

Experiment no 05

Code:

The experiment include 4 program files and they should be store in same folder

1)

```
//one.java
import java.rmi.*;
interface one1 extends Remote
{

    public int palin(String a) throws RemoteException;
}
```

2)

```
//two1.java

import java.rmi.*;
import java.rmi.server.*;
public class two1 extends UnicastRemoteObject implements one1
{
    public two1() throws RemoteException { }
    public int palin(String a) throws RemoteException
    {
        System.out.println("Hello");
        StringBuffer str = new StringBuffer(a);
        String str1 = str.toString();
        System.out.println("Print : " + str1.toString());
        StringBuffer str2 = str.reverse();
        System.out.println("Print : " + str2.toString());
        int b = str1.compareTo(str2.toString());
        System.out.println("Print : " + b);
        if (b == 0)
            return 1;
        else
            return 0;
    }
}
```

3)

```
//rmiclient.java
import java.rmi.*;
```

```

public class rmiclient1
{
    public static void main(String args[]) throws Exception
    {
        try
        {
            String s1 = "rmi://localhost/palin";
            one1 onex = (one1)Naming.lookup(s1);
            int m = onex.palin("madam");
            System.out.println("Print : " + m);
            if (m == 1)
            {
                System.out.println("The given string is a Palindrome");
            }
            else
            {
                System.out.println("The given string is not a Palindrome");
            }
        }
        catch (Exception e)
        {
            System.out.println("Exception" + e);
        }
    }
}

```

4)

```

import java.rmi.registry.*;

public class rmiserver1 {
    public static void main(String args[]) {
        try {
            two1 twox = new two1();

            // Create and bind the RMI registry dynamically
            Registry registry = LocateRegistry.createRegistry(1099);
            registry.rebind("palin", twox);

            System.out.println("Object registered");
        } catch (Exception e) {
            System.out.println("Exception: " + e);
        }
    }
}

```

Commands:

★ First run the server then client using java command

```
PS D:\College_AJP\Experments> javac .\one1.java
PS D:\College_AJP\Experments> javac .\two1.java
PS D:\College_AJP\Experments> javac .\rmiserver1.java
PS D:\College_AJP\Experments> javac .\rmiclient1.java
```

```
PS D:\College_AJP\Experments> java .\rmiserver1.java
Object registered
```

2nd Terminal:

```
PS D:\College_AJP\Experments> javac .\rmiclient1.java
PS D:\College_AJP\Experments> java rmiclient1
```

Output:

```
Print : 0
The given string is not a Palindrome
```

```
Object registered
Hello
Print : madam
Print : madam
Print : 0
Hello
Print : madama
Print : amadam
Print : 12
PS D:\College_AJP\Experments>
```