

**UNIVERSITY OF MUMBAI Practical Examination – November 2018 M.C.A Semester – V**  
**LABORATORY L502 [Open Source System for ADC Lab]**

**Remote Method Invocation ( RMI )**

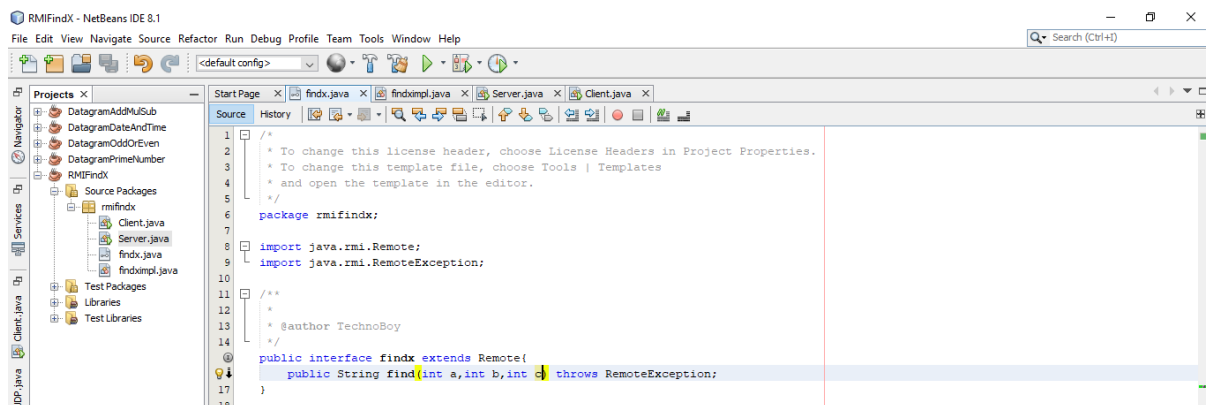
### Remote Method Invocation

The client should provide the values of a, b & c. The server will solve the equation

$(ax^2 + bx + c = 0)$  and will give back the value of x.

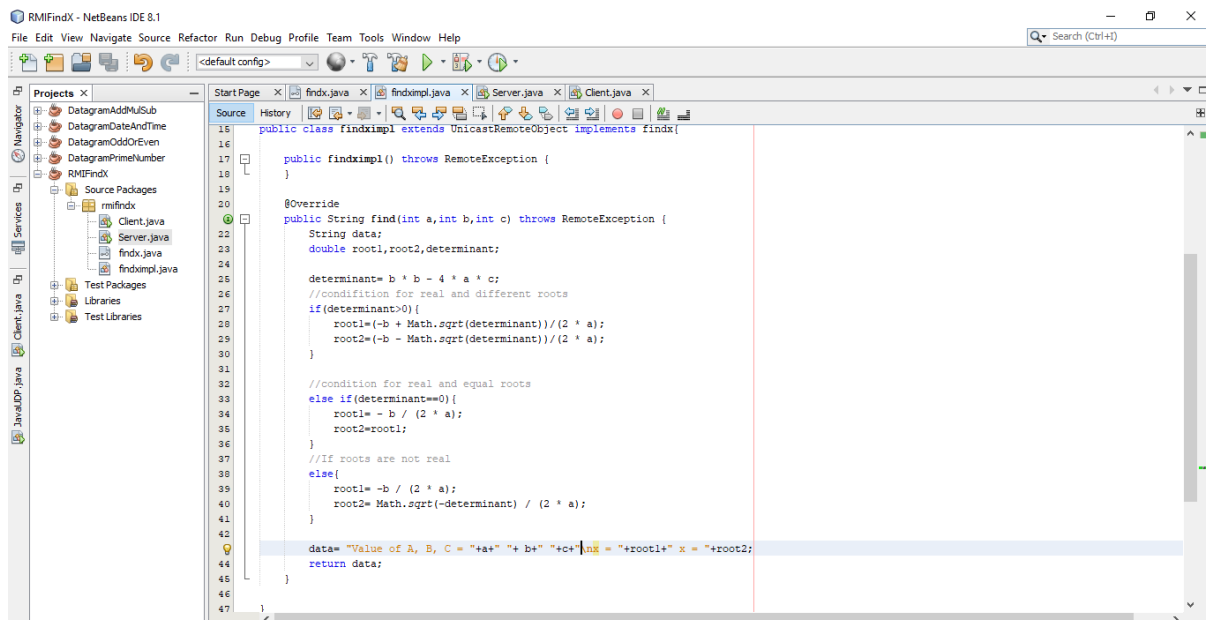
If a = 1, b = 5 and c = 6 then return value will be x = -2 or x = -3.

### Findx.java



```
1  /*
2  * To change this license header, choose License Headers in Project Properties.
3  * To change this template file, choose Tools | Templates
4  * and open the template in the editor.
5  */
6  package rmiFindx;
7
8  import java.rmi.Remote;
9  import java.rmi.RemoteException;
10
11  /**
12   *
13   * @author TechnoBoy
14   */
15  @Remote
16  public interface findx extends Remote{
17      public String find(int a,int b,int c) throws RemoteException;
18  }
```

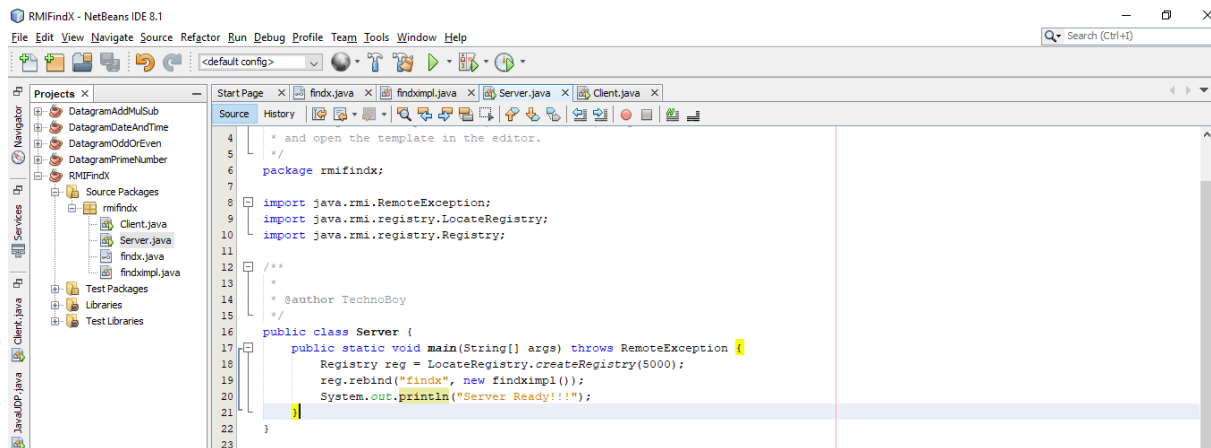
### Findximpl.java



```
15  package rmiFindx;
16
17  import java.rmi.RemoteException;
18  import java.util.ArrayList;
19
20  /**
21   *
22   * @author TechnoBoy
23   */
24  public class findximpl extends UnicastRemoteObject implements findx{
25
26      public findximpl() throws RemoteException {
27      }
28
29      @Override
30      public String find(int a,int b,int c) throws RemoteException {
31          String data;
32          double root1,root2,determinant;
33
34          determinant= b * b - 4 * a * c;
35          //condition for real and different roots
36          if(determinant>0){
37              root1=(-b + Math.sqrt(determinant))/(2 * a);
38              root2=(-b - Math.sqrt(determinant))/(2 * a);
39          }
40          //condition for real and equal roots
41          else if(determinant==0){
42              root1= - b / (2 * a);
43              root2=root1;
44          }
45          //If roots are not real
46          else{
47              root1= -b / (2 * a);
48              root2= Math.sqrt(-determinant) / (2 * a);
49          }
50
51          data= "Value of A, B, C = "+a+" "+b+" "+c+" \nX = "+root1+" x = "+root2;
52          return data;
53      }
54  }
```

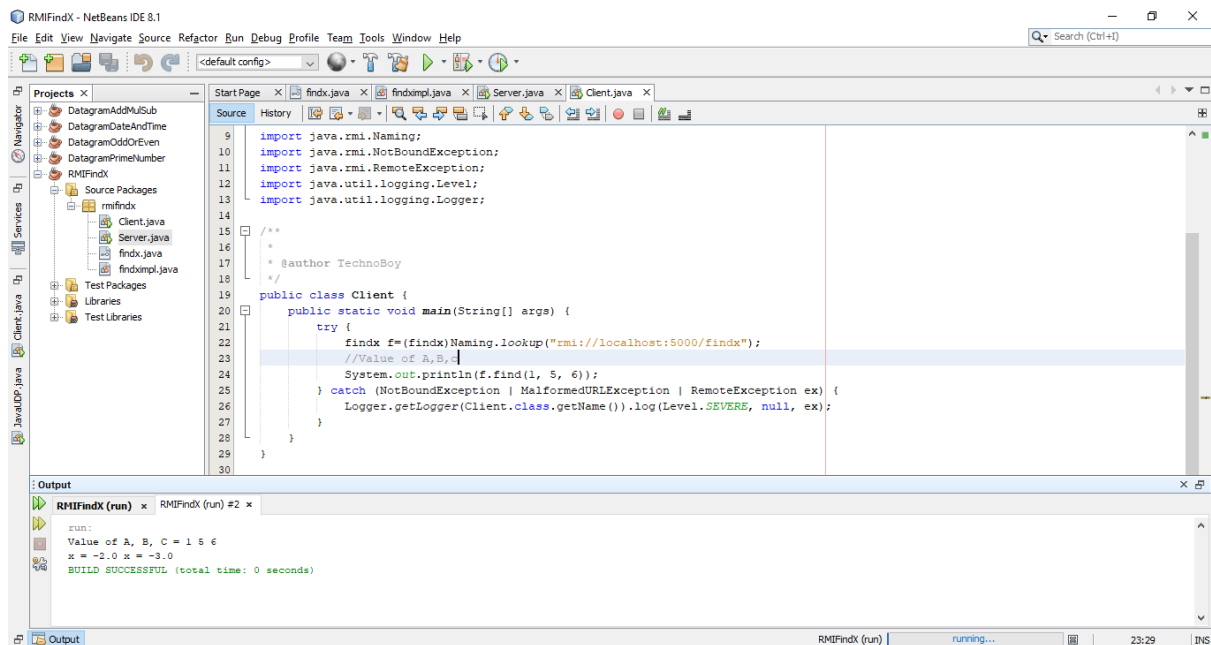
# UNIVERSITY OF MUMBAI Practical Examination – November 2018 M.C.A Semester – V LABORATORY L502 [Open Source System for ADC Lab]

## Server.java



```
1  * and open the template in the editor.
2  */
3
4  package rmifindx;
5
6
7
8  import java.rmi.RemoteException;
9  import java.rmi.registry.LocateRegistry;
10 import java.rmi.registry.Registry;
11
12 /**
13  *
14  * @author TechnoBoy
15  */
16 public class Server {
17     public static void main(String[] args) throws RemoteException {
18         Registry reg = LocateRegistry.createRegistry(5000);
19         reg.rebind("findx", new findximpl());
20         System.out.println("Server Ready!!!");
21     }
22 }
23
```

## Client.java



```
9  import java.rmi.Naming;
10 import java.rmi.NotBoundException;
11 import java.rmi.RemoteException;
12 import java.util.logging.Level;
13 import java.util.logging.Logger;
14
15 /**
16  *
17  * @author TechnoBoy
18  */
19 public class Client {
20     public static void main(String[] args) {
21         try {
22             findx f=(findx)Naming.lookup("rmi://localhost:5000/findx");
23             //Value of A,B,C
24             System.out.println(f.find(1, 5, 6));
25         } catch (NotBoundException | MalformedURLException | RemoteException ex) {
26             Logger.getLogger(Client.class.getName()).log(Level.SEVERE, null, ex);
27         }
28     }
29 }
30
```

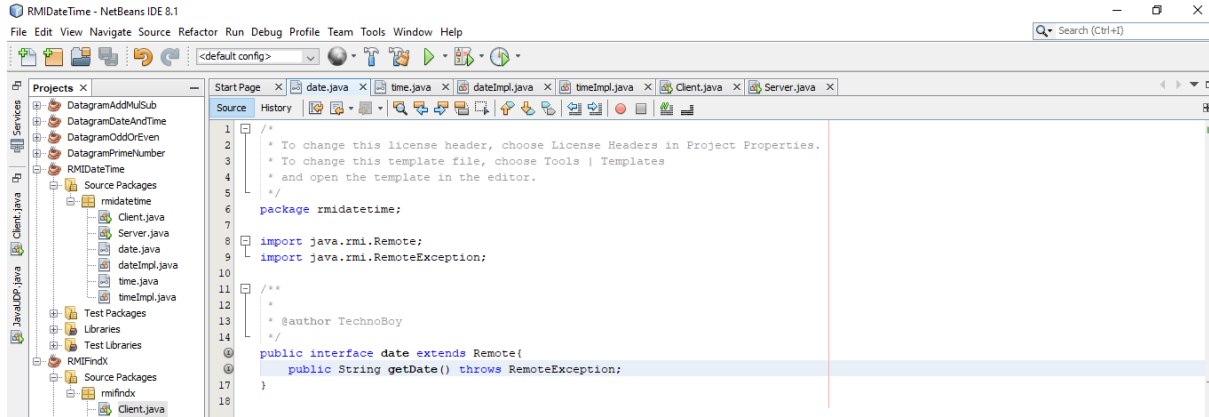
Output

```
run:
Value of A, B, C = 1 5 6
x = -2.0 x = -3.0
BUILD SUCCESSFUL (total time: 0 seconds)
```

**UNIVERSITY OF MUMBAI Practical Examination – November 2018 M.C.A Semester – V**  
**LABORATORY L502 [Open Source System for ADC Lab]**

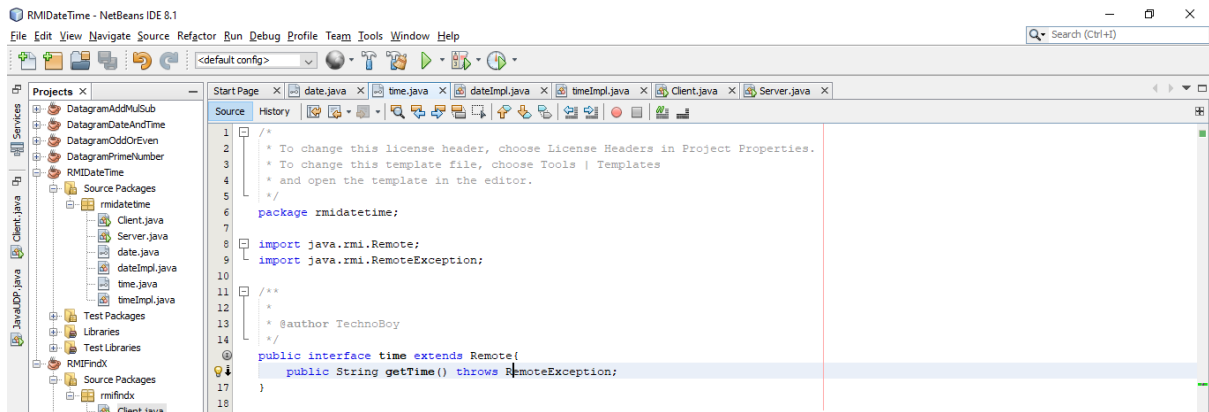
Retrieve time and date function from server to client. This program should display server date and time by implementing RMI.

### Date.java



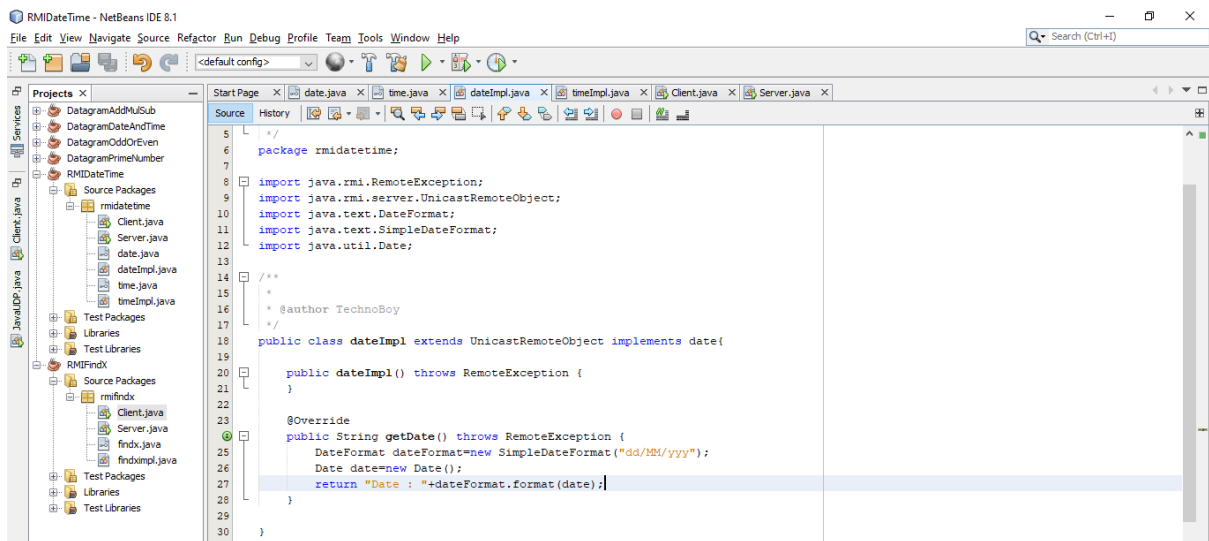
```
1  /*
2  * To change this license header, choose License Headers in Project Properties.
3  * To change this template file, choose Tools | Templates
4  * and open the template in the editor.
5  */
6
7  package rmidatetime;
8
9  import java.rmi.Remote;
10 import java.rmi.RemoteException;
11
12 /**
13  *
14  * @author TechnoBoy
15  */
16 public interface date extends Remote {
17     public String getDate() throws RemoteException;
18 }
```

### Time.java



```
1  /*
2  * To change this license header, choose License Headers in Project Properties.
3  * To change this template file, choose Tools | Templates
4  * and open the template in the editor.
5  */
6
7  package rmidatetime;
8
9  import java.rmi.Remote;
10 import java.rmi.RemoteException;
11
12 /**
13  *
14  * @author TechnoBoy
15  */
16 public interface time extends Remote {
17     public String getTime() throws RemoteException;
18 }
```

### dateImpl.java



```
5  /*
6  * To change this license header, choose License Headers in Project Properties.
7  * To change this template file, choose Tools | Templates
8  * and open the template in the editor.
9  */
10
11 package rmidatetime;
12
13 import java.rmi.RemoteException;
14 import java.rmi.server.UnicastRemoteObject;
15 import java.text.DateFormat;
16 import java.text.SimpleDateFormat;
17 import java.util.Date;
18
19 /**
20  *
21  * @author TechnoBoy
22  */
23 public class dateImpl extends UnicastRemoteObject implements date {
24
25     public dateImpl() throws RemoteException {
26     }
27
28     @Override
29     public String getDate() throws RemoteException {
30         DateFormat dateFormat = new SimpleDateFormat("dd/MM/yyyy");
31         Date date = new Date();
32         return "Date : " + dateFormat.format(date);
33     }
34 }
```

**UNIVERSITY OF MUMBAI Practical Examination – November 2018 M.C.A Semester – V**  
**LABORATORY L502 [Open Source System for ADC Lab]**

### timeImpl.java

```
1 package rmidatetime;
2
3 import java.rmi.RemoteException;
4 import java.rmi.server.UnicastRemoteObject;
5 import java.text.DateFormat;
6 import java.text.SimpleDateFormat;
7 import java.util.Date;
8
9 /**
10  * @author TechnoBoy
11  */
12 public class timeImpl extends UnicastRemoteObject implements time {
13
14     public timeImpl() throws RemoteException {
15     }
16
17     @Override
18     public String getTime() throws RemoteException {
19         DateFormat dateFormat = new SimpleDateFormat("HH:mm:ss");
20         Date date = new Date();
21         return "Time : " + dateFormat.format(date);
22     }
23 }
```

### Server.java

```
1 package rmidatetime;
2
3 import java.rmi.RemoteException;
4 import java.rmi.registry.LocateRegistry;
5 import java.rmi.registry.Registry;
6
7 /**
8  * @author TechnoBoy
9  */
10 public class Server {
11
12     public static void main(String[] args) throws RemoteException {
13         Registry reg = LocateRegistry.createRegistry(5000);
14         reg.rebind("time", new timeImpl());
15         reg.rebind("date", new dateImpl());
16         System.out.println("Server Ready!!!");
17     }
18 }
```

### Client.java

```
1 package rmidatetime;
2
3 import java.rmi.RemoteException;
4 import java.text.DateFormat;
5 import java.text.SimpleDateFormat;
6 import java.util.Date;
7 import java.util.Scanner;
8
9 /**
10  * @author TechnoBoy
11  */
12 public class Client {
13
14     public static void main(String[] args) throws RemoteException, NotBoundException, MalformedURLException {
15         time t = (time) Naming.lookup("rmi://localhost:5000/time");
16         date d = (date) Naming.lookup("rmi://localhost:5000/date");
17         Scanner sc = new Scanner(System.in);
18         while (true) {
19             System.out.println("Enter Operation to Perform (Date,Time)");
20             op = sc.next();
21             if (op.compareToIgnoreCase("time") == 0) {
22                 System.out.println(t.getTime());
23             }
24             else if (op.compareToIgnoreCase("date") == 0) {
25                 System.out.println(d.getDate());
26             }
27             else {
28                 System.out.println("Invalid Selection");
29             }
30         }
31     }
32 }
```

**Output**

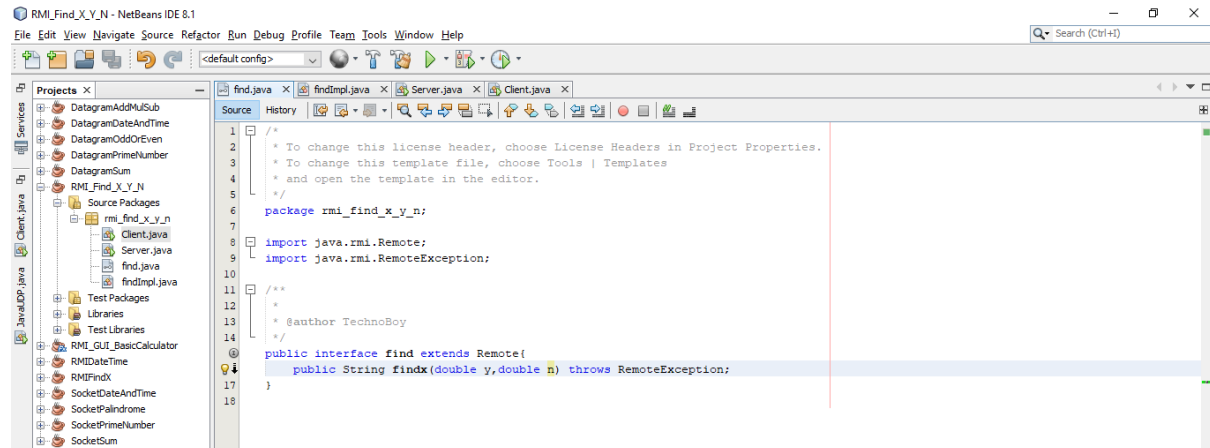
```
RMIDateTime (run) x RMIDateTime (run) #2 x
Enter Operation to Perform (Date,Time)
date
Date : 03/11/2019
Enter Operation to Perform (Date,Time)
time
Time : 21:38:48
Enter Operation to Perform (Date,Time)
abc
Invalid Selection
Enter Operation to Perform (Date,Time)
```

**UNIVERSITY OF MUMBAI Practical Examination – November 2018 M.C.A Semester – V**  
**LABORATORY L502 [Open Source System for ADC Lab]**

Find X, where 'Y' and 'n' values provided. The client should provide equation and values to the server through an interface. The server will solve the expression given by the client. Use RMI

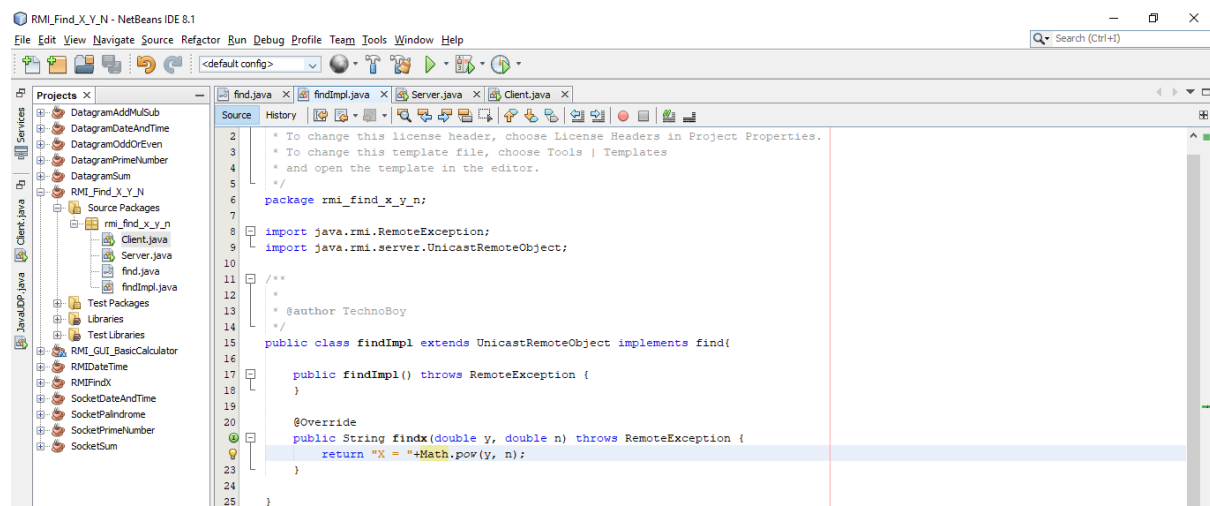
$$X=Y^n$$

### Find.java



```
1  /*
2  * To change this license header, choose License Headers in Project Properties.
3  * To change this template file, choose Tools | Templates
4  * and open the template in the editor.
5  */
6  package rmi_find_x_y_n;
7
8  import java.rmi.Remote;
9  import java.rmi.RemoteException;
10
11  /**
12   *
13   * @author TechnoBoy
14   */
15  public interface find extends Remote{
16      public String findx(double y, double n) throws RemoteException;
17  }
18
```

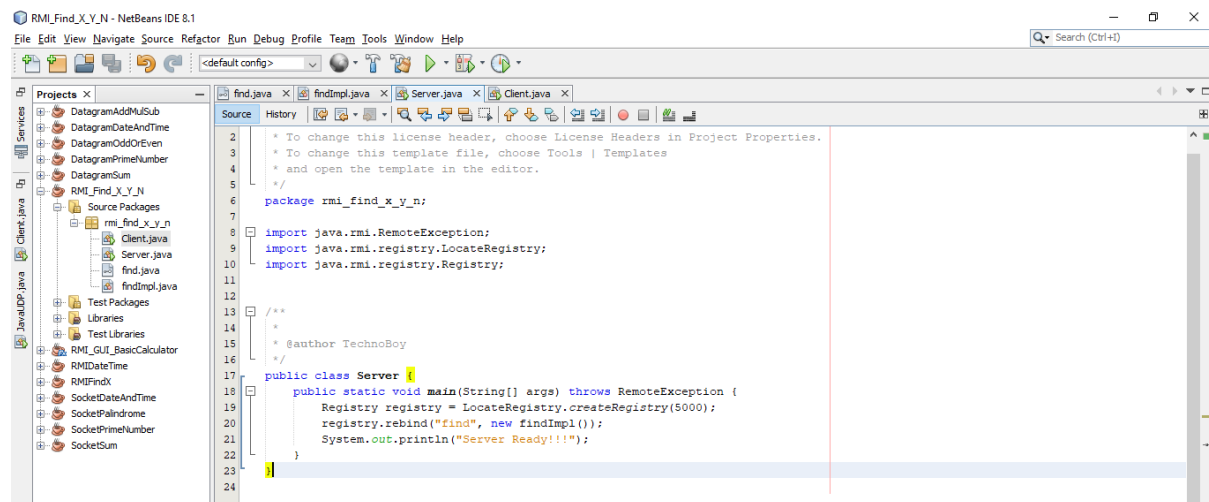
### findImpl.java



```
1  /*
2  * To change this license header, choose License Headers in Project Properties.
3  * To change this template file, choose Tools | Templates
4  * and open the template in the editor.
5  */
6  package rmi_find_x_y_n;
7
8  import java.rmi.RemoteException;
9  import java.rmi.server.UnicastRemoteObject;
10
11  /**
12   *
13   * @author TechnoBoy
14   */
15  public class findImpl extends UnicastRemoteObject implements find{
16
17      public findImpl() throws RemoteException {
18      }
19
20      @Override
21      public String findx(double y, double n) throws RemoteException {
22          return "X = "+Math.pow(y, n);
23      }
24  }
25
```

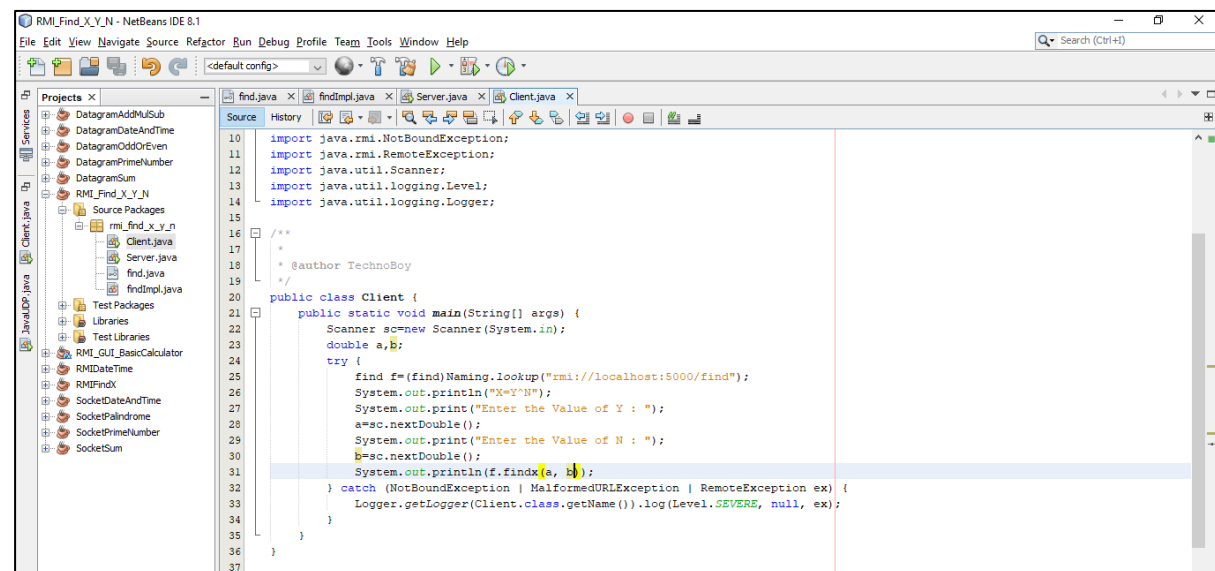
## UNIVERSITY OF MUMBAI Practical Examination – November 2018 M.C.A Semester – V LABORATORY L502 [Open Source System for ADC Lab]

### Server.java



```
1  * To change this license header, choose License Headers in Project Properties.
2  * To change this template file, choose Tools | Templates
3  * and open the template in the editor.
4  */
5  package rmi_find_x_y_n;
6
7
8  import java.rmi.RemoteException;
9  import java.rmi.registry.LocateRegistry;
10 import java.rmi.registry.Registry;
11
12
13 /**
14  *
15  * @author TechnoBoy
16  */
17 public class Server {
18     public static void main(String[] args) throws RemoteException {
19         Registry registry = LocateRegistry.createRegistry(5000);
20         registry.rebind("find", new findImpl());
21         System.out.println("Server Ready!!!");
22     }
23 }
24
```

### Client.java



```
10 import java.rmi.NotBoundException;
11 import java.rmi.RemoteException;
12 import java.util.Scanner;
13 import java.util.logging.Level;
14 import java.util.logging.Logger;
15
16 /**
17  *
18  * @author TechnoBoy
19  */
20 public class Client {
21     public static void main(String[] args) {
22         Scanner sc = new Scanner(System.in);
23         double a, b;
24         try {
25             find f = (find) Naming.lookup("rmi://localhost:5000/find");
26             System.out.println("X=Y*N");
27             System.out.print("Enter the Value of Y : ");
28             a = sc.nextDouble();
29             System.out.print("Enter the Value of N : ");
30             b = sc.nextDouble();
31             System.out.println(f.findX(a, b));
32         } catch (NotBoundException | MalformedURLException | RemoteException ex) {
33             Logger.getLogger(Client.class.getName()).log(Level.SEVERE, null, ex);
34         }
35     }
36 }
37
```