.io.\*;

import java.net.\*;

import java.sql.\*;

public class Server implements Runnable {

private final Socket Socket;

public Server(Socket clientSocket) {

this.Socket = clientSocket;

}

Server() {

throw new UnsupportedOperationException("Not supported yet.");

}

@Override

public void run() {

try{

DataOutputStream dos;

Connection connection;

try (DataInputStream dis = new DataInputStream(Socket.getInputStream())) {

ServerSocket serverSocket = new ServerSocket(8151);

dos = new DataOutputStream(Socket.getOutputStream());

int repID = dis.readInt();

int laptopsSold = dis.readInt();

double unitPrice = 90.0;

double salesProfit = laptopsSold \* unitPrice;

String chargeCode;

if (salesProfit > 20000) {

chargeCode = "1";

} else if (salesProfit > 10000) {

chargeCode = "2";

} else {

chargeCode = "3";

}

String URL = "jdbc:derby://localhost:1527/DG";

String name = "DG";

String pass = "DG";

connection = DriverManager.getConnection(URL, name, pass);

PreparedStatement statement = connection.prepareStatement("SELECT ChargeRate FROM ChargeRates WHERE ChargeCode = ?");

statement.setString(1, chargeCode);

ResultSet resultSet = statement.executeQuery();

double commissionR = 0.0;

if (resultSet.next()) {

commissionR = resultSet.getDouble("ChargeRate");

}

double commissionV = salesProfit \* (commissionR / 100);

dos.writeDouble(salesProfit);

dos.writeDouble(commissionR);

dos.writeDouble(commissionV);

}

dos.close();

Socket.close();

connection.close();

} catch (IOException | SQLException e) {

}

}

}

**Main Server**

public class Mainnserver {

public static void main(String[] args) {

try {

Server server = new Server(); // Ensure Server implements Runnable

Thread serverThread = new Thread(server);

serverThread.start();

System.out.println("Main server has started successfully.");

} catch (Exception e) {

System.out.println("Server is running");

}

}

}

**Client**

import java.io.\*;

import java.net.\*;

import java.util.\*;

public class Client implements Runnable {

@Override

public void run() {

Scanner scanner = new Scanner(System.in);

try {

Socket sN = new Socket("localhost", 8151);

System.out.println("Connected to the server.");

System.out.print("Enter Sales Representative ID: ");

int repID = scanner.nextInt();

int sSold;

while (true) {

System.out.print("Enter the number of laptops sold: ");

sSold = scanner.nextInt();

if (sSold >= 0) break;

System.out.println("Invalid input. Please enter a non-negative number.");

}

DataOutputStream DOS = new DataOutputStream(sN.getOutputStream());

DOS.writeInt(repID);

DOS.writeInt(sSold);

System.out.println("Data sent to the server.");

DataInputStream DIS = new DataInputStream(sN.getInputStream());

double salesProfit = DIS.readDouble();

double commissionR = DIS.readDouble();

double commissionV = DIS.readDouble();

System.out.println("=== Commission Details ===");

System.out.println("Sales Representative ID: " + repID);

System.out.printf("Sales Profit: %.2f OMR%n", salesProfit);

System.out.printf("Commission Rate: %.2f%%%n", commissionR);

System.out.printf("Commission Value: %.2f OMR%n", commissionV);

DIS.close();

DOS.close();

sN.close();

} catch (IOException e) {

System.out.println("Error: " + e.getMessage());

}

}

}

**Main Client**

class MainClient {

public static void main(String[] args) {

Client client = new Client();

Thread clientThread = new Thread(client);

clientThread.start();

}

}