

PRACTICAL 3

Aim: RMI Program

Practical 3A: A RMI based application program to display current date and time.

Code:

Pramod37_RMIIInterfaceDate.java

```
import java.rmi.*;

public interface Pramod37_RMIIInterfaceDate extends Remote{

    public String printDate() throws Exception;

}
```

Pramod37_RMIServerDate.java

```
import java.rmi.*;
import java.rmi.server.*;
import java.util.*;
import java.util.Date;
import java.text.SimpleDateFormat;

public class Pramod37_RMIServerDate extends UnicastRemoteObject implements
Pramod37_RMIIInterfaceDate{

    public Pramod37_RMIServerDate() throws Exception{

        System.out.println("Pramod Joshi 248637 \n Server is initialised");

    }

    public String printDate() throws Exception{
```

```
        Date d = new Date();

        SimpleDateFormat myFormat = new SimpleDateFormat("dd-MM-yyyy hh:mm:ss");

        String myDate = myFormat.format(d);

        System.out.println("Server: "+myDate);

        return myDate;

    }
```

```
public static void main(String[] args) throws Exception{
    System.out.println("RMIServerDate started.");
    Pramod37_RMIServerDate obj = new Pramod37_RMIServerDate();
    Naming.bind("RMIServerDate",obj);
    System.out.println("Object registered");
}
}
```

Pramod37_RMIClientDate.java

```
import java.rmi.*;
import java.io.*;

public class Pramod37_RMIClientDate{

    public static void main(String[] args) throws Exception{

        System.out.println("Pramod 248637");

        System.out.println("RMIClientDate started");

        Pramod37_RMIIInterfaceDate server =
        (Pramod37_RMIIInterfaceDate)Naming.lookup("RMIServerDate");

        String serverDate = server.printDate();

        System.out.println("Server: "+serverDate);

    }

}
```

Output:

```

C:\Users\admin\Desktop\248637>javac Pramod37_RMIInterfaceDate.java
C:\Users\admin\Desktop\248637>javac Pramod37_RMIServerDate.java
C:\Users\admin\Desktop\248637>javac Pramod37_RMIClientDate.java
C:\Users\admin\Desktop\248637>rmic Pramod37_RMIServerDate
Warning: generation and use of skeletons and static stubs for JRMP
is deprecated. Skeletons are unnecessary, and static stubs have
been superseded by dynamically generated stubs. Users are
encouraged to migrate away from using rmic to generate skeletons and static
stubs. See the documentation for java.rmi.server.UnicastRemoteObject.
C:\Users\admin\Desktop\248637>rmiregistry
-

C:\Users\admin\Desktop\248637>java Pramod37_RMIServerDate
RMIServerDate started.
Pramod Joshi 248637
Server is initialised
Object registered
Server: 28-11-2024 05:10:59
-

C:\Users\admin\Desktop\248637>javac Pramod37_RMIClientDate.java
C:\Users\admin\Desktop\248637>java Pramod37_RMIClientDate
Pramod 248637
RMIClientDate started
Server: 28-11-2024 05:10:59

```

Practical 3B: A RMI based application program that converts digits to words, e.g. 123 will be converted to one two three

Code:

Pramod37_InterConvert.java

```

import java.rmi.*;

public interface Pramod37_InterConvert extends Remote{
    public String convertDigit(String no) throws Exception;
}

```

Pramod37_ServerConvert.java

```

import java.rmi.server.*;

```

```
import java.rmi.*;

public class Pramod37_ServerConvert extends UnicastRemoteObject implements
Pramod37_InterConvert{

public Pramod37_ServerConvert() throws Exception{ }

public String convertDigit(String no) throws Exception{

String str = "";
for(int i = 0;i<no.length();i++){
int p = no.charAt(i);
if (p== 48){str += "zero";}
if (p==49){str+="one";}
if (p==50){str+="two";}
if (p==51){str+="three";}
if (p==52){str+="four";}
if (p==53){str+="five";}
if (p==54){str+="six";}
if (p==55){str+="seven";}
if (p==56){str+="eight";}
if (p==57){str+="nine";}
}
return str;
}

public static void main(String[] args) throws Exception{
Pramod37_ServerConvert s1 = new Pramod37_ServerConvert();
Naming.bind("Wrd",s1);
System.out.println("Pramod 248637\nObject Registered");
}
}
```

Pramod37_ClientConvert.java

```
import java.io.*;
```

```
import java.rmi.*;

public class Pramod37_ClientConvert{

    public static void main(String[] args) throws Exception{

        Pramod37_InterConvert h1 = (Pramod37_InterConvert)Naming.lookup("Wrd");

        BufferedReader br = new BufferedReader(new InputStreamReader(System.in));

        System.out.println("Pramod 248637\nEnter a number: ");

        String no = br.readLine();

        String ans = h1.convertDigit(no);

        System.out.println("The word representation of entered digit is: "+ans);

    }

}
```

Output:

```
C:\Users\admin\Desktop\248637>javac Pramod37_InterConvert.java
C:\Users\admin\Desktop\248637>javac Pramod37_ServerConvert.java
C:\Users\admin\Desktop\248637>javac Pramod37_ClientConvert.java
C:\Users\admin\Desktop\248637>rmic Pramod37_ServerConvert
Warning: generation and use of skeletons and static stubs for JRMP
is deprecated. Skeletons are unnecessary, and static stubs have
been superseded by dynamically generated stubs. Users are
encouraged to migrate away from using rmic to generate skeletons and static
stubs. See the documentation for java.rmi.server.UnicastRemoteObject.
C:\Users\admin\Desktop\248637>rmiregistry
■
```

```
C:\Users\admin\Desktop\248637>javac Pramod37_ServerConvert.java
C:\Users\admin\Desktop\248637>java Pramod37_ServerConvert
Pramod 248637
Object Registered
■
```

```
C:\Users\admin\Desktop\248637>javac Pramod37_ClientConvert.java
C:\Users\admin\Desktop\248637>java Pramod37_ClientConvert
Pramod 248637
Enter a number:
486
The word representation of entered digit is: foureightsix
```