Distributed System Lab Assignment 7

Ashish Verma
20204041
CS - A

Q2.

Step 1: Defining the remote interface

```
import java.rmi.*;
public interface Search extends Remote
{
   // Declaring the method prototype
   public String query(String search) throws RemoteException;
}
```

Step 2: Implementing the remote interface

```
import java.rmi.*;
import java.rmi.server.*;
public class SearchQuery extends UnicastRemoteObject
                    implements Search
{
   // Default constructor to throw RemoteException
   // from its parent constructor
   SearchQuery() throws RemoteException
   {
      super();
   }

   // Implementation of the query interface
   public String query(String search)
                  throws RemoteException
   {
      String result;
      if (search.equals("Reflection in Java"))
         result = "Found";
      else
         result = "Not Found";

      return result;
   }
}
```

Step 3: Creating Stub and Skeleton objects from the implementation class using rmic

at the command prompt:

rmic SearchQuery

Step 4: Start the rmiregistry

Start the registry service by issuing the following command at the command prompt start rmiregistry

Step 5: Create and execute the server application program

```
import java.rmi.*;
import java.rmi.registry.*;
public class SearchServer
{
   public static void main(String args[])
   {
      try
      {
         // Create an object of the interface
         // implementation class
         Search obj = new SearchQuery();

         // rmiregistry within the server JVM with
         // port number 1900
         LocateRegistry.createRegistry(1900);

         // Binds the remote object by the name
         // geeksforgeeks
         Naming.rebind("rmi://localhost:1900"+
                  "/geeksforgeeks",obj);
      }
      catch(Exception ae)
      {
         System.out.println(ae);
      }
   }
}
```

Step 6: Create and execute the client application program

```
// Java program for client application
import java.rmi.*;
public class ClientRequest
{
   public static void main(String args[])
   {
      String answer,value="Reflection in Java";
      try
      {
```

```
            // lookup method to find reference of remote object
            Search access =
               (Search)Naming.lookup("rmi://localhost:1900"+
                                  "/geeksforgeeks");
            answer = access.query(value);
            System.out.println("Article on " + value +
                        " " + answer+" at GeeksforGeeks");
         }
         catch(Exception ae)
         {
            System.out.println(ae);
         }
      }
}
```