```
//Addition.idl

/**
 *
 * @author imed
 */
module AdditionApp
{
  interface Addition
  {
    double add(in double a, in double b);
  double subtract(in double a, in double b);
  double multiply(in double a, in double b);
  double divide(in double a, in double b);
    };
};
```

```java
//AdditionObj.java

    /**
     *
     * @author imed
     */
    import AdditionApp.*;
    import org.omg.CosNaming.*;
    import org.omg.CosNaming.NamingContextPackage.*;
    import org.omg.CORBA.*;
    import org.omg.PortableServer.*;
    import org.omg.PortableServer.POA;
    import java.util.Properties;

    class AdditionObj extends AdditionPOA {
      private ORB orb;

      public void setORB(ORB orb_val) {
        orb = orb_val;
      }

      public double add(double a, double b) {
    return a + b;
  }

  public double subtract(double a, double b) {
    return a - b;
  }

  public double multiply(double a, double b) {
    return a * b;
  }

  public double divide(double a, double b) {
    return a/ b;
  }

      // implement shutdown() method
      public void shutdown() {
        orb.shutdown(false);
      }
    }
```

```java
//StartClient.java

    /**
     *
     * @author imed
     */
    import AdditionApp.*;

    import org.omg.CosNaming.*;
    import org.omg.CosNaming.NamingContextPackage.*;
    import org.omg.CORBA.*;
    import java.io.*;
    import java.util.*;

    public class StartClient {

        /**
         * @param args the command line arguments
         */
        public static void main(String[] args) {
          try {
          ORB orb = ORB.init(args, null);
          org.omg.CORBA.Object objRef =
orb.resolve_initial_references("NameService");
          NamingContextExt ncRef = NamingContextExtHelper.narrow(objRef);
          Addition addobj = (Addition)
AdditionHelper.narrow(ncRef.resolve_str("ABC"));

                  System.out.println("Welcome to the Calculator system:");
            Scanner c=new Scanner(System.in);
            for(;;){
                System.out.println("Enter Your Choice ");
                 System.out.println("1 for Addition");
                 System.out.println("2 for Subtraction");
                 System.out.println("3 for Multiplication");
                 System.out.println("4 for Division");
                System.out.println("5 for Exit");
                int choice=c.nextInt();
                if(choice==5){
                break;
                    }
                System.out.println("Enter a:");
                    double a=c.nextDouble();
                 System.out.println("Enter b:");
                 double b=c.nextDouble();
                 switch(choice){
                    case 1:
                      System.out.println((int)a + " + " + (int)b + " =
" + (int)addobj.add(a, b));
                        break;
                    case 2:
                      System.out.println((int)a + " - " + (int)b + " =
" + (int)addobj.subtract(a, b));
                        break;
                    case 3:
                      System.out.println((int)a + " * " + (int)b + " =
" + (int)addobj.multiply(a, b));
                        break;
                    case 4:
```

```java
                    System.out.println((int)a + " / " + (int)b + " =
" + (int)addobj.divide(a, b));
                    break;
                    default:
                        System.out.println("Invalid Choice");
                }

            }
        }

        catch (Exception e) {
            System.out.println("Hello Client exception: " + e);
        e.printStackTrace();
        }

    }

}
```

```java
//StartServer.java

    /**
     *
     * @author imed
     */
    import AdditionApp.*;

    import org.omg.CosNaming.*;
    import org.omg.CosNaming.NamingContextPackage.*;
    import org.omg.CORBA.*;
    import org.omg.PortableServer.*;
    import org.omg.PortableServer.POA;
    import java.util.Properties;

    public class StartServer {

      public static void main(String args[]) {
        try{
          // create and initialize the ORB //// get reference to rootpoa
& activate the POAManager
          ORB orb = ORB.init(args, null);
          POA rootpoa =
POAHelper.narrow(orb.resolve_initial_references("RootPOA"));
          rootpoa.the_POAManager().activate();

          // create servant and register it with the ORB
          AdditionObj addobj = new AdditionObj();
          addobj.setORB(orb);

          // get object reference from the servant
          org.omg.CORBA.Object ref =
rootpoa.servant_to_reference(addobj);
          Addition href = AdditionHelper.narrow(ref);

          org.omg.CORBA.Object objRef =
orb.resolve_initial_references("NameService");
          NamingContextExt ncRef = NamingContextExtHelper.narrow(objRef);

          NameComponent path[] = ncRef.to_name( "ABC" );
          ncRef.rebind(path, href);

          System.out.println("Calculator Server is ready and waiting
...");

          // wait for invocations from clients
          for (;;){
        orb.run();
          }
        }

          catch (Exception e) {
            System.err.println("ERROR: " + e);
            e.printStackTrace(System.out);
          }

          System.out.println("Server Exiting ...");

      }
    }
```