

Banking System

(Experiment 1 - Communication)

Yash Brid 2019130008

Abhishek Chopra 2019130009

Sumeet Haldipur 2019130018

Aim : To implement a Client Server based program using RMI.

Objective :

- To learn about remote process communication using RMI.
- To implement a client server based program for a Banking System using RMI in Java.

Problem Statement : Implement a Banking System using client server communication with the help of RMI in Java.

Code:

```
import java.rmi.RemoteException;
import java.rmi.registry.LocateRegistry;
import java.rmi.registry.Registry;
import java.rmi.server.UnicastRemoteObject;
import java.util.ArrayList;
public class Server extends UnicastRemoteObject implements checkBal {
    public Server() throws RemoteException {
        super();
    }
    static ArrayList<Account> a = new ArrayList<Account>();
    public double checkBalance(String acc_no, String password) throws
RemoteException {
    System.out.println("Request received for account number "+acc_no); for
        (int i = 0; i < a.size(); i++) {
    double bal = a.get(i).checkBalance(acc_no, password); if (bal !=
        -1)
    return bal;
    }
    return -1.0;
}
```

Server.java

```

        public static void main(String[] args)
        { try {
Registry reg = LocateRegistry.createRegistry(8000);
        reg.rebind("bankServer", new Server());
        System.out.println("Server is running..");
        a.add(new Account("123456", "password1",
        2000.0)); a.add(new Account("456789",
        "password2", 3700.50));
        } catch (Exception e)
        {
            e.printStackTrace(
            );
        }
    }
}

class Account {
    String acc_no;
    String
    password;
    double
    balance;

    Account(String acc_no, String password, double balance)
    { this.acc_no = acc_no;
this.password = password;
        this.balance = balance;
    }
}

```

```

public double checkBalance(String acc_no, String password) {
    if (this.acc_no.equals(acc_no) &&
        ...
    )
    ...
}

```

```

import java.rmi.*;

public interface checkBal extends Remote {
    public double checkBalance(String acc_no, String password) throws
    RemoteException;
}

```

checkBal.java

Client.java

```
import java.rmi.RemoteException;
import java.rmi.registry.LocateRegistry;
import java.rmi.registry.Registry;
import java.util.Scanner;

public class Client {
    public static void main(String args[]) throws RemoteException {
        try {
            Scanner sc = new Scanner(System.in);
            Registry reg = LocateRegistry.getRegistry("localhost", 8000); checkBal ob
                = (checkBal) reg.lookup("bankServer");
            System.out.print("\nEnter account number:");
            String acc_no = sc.nextLine();
            System.out.print("Enter password:");
            String password = sc.nextLine();
            double bal = ob.checkBalance(acc_no, password); if (bal ==
                -1) {
            System.out.println("\nInvalid credentials"); return;
            } else {
            System.out.println("\nBalance: Rs." + bal+"\n");
            }
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

Output

```
Javadoc Declaration Console Error Log Console x
<terminated> Client [Java Application] C:\Program Files\Java\jdk-14.0.1\bin\javaw.exe (07-Dec-2021, 1:14:43 pm – 1:14:50 pm)

Enter account number:123456
Enter password:password1

Balance: Rs.2000.0

Javadoc Declaration Console Error Log Console x
<terminated> Client [Java Application] C:\Program Files\Java\jdk-14.0.1\bin\javaw.exe (07-Dec-2021, 1:16:06 pm – 1:16:29 pm)

Enter account number:456789
Enter password:password2

Balance: Rs.3700.5

Javadoc Declaration Console Error Log Console x
<terminated> Client [Java Application] C:\Program Files\Java\jdk-14.0.1\bin\javaw.exe (07-Dec-2021, 1:36:38 pm – 1:36:44 pm)

Enter account number:796474739
Enter password:ghfhgf

Invalid credentials

Javadoc Declaration Console Error Log Console x
Server [Java Application] C:\Program Files\Java\jdk-14.0.1\bin\javaw.exe (07-Dec-2021, 1:36:05 pm)
Server is running..
Request received for account number 123456
Request received for account number 456789
Request received for account number 796474739|
```

Conclusion: Implemented a banking server, where user information and methods to retrieve that information are stored. The client can remotely call the methods to check his/her balance. Client server communication is achieved using RMI.