Lab#04

Object: To become familiar with rmi call back using primitive data types as arguments.

Task: Develop an RMI application for that handles the payroll and sends message back to client as an example of callback.

payrollInterface.java

```
Payrollnterface_java - Notepad -  

File Edit Format View Help

import java.rmi.*;
public interface PayrollInterface extends Remote{
public double earning(int id,double hours) throws RemoteException;
}
```

payrollMessageIterface.java

```
PayrollMessageInterface.java - Notepad -  

File Edit Format View Help

import java.rmi.*;

public interface PayrollMessageInterface extends Remote{

public void message(String message) throws RemoteException;

}
```

payroll.java

```
Payroll.java - Notepad
File Edit Format View Help
import java.rmi.*;
import java.rmi.server.*;
import java.net.*;
public class Payroll extends UnicastRemoteObject implements PayrollInterface{
private double pay;
public Payroll () throws RemoteException{}
public double earning(int id, double hours) throws RemoteException{
if(id==1)
pay=10*hours;
else if(id>1)
pay=100*hours;
String clientService="rmi: //localhost /PayrollMessage";
PayrollMessageInterface pm=(PayrollMessageInterface)Naming.lookup(clientService);
pm.message("I am ready to send payroll amount");
}catch(Exception e){e.printStackTrace() ;}
return pay;
```

payrollServer.java

```
File Edit Format View Help

|import java.rmi.*;
| class PayrollServer{
| public static void main(String args[]) {
| try{
| Naming.rebind("rmi. //localhost/PayrollServer" ,new Payroll());
| System.out.println("Payroll Server is ready");
| ) catch(Exception e){
| System.out.println("Payroll Server failed: "+e);
| }
| }
| }
| }
```

payrollMessage.java

```
PayrollMessage.java - Notepad

File Edit Format View Help

import java.rmi.*;
import java.rmi.server.*;
import java.net.*;
public class PayrollMessage extends UnicastRemoteObject implements PayrollMessageInterface{
public PayrollMessage () throws RemoteException{}
public void message(String message) throws RemoteException{
System.out.print("Message from server "+message);
}
}
```

payrollMessageServer.java

```
File Edit Format View Help

|import java.rmi.*;
| class PayrollMessageServer{
| public static void main(String args[]){
| String name="rmi: //localhost/PayrollMessage";
| try{
| Naming.rebind(name,new PayrollMessage());
| String severName="rmi: //localhost/PayrollServer";
| PayrollInterface pr=(PayrollInterface)Naming.lookup(serverName);
| System.out.print("\n"+pr.earning(1,12));
| catch(Exception e){e.printStackTrace();
| }
| }
| }
| }
```

Output:



