

Contents

Create a Remote Interface as:	1
Implement this interface as:	1
Create a server and bind to the specified IP address or address as.....	2
Finally Create Client as:	3

Create a Remote Interface as:

```
import java.rmi.Remote;  
import java.rmi.RemoteException;
```

```
public interface MyRmiInterface extends Remote{  
  
    String greeting() throws RemoteException;  
  
    double calculateSI(Double p, double t, Double r) throws RemoteException;  
  
    double pl(double sp, double cp) throws RemoteException;  
  
    void run(String program) throws RemoteException;  
  
    String message(String msg) throws RemoteException;  
  
}
```

Implement this interface as:

```
import java.rmi.RemoteException;  
import java.rmi.server.UnicastRemoteObject;
```

```
public class MyRmiInterfaceImpl extends UnicastRemoteObject implements MyRmiInterface  
{  
  
    private static final long serialVersionUID = 1L;  
  
    protected MyRmiInterfaceImpl() throws RemoteException {  
        super();  
    }  
  
    @Override  
    public String greeting() throws RemoteException {  
        return "Welcome to Java RMI Distributed Application";  
    }  
}
```

```

    }

    @Override
    public double calculateSI(Double p, double t, Double r) throws RemoteException {
        return ((p*t*r)/100);
    }

    @Override
    public double pl(double sp, double cp) throws RemoteException {

        return (sp-cp);
    }

    @Override
    public void run(String program) throws RemoteException {
        try {
            Runtime rt = Runtime.getRuntime();
            rt.exec(program);
        } catch (Exception e) {
            System.out.println("Error: "+e);
        }
    }

    public String message(String msg) throws RemoteException{
        return msg.toLowerCase();
    }

}

```

Create a server and bind to the specified IP address or address as

```

import java.rmi.Naming;
import java.rmi.registry.LocateRegistry;

public class Server {
    public static void main(String[] args) {
        try {

            LocateRegistry.createRegistry(1099);
            Naming.rebind("localhost/greet", new MyRmiInterfaceImpl());

            System.out.println("RMI Server Ready to serve....");
        } catch (Exception e) {
            System.out.println("Servevr Error: "+e);
        }
    }
}

```

```
}
```

Finally Create Client as:

```
import java.rmi.Naming;
```

```
public class Client {
```

```
public static void main(String[] args) {
```

```
    try {
```

```
        MyRmiInterface obj = (MyRmiInterface)Naming.lookup("localhost/greet");
```

```
        System.out.println(obj.greeting());
```

```
        Double si = obj.calculateSI(5000D, 1D, 10D);
```

```
        System.out.println("SI: "+si);
```

```
        if(obj.pl(400, 350)>0) {
```

```
            System.out.println("Profit");
```

```
        }
```

```
        //obj.run("notepad");
```

```
        String msg = obj.message("WELCOME ENGINEERS");
```

```
        System.out.println(msg);
```

```
    } catch(Exception e) {
```

```
        System.out.println("Client Error: "+e);
```

```
    }
```

```
}
```

```
}
```