DS Lab 2

Code:

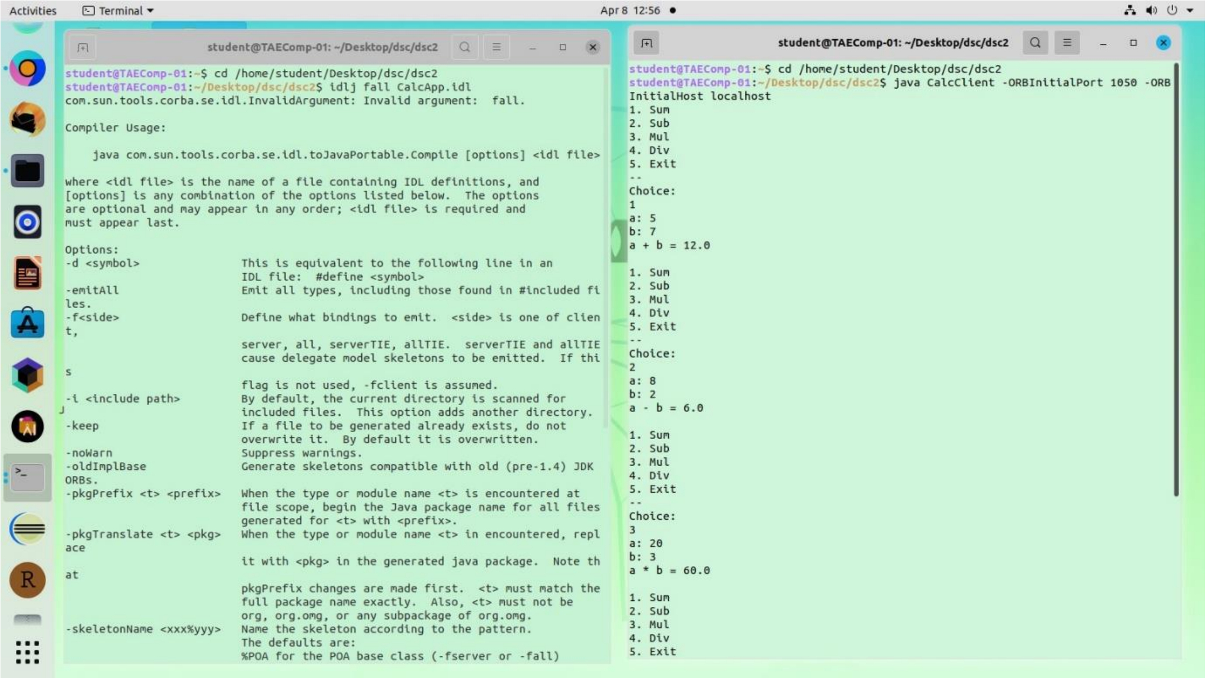
CalcServer.java

```
import CalcApp.*;

import CalcApp.CalcPackage.DivisionByZero;

import org.omg.CosNaming.*;
import org.omg.CosNaming.NamingContextPackage.*;
import org.omg.CORBA.*;
import org.omg.PortableServer.*;

import java.util.Properties;

class CalcImpl extends CalcPOA {
    private ORB orb;

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

    @Override
    public float sum(float a, float b) {
        return a + b;
    }

    @Override
    public float div(float a, float b) throws DivisionByZero {
        if (b == 0) {
            throw new CalcApp.CalcPackage.DivisionByZero();
        } else {
            return a / b;
        }
    }

    @Override
    public float mul(float a, float b) {
        return a * b;
    }

    @Override
    public float sub(float a, float b) {
        return a - b;
    }
}

public class CalcServer {
    public static void main(String args[]) {
        try {
            // create and initialize the ORB
            ORB orb = ORB.init(args, null);

            // get reference to rootpoa & activate the POAManager
            POA rootpoa =
POAHelper.narrow(orb.resolve_initial_references("RootPOA"));
            rootpoa.the_POAManager().activate();
            // create servant and register it with the ORB
            CalcImpl calcImpl = new CalcImpl();
            calcImpl.setORB(orb);

            // get object reference from the servant
            org.omg.CORBA.Object ref =
rootpoa.servant_to_reference(calcImpl);
            Calc href = CalcHelper.narrow(ref);

            // get the root naming context
            // NameService invokes the name service
            org.omg.CORBA.Object objRef =
orb.resolve_initial_references("NameService");
            // Use NamingContextExt which is part of the Interoperable
            // Naming Service (INS) specification.
            NamingContextExt ncRef =
NamingContextExtHelper.narrow(objRef);

            // bind the Object Reference in Naming
            String name = "Calc";
            NameComponent path[] = ncRef.to_name(name);
            ncRef.rebind(path, href);

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

            // wait for invocations from clients
            orb.run();
        } catch (Exception e) {
            System.err.println("ERROR: " + e);
            e.printStackTrace(System.out);
        }
        System.out.println("Server exiting...");
    }
}
```

CalcClient.java

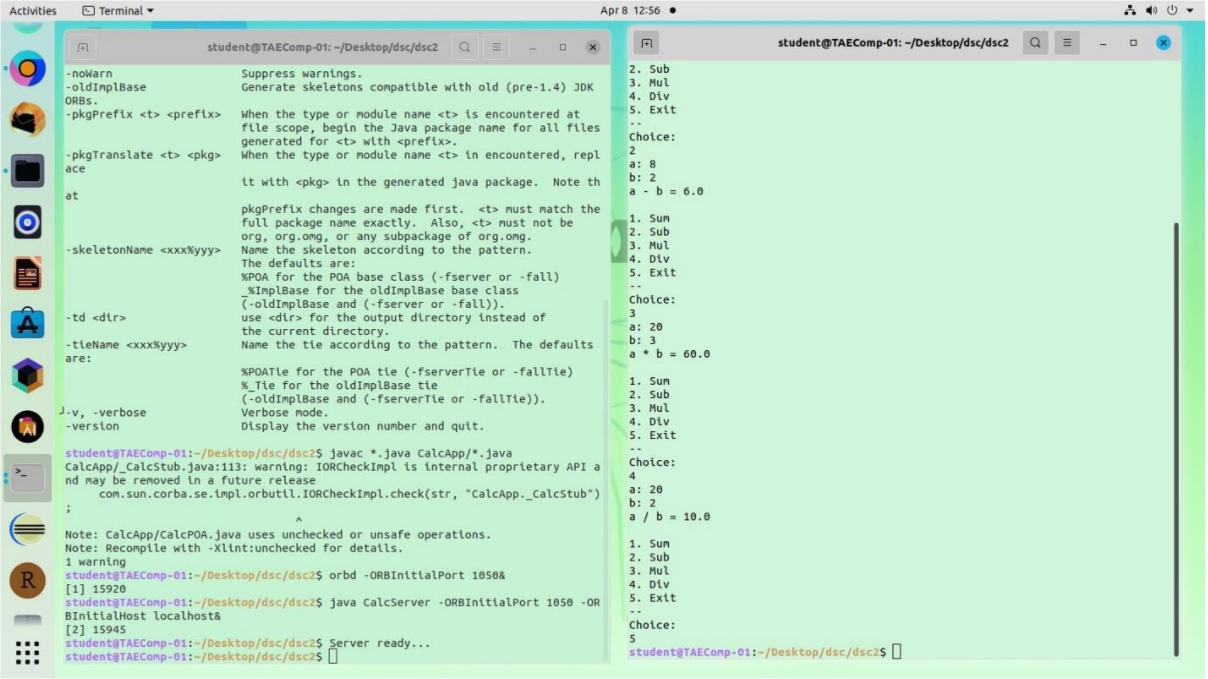
```
import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;

import CalcApp.*;
import CalcApp.CalcPackage.DivisionByZero;

import org.omg.CosNaming.*;
import org.omg.CosNaming.NamingContextPackage.*;
import org.omg.CORBA.*;
import static java.lang.System.out;

public class CalcClient {
    static Calc calcImpl;
    static BufferedReader br = new BufferedReader(new
InputStreamReader(System.in));

    public static void main(String args[]) {
        try {
            // create and initialize the ORB
            ORB orb = ORB.init(args, null);
            // get the root naming context
            org.omg.CORBA.Object objRef =
orb.resolve_initial_references("NameService");

            // Use NamingContextExt instead of NamingContext.
            // This is part of the Interoperable naming Service.
            NamingContextExt ncRef =
NamingContextExtHelper.narrow(objRef);

            // resolve the Object Reference in Naming
            String name = "Calc";
            calcImpl = CalcHelper.narrow(ncRef.resolve_str(name));

            while (true) {
                out.println("1. Sum");
                out.println("2. Sub");
                out.println("3. Mul");
                out.println("4. Div");
                out.println("5. Exit");
                out.println("--");
                out.println("Choice: ");

                try {
                    String opt = br.readLine();
                    if (opt.equals("5")) {
                        break;
                    } else if (opt.equals("1")) {
                        out.println("a + b = " +
calcImpl.sum(getFloat("a"), getFloat("b")));
                    } else if (opt.equals("2")) {
                        out.println("a - b = " +
calcImpl.sub(getFloat("a"), getFloat("b")));
                    } else if (opt.equals("3")) {
                        out.println("a * b = " +
calcImpl.mul(getFloat("a"), getFloat("b")));
                    } else if (opt.equals("4")) {
                        try {
                            out.println("a / b = " +
calcImpl.div(getFloat("a"), getFloat("b")));
                        } catch (DivisionByZero de) {
                            out.println("Division by zero!!!");
                        }
                    }
                } catch (Exception e) {
                    out.println("===");
                    out.println("Error with numbers");
                    out.println("===");
                }
                out.println();
            }
        } catch (Exception e) {
            System.out.println("ERROR: " + e);
            e.printStackTrace(System.out);
        }
    }

    static float getFloat(String number) throws Exception {
        out.print(number + ": ");
        return Float.parseFloat(br.readLine());
    }
}
```

Output:-


Activities
Terminal
Apr 8 12:56
student@TAEComp-01: ~/Desktop/dsc/dsc2
student@TAEComp-01: ~/Desktop/dsc/dsc2

```
student@TAEComp-01:~$ cd /home/student/Desktop/dsc/dsc2
student@TAEComp-01:~/Desktop/dsc/dsc2$ idlj fall CalcApp.idl
com.sun.tools.corba.se.idl.InvalidArgument: Invalid argument:  fall.

Compiler Usage:

    java com.sun.tools.corba.se.idl.toJavaPortable.Compile [options] <idl file>

where <idl file> is the name of a file containing IDL definitions, and
[options] is any combination of the options listed below.  The options
are optional and may appear in any order; <idl file> is required and
must appear last.

Options:
-d <symbol>               This is equivalent to the following line in an
                          IDL file:  #define <symbol>
-emitAll                  Emit all types, including those found in #included fi
les.
-f<side>                  Define what bindings to emit.  <side> is one of clien
t,
                          server, all, serverTIE, allTIE.  serverTIE and allTIE
                          cause delegate model skeletons to be emitted.  If thi
s
                          flag is not used, -fclient is assumed.
-i <include path>         By default, the current directory is scanned for
                          included files.  This option adds another directory.
-keep                     If a file to be generated already exists, do not
                          overwrite it.  By default it is overwritten.
-noWarn                   Suppress warnings.
-oldImplBase              Generate skeletons compatible with old (pre-1.4) JDK
ORBs.
-pkgPrefix <t> <prefix>   When the type or module name <t> is encountered at
                          file scope, begin the Java package name for all files
                          generated for <t> with <prefix>.
-pkgTranslate <t> <pkg>   When the type or module name <t> in encountered, repl
ace
                          it with <pkg> in the generated java package.  Note th
at
                          pkgPrefix changes are made first.  <t> must match the
                          full package name exactly.  Also, <t> must not be
                          org, org.omg, or any subpackage of org.omg.
-skeletonName <xxx%yyy>   Name the skeleton according to the pattern.
                          The defaults are:
                          %POA for the POA base class (-fserver or -fall)
```

```
student@TAEComp-01:~$ cd /home/student/Desktop/dsc/dsc2
student@TAEComp-01:~/Desktop/dsc/dsc2$ java CalcClient -ORBInitialPort 1050 -ORB
InitialHost localhost
1. Sum
2. Sub
3. Mul
4. Div
5. Exit
--
Choice:
1
a: 5
b: 7
a + b = 12.0

1. Sum
2. Sub
3. Mul
4. Div
5. Exit
--
Choice:
2
a: 8
b: 2
a - b = 6.0

1. Sum
2. Sub
3. Mul
4. Div
5. Exit
--
Choice:
3
a: 20
b: 3
a * b = 60.0

1. Sum
2. Sub
3. Mul
4. Div
5. Exit
```


Activities
Terminal
Apr 8 12:56
student@TAEComp-01: ~/Desktop/dsc/dsc2
student@TAEComp-01: ~/Desktop/dsc/dsc2

```
-noWarn                   Suppress warnings.
-oldImplBase              Generate skeletons compatible with old (pre-1.4) JDK
ORBs.
-pkgPrefix <t> <prefix>   When the type or module name <t> is encountered at
                          file scope, begin the Java package name for all files
                          generated for <t> with <prefix>.
-pkgTranslate <t> <pkg>   When the type or module name <t> in encountered, repl
ace
                          it with <pkg> in the generated java package.  Note th
at
                          pkgPrefix changes are made first.  <t> must match the
                          full package name exactly.  Also, <t> must not be
                          org, org.omg, or any subpackage of org.omg.
-skeletonName <xxx%yyy>   Name the skeleton according to the pattern.
                          The defaults are:
                          %POA for the POA base class (-fserver or -fall)
                          _%ImplBase for the oldImplBase base class
                          (-oldImplBase and (-fserver or -fall)).
-td <dir>                 use <dir> for the output directory instead of
                          the current directory.
-tieName <xxx%yyy>        Name the tie according to the pattern.  The defaults
are:
                          %POATie for the POA tie (-fserverTie or -fallTie)
                          %_Tie for the oldImplBase tie
                          (-oldImplBase and (-fserverTie or -fallTie)).
-v, -verbose              Verbose mode.
-version                  Display the version number and quit.

student@TAEComp-01:~/Desktop/dsc/dsc2$ javac *.java CalcApp/*.java
CalcApp/_CalcStub.java:113: warning: IORCheckImpl is internal proprietary API a
nd may be removed in a future release
     com.sun.corba.se.impl.orbutil.IORCheckImpl.check(str, "CalcApp._CalcStub")
;
                                        ^
Note: CalcApp/CalcPOA.java uses unchecked or unsafe operations.
Note: Recompile with -Xlint:unchecked for details.
1 warning
student@TAEComp-01:~/Desktop/dsc/dsc2$ orbd -ORBInitialPort 1050&
[1] 15920
student@TAEComp-01:~/Desktop/dsc/dsc2$ java CalcServer -ORBInitialPort 1050 -OR
BInitialHost localhost&
[2] 15945
student@TAEComp-01:~/Desktop/dsc/dsc2$ Server ready...
student@TAEComp-01:~/Desktop/dsc/dsc2$ 
```

```
2. Sub
3. Mul
4. Div
5. Exit
--
Choice:
2
a: 8
b: 2
a - b = 6.0

1. Sum
2. Sub
3. Mul
4. Div
5. Exit
--
Choice:
3
a: 20
b: 3
a * b = 60.0

1. Sum
2. Sub
3. Mul
4. Div
5. Exit
--
Choice:
4
a: 20
b: 2
a / b = 10.0

1. Sum
2. Sub
3. Mul
4. Div
5. Exit
--
Choice:
5
student@TAEComp-01:~/Desktop/dsc/dsc2$ 
```