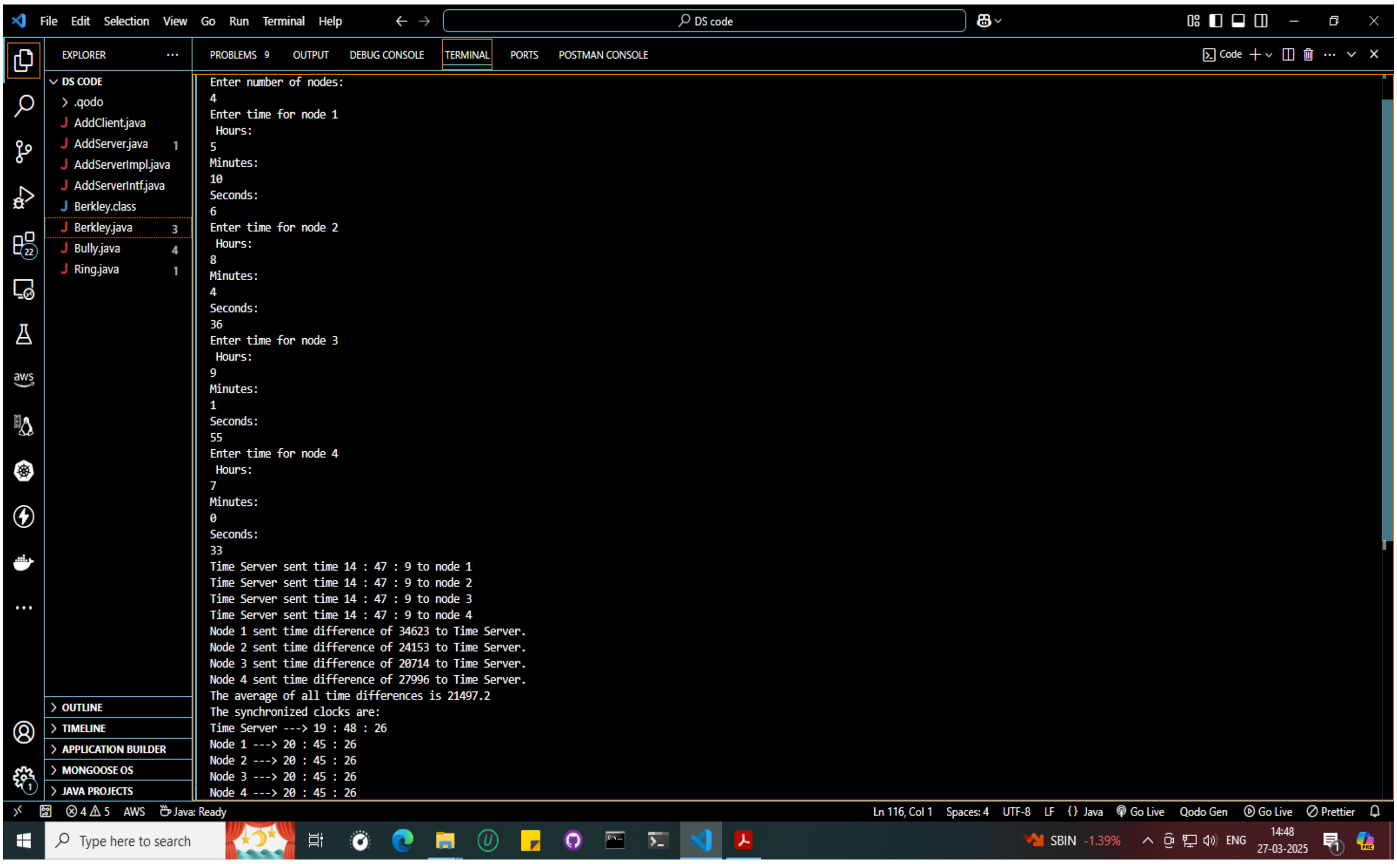
Assign4

```
//Program to demonstrate Berkeley clock synchronization algorithm
import java.io.*;
import java.util.*;
public class Berkley
{
float diff(int h, int m, int s, int nh, int nm, int ns){
int dh = h-nh;
int dm = m-nm;
int ds = s-ns;
int diff = (dh*60*60)+(dm*60)+ds;
return diff;
}
float average(float diff[], int n){
int sum=0;
for(int i=0; i<n; i++)
{
sum+=diff[i];
}
float average = (float)sum/(n+1);
System.out.println("The average of all time differences is "+average);
return average;
}
void sync(float diff[], int n, int h, int m, int s, int nh[], int nm[], int ns[], float average)
{
for(int i=0;i<n;i++)
{
diff[i]+=average;
int dh=(int)diff[i]/(60*60);
diff[i]%=(60*60);

int dm=(int)diff[i]/60;
diff[i]%=60;
int ds=(int)diff[i];nh[i]+=dh;
if(nh[i]>23)
{
nh[i]%=24;
}
nm[i]+=dm;
if(nm[i]>59)
```

```
{
nh[i]++;
nm[i]%=60;
}
ns[i]+=ds;
if(ns[i]>59)
{
nm[i]++;
ns[i]%=60;
}
if(ns[i]<0)
{
nm[i]--;
ns[i]+=60;
}
}
h+=(int)(average/(60*60));
if(h>23)
{ h%=24;
}
m+=(int)(average/(60*60*60));
if(m>59)
{
h++;
m%=60;
}
s+=(int)(average%(60*60*60));
if(s>59)
{
m++;s%=60;
}
if(s<0)
{
m--;
s+=60;
}
System.out.println("The synchronized clocks are:\nTime Server ---> "+h+" : "+m+" : "+s);
for(int i=0;i<n;i++)
{
System.out.println("Node "+(i+1)+" ---> "+nh[i]+" : "+nm[i]+" : "+ns[i]);
```

```
}
}
public static void main(String[] args) throws IOException {
Berkley b = new Berkley();

Date date = new Date();
BufferedReader obj = new BufferedReader(new InputStreamReader(System.in));
System.out.println("Enter number of nodes:");
int n = Integer.parseInt(obj.readLine()); int h = date.getHours();
int m = date.getMinutes();
int s = date.getSeconds();
int nh[] = new int[n];
int nm[] = new int[n];
int ns[] = new int[n];
for(int i=0; i<n; i++)
{
System.out.println("Enter time for node "+(i+1)+"\n Hours:");
nh[i]=Integer.parseInt(obj.readLine());
System.out.println("Minutes:");
nm[i]=Integer.parseInt(obj.readLine());
System.out.println("Seconds:");
ns[i]=Integer.parseInt(obj.readLine());
}
for(int i=0; i<n; i++)
{
System.out.println("Time Server sent time "+h+" : "+m+" : "+s+" to node "+(i+1));
}float diff[] = new float[n];
for(int i=0;i<n;i++)
{
diff[i] = b.diff(h,m,s,nh[i],nm[i],ns[i]);
System.out.println("Node "+(i+1)+" sent time difference of "+(int)diff[i]+" to Time Server.");
}
float average = b.average(diff,n);
b.sync(diff, n, h, m, s, nh, nm, ns, average);
}
}
```



```
Enter number of nodes:
4
Enter time for node 1
 Hours:
5
Minutes:
10
Seconds:
6
Enter time for node 2
 Hours:
8
Minutes:
4
Seconds:
36
Enter time for node 3
 Hours:
9
Minutes:
1
Seconds:
55
Enter time for node 4
 Hours:
7
Minutes:
0
Seconds:
33
Time Server sent time 14 : 47 : 9 to node 1
Time Server sent time 14 : 47 : 9 to node 2
Time Server sent time 14 : 47 : 9 to node 3
Time Server sent time 14 : 47 : 9 to node 4
Node 1 sent time difference of 34623 to Time Server.
Node 2 sent time difference of 24153 to Time Server.
Node 3 sent time difference of 20714 to Time Server.
Node 4 sent time difference of 27996 to Time Server.
The average of all time differences is 21497.2
The synchronized clocks are:
Time Server ---> 19 : 48 : 26
Node 1 ---> 20 : 45 : 26
Node 2 ---> 20 : 45 : 26
Node 3 ---> 20 : 45 : 26
Node 4 ---> 20 : 45 : 26
```

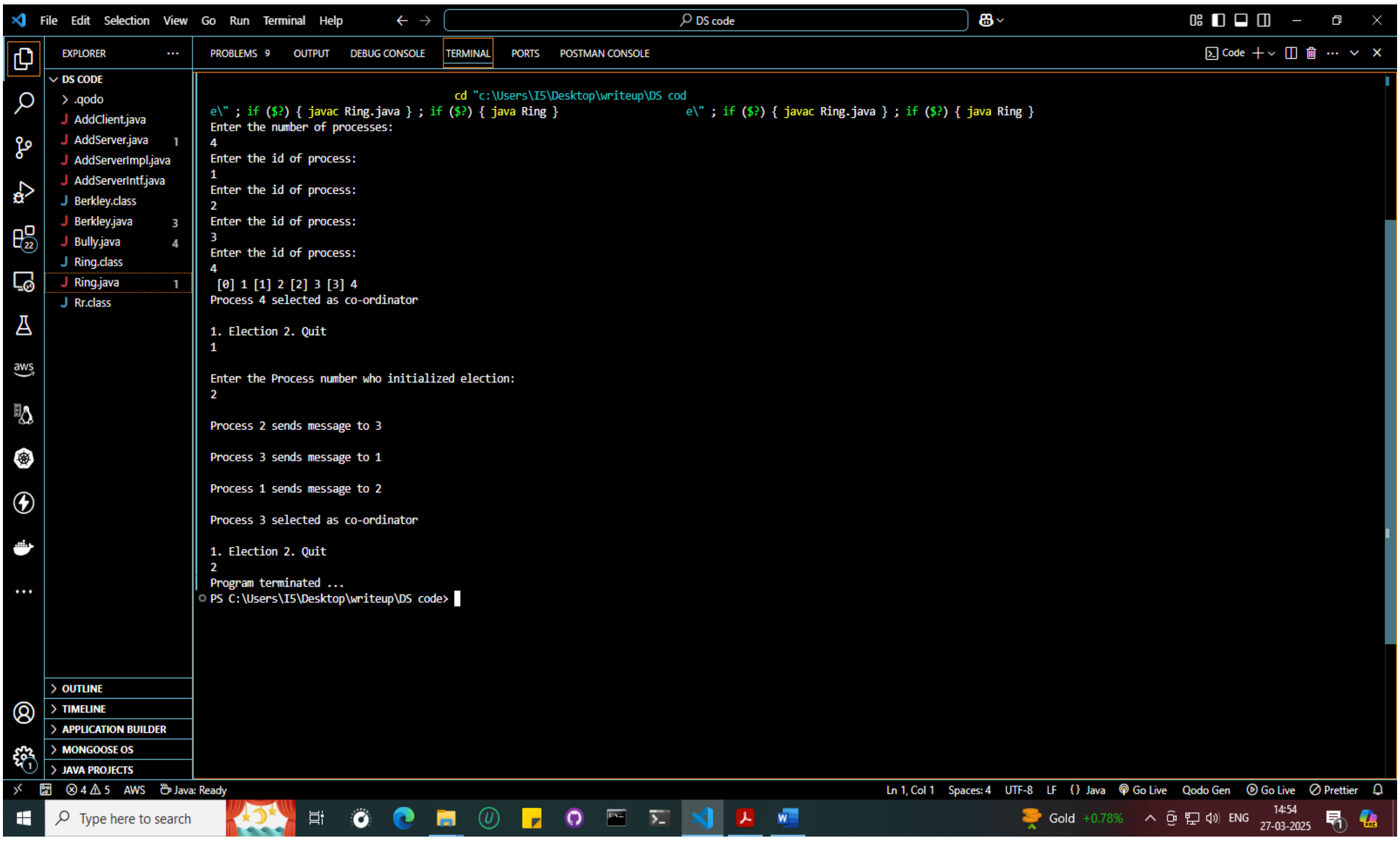
Assign6

```
//Ring.java
import java.util.Scanner;
import java.util.InputMismatchException;
public class Ring {
    public static void main(String[] args) {
        int temp, i, j;
        Rr proc[] = new Rr[10];
        for (i = 0; i < proc.length; i++) proc[i] = new Rr();
        Scanner in = new Scanner(System.in);
        int num = 0;
        while (true) {
            try {
                System.out.println("Enter the number of processes: ");
                num = in.nextInt();
                if (num <= 0 || num > 10) {
                    System.out.println("Please enter a number between 1 and 10.");
                    continue;
                }
                break; // Exit the loop if input is valid
            } catch (InputMismatchException e) {
                System.out.println("Invalid input. Please enter an integer.");
                in.next(); // Clear the invalid input
            }
        }
        for (i = 0; i < num; i++) {
            proc[i].index = i;
            while (true) {
                try {
                    System.out.println("Enter the id of process: ");
                    proc[i].id = in.nextInt();
                    break; // Exit the loop if input is valid
                } catch (InputMismatchException e) {
                    System.out.println("Invalid input. Please enter an integer.");
                    in.next(); // Clear the invalid input
                }
            }
            proc[i].state = "active";
            proc[i].f = 0;
        }
```

```
for (i = 0; i < num - 1; i++) {
    for (j = 0; j < num - 1; j++) {
        if (proc[j].id > proc[j + 1].id) {
            temp = proc[j].id;
            proc[j].id = proc[j + 1].id;
            proc[j + 1].id = temp;
        }
    }
}
for (i = 0; i < num; i++) {
    System.out.print(" [" + i + "]" + " " + proc[i].id);
}
int init;
int ch;
int temp1;
int temp2;
int arr[] = new int[10];
proc[num - 1].state = "inactive";
System.out.println("\nProcess " + proc[num - 1].id + " selected as co-ordinator");
while (true) {
    System.out.println("\n1. Election 2. Quit ");
    while (true) {
        try {
            ch = in.nextInt();
            break; // Exit the loop if input is valid
        } catch (InputMismatchException e) {
            System.out.println("Invalid input. Please enter an integer.");
            in.next(); // Clear the invalid input
        }
    }
    for (i = 0; i < num; i++) {
        proc[i].f = 0;
    }
    switch (ch) {
        case 1:
            System.out.println("\nEnter the Process number who initialized election: ");
            while (true) {
                try {
                    init = in.nextInt();
                    init--;
```

```
                            if (init < 0 || init >= num) {
                                System.out.println("Invalid process number. Please try
again.");                                      continue;
                            }
                            break; // Exit the loop if input is valid
                        } catch (InputMismatchException e) {
                            System.out.println("Invalid input. Please enter an integer.");
                            in.next(); // Clear the invalid input
                        }
                    }
                    temp2 = init;
                    temp1 = init + 1;
                    i = 0;
                    while (temp2 != temp1) {
                        if ("active".equals(proc[temp1].state) && proc[temp1].f == 0) {
                            System.out.println("\nProcess " + proc[init].id + " sends message to "
+proc[temp1].id);
                            proc[temp1].f = 1;
                            init = temp1;
                            arr[i] = proc[temp1].id;
                            i++;
                        }
                        if (temp1 == num) {
                            temp1 = 0;
                        } else {
                            temp1++;
                        }
                    }
                    System.out.println("\nProcess " + proc[init].id + " sends message to " + proc[temp1].id);
                    arr[i] = proc[temp1].id;
                    i++;
                    int max = -1;
                    for (j = 0; j < i; j++) {
                        if (max < arr[j]) {
                            max = arr[j];
                        }
                    }
                    System.out.println("\nProcess " + max + " selected as co-ordinator");
                    for (i = 0; i < num; i++) {
                        if (proc[i].id == max) {
```

```
                    proc[i].state = "inactive";
                  }
                }
                break;
            case 2:
                System.out.println("Program terminated ...");
                return;
            default:
                System.out.println("\nInvalid response \n");
                break;
            }
        }
    }
}
class Rr {
    public int index;
    public int id;
    public int f;
    String state;
}
```

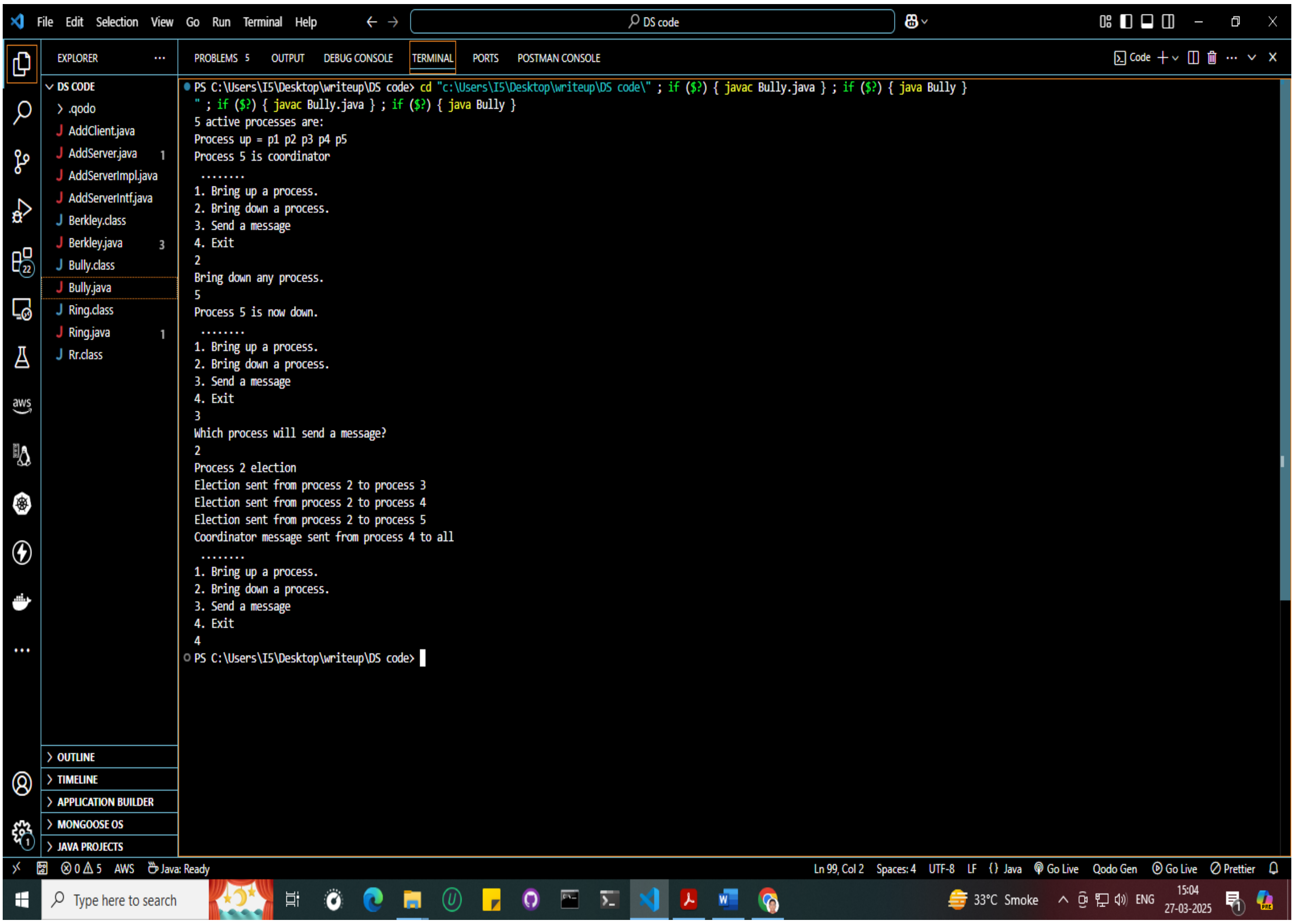


```
                                        cd "c:\Users\I5\Desktop\writeup\DS cod
e\" ; if ($?) { javac Ring.java } ; if ($?) { java Ring }                e\" ; if ($?) { javac Ring.java } ; if ($?) { java Ring }
Enter the number of processes:
4
Enter the id of process:
1
Enter the id of process:
2
Enter the id of process:
3
Enter the id of process:
4
 [0] 1 [1] 2 [2] 3 [3] 4
Process 4 selected as co-ordinator

1. Election 2. Quit
1

Enter the Process number who initialized election:
2

Process 2 sends message to 3

Process 3 sends message to 1

Process 1 sends message to 2

Process 3 selected as co-ordinator

1. Election 2. Quit
2
Program terminated ...
PS C:\Users\I5\Desktop\writeup\DS code>
```

```
//Bully.java
import java.util.Scanner;
public class Bully {
    static boolean[] state = new boolean[5];
    public static void up(int up) {
        if (state[up - 1]) {
            System.out.println("Process " + up + " is already up");
        } else {
            int i;
            state[up - 1] = true;
            System.out.println("Process " + up + " held election");
            for (i = up; i < 5; ++i) {
                System.out.println("Election message sent from process " + up + " to process " + (i + 1));
            }
            for (i = up + 1; i <= 5; ++i) {
                if (!state[i - 1]) continue;
                System.out.println("Alive message sent from process " + i + " to process " + up);
                break;
            }
        }
    }
    public static void down(int down) {
        if (!state[down - 1]) {
            System.out.println("Process " + down + " is already down.");
        } else {
            state[down - 1] = false;
            System.out.println("Process " + down + " is now down.");
        }
    }
    public static void mess(int mess) {
        if (state[mess - 1]) {
            if (state[4]) {
                System.out.println("OK");
            } else if (!state[4]) {
                int i;
                System.out.println("Process " + mess + " election");
                for (i = mess; i < 5; ++i) {
                    System.out.println("Election sent from process " + mess + " to process " + (i + 1));
                }
                for (i = 5; i >= mess; --i) {
```

```
                if (!state[i - 1]) continue;
                System.out.println("Coordinator message sent from process " + i + " to all");
                break;
            }
        }
    } else {
        System.out.println("Process " + mess + " is down");
    }
}
public static void main(String[] args) {
    int choice;
    Scanner sc = new Scanner(System.in);
    for (int i = 0; i < 5; ++i) {
        state[i] = true;
    }
    System.out.println("5 active processes are:");
    System.out.println("Process up = p1 p2 p3 p4 p5");
    System.out.println("Process 5 is coordinator");

    do {
        System.out.println(" ........ ");
        System.out.println("1. Bring up a process.");
        System.out.println("2. Bring down a process.");
        System.out.println("3. Send a message");
        System.out.println("4. Exit");
        choice = sc.nextInt();
        switch (choice) {
            case 1: {
                System.out.println("Bring process up");
                int up = sc.nextInt();
                if (up == 5) {
                    System.out.println("Process 5 is co-ordinator");
                    state[4] = true;
                    break;
                }
                up(up);
                break;
            }
            case 2: {
                System.out.println("Bring down any process.");
```

```
                    int down = sc.nextInt();
                    down(down);
                    break;
                }
                case 3: {
                    System.out.println("Which process will send a message?");
                    int mess = sc.nextInt();
                    mess(mess);
                    break;
                }
            }
        } while (choice != 4);
        sc.close();
    }
}
```



```
PS C:\Users\I5\Desktop\writeup\DS code> cd "c:\Users\I5\Desktop\writeup\DS code\" ; if ($?) { javac Bully.java } ; if ($?) { java Bully }
" ; if ($?) { javac Bully.java } ; if ($?) { java Bully }
5 active processes are:
Process up = p1 p2 p3 p4 p5
Process 5 is coordinator
 ........
1. Bring up a process.
2. Bring down a process.
3. Send a message
4. Exit
2
Bring down any process.
5
Process 5 is now down.
 .......
1. Bring up a process.
2. Bring down a process.
3. Send a message
4. Exit
3
Which process will send a message?
2
Process 2 election
Election sent from process 2 to process 3
Election sent from process 2 to process 4
Election sent from process 2 to process 5
Coordinator message sent from process 4 to all
 ........
1. Bring up a process.
2. Bring down a process.
3. Send a message
4. Exit
4
PS C:\Users\I5\Desktop\writeup\DS code> 
```

Assign1

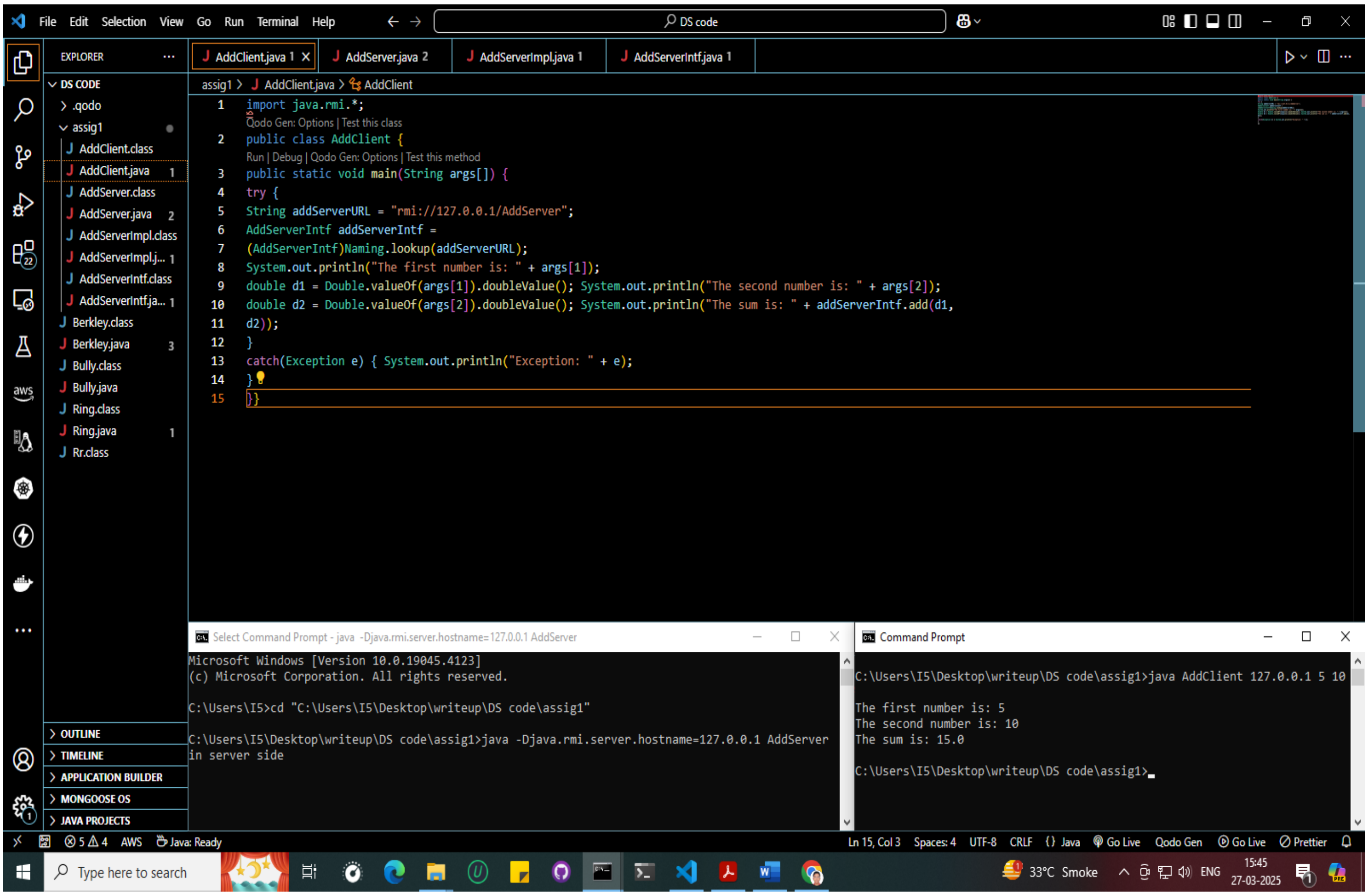
```
//AddClient.java
import java.rmi.*;
public class AddClient {
public static void main(String args[]) {
try {
String addServerURL = "rmi://127.0.0.1/AddServer";
AddServerIntf addServerIntf =
(AddServerIntf)Naming.lookup(addServerURL);
System.out.println("The first number is: " + args[1]);
double d1 = Double.valueOf(args[1]).doubleValue(); System.out.println("The second number is: " +
args[2]);
double d2 = Double.valueOf(args[2]).doubleValue(); System.out.println("The sum is: " +
addServerIntf.add(d1,
d2));
}
catch(Exception e) { System.out.println("Exception: " + e);
}
}}
```

```
//AddServer.java
import java.net.*;
import java.rmi.*;
public class AddServer {
public static void main(String args[]) {
try {
AddServerImpl addServerImpl = new AddServerImpl(); Naming.rebind("//127.0.0.1/AddServer", obj);
System.out.println("in server side");
}
catch(Exception e) { System.out.println("Exception: " + e);
}
}
}
```

```
//AddServerImpl
import java.rmi.*;
import java.rmi.server.*;
public class AddServerImpl extends UnicastRemoteObject implements
AddServerIntf {
public AddServerImpl() throws RemoteException {
}
public double add(double d1, double d2) throws RemoteException { return d1 + d2;
}
}
```

//AddClientIntf

```
import java.rmi.*;
public interface AddServerIntf extends Remote {
double add(double d1, double d2) throws RemoteException;
}
```



```
import java.rmi.*;
public class AddClient {
public static void main(String args[]) {
try {
String addServerURL = "rmi://127.0.0.1/AddServer";
AddServerIntf addServerIntf =
(AddServerIntf)Naming.lookup(addServerURL);
System.out.println("The first number is: " + args[1]);
double d1 = Double.valueOf(args[1]).doubleValue(); System.out.println("The second number is: " + args[2]);
double d2 = Double.valueOf(args[2]).doubleValue(); System.out.println("The sum is: " + addServerIntf.add(d1,
d2));
}
catch(Exception e) { System.out.println("Exception: " + e);
}
}}
```

Select Command Prompt - java -Djava.rmi.server.hostname=127.0.0.1 AddServer

```
Microsoft Windows [Version 10.0.19045.4123]
(c) Microsoft Corporation. All rights reserved.

C:\Users\I5>cd "C:\Users\I5\Desktop\writeup\DS code\assig1"

C:\Users\I5\Desktop\writeup\DS code\assig1>java -Djava.rmi.server.hostname=127.0.0.1 AddServer
in server side
```

```
C:\Users\I5\Desktop\writeup\DS code\assig1>java AddClient 127.0.0.1 5 10

The first number is: 5
The second number is: 10
The sum is: 15.0

C:\Users\I5\Desktop\writeup\DS code\assig1>
```

⊗ 5 ⚠ 4 AWS Java: Ready
Ln 15, Col 3 Spaces: 4 UTF-8 CRLF {} Java Go Live Qodo Gen Go Live Prettier
Type here to search
33°C Smoke ENG 15:45 27-03-2025

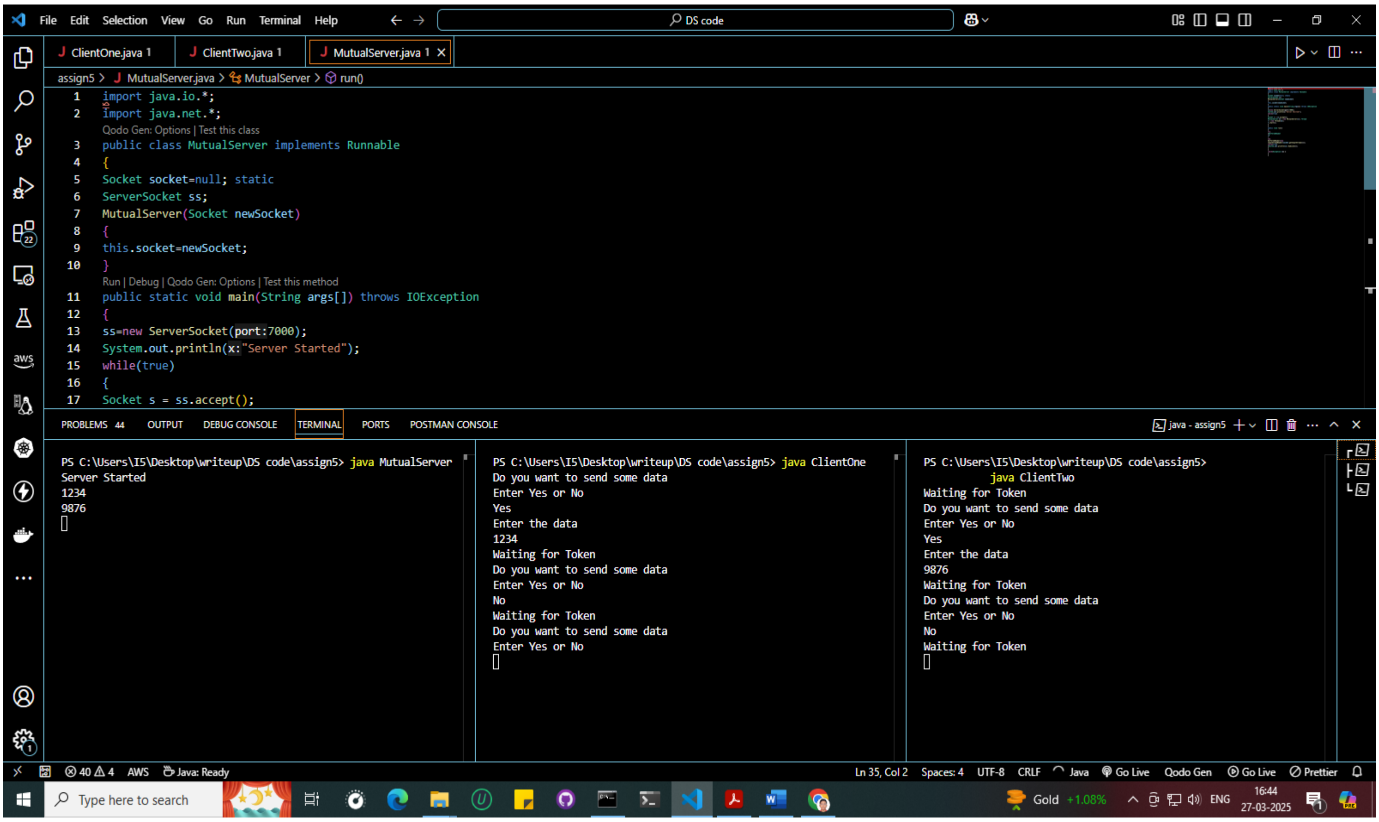
Assign5

```
//MutualServer.java
import java.io.*;
import java.net.*;
public class MutualServer implements Runnable
{
Socket socket=null; static
ServerSocket ss;
MutualServer(Socket newSocket)
{
this.socket=newSocket;
}
public static void main(String args[]) throws IOException
{
ss=new ServerSocket(7000);
System.out.println("Server Started");
while(true)
{
Socket s = ss.accept();
MutualServer es = new MutualServer(s); Thread
t = new Thread(es);
t.start();
}
}
public void run()
{
try
{BufferedReader
in
=
new
BufferedReader(new
InputStreamReader(socket.getInputStream()));
while(true)
{System.out.println(in.readLine());
}
}
catch(Exception e){ }
}
}
```

```
//ClientOne.java
import java.io.*;
import java.net.*; public
class ClientOne
{
public static void main(String args[])throws IOException
{
Socket s=new Socket("localhost",7000);
PrintStream out = new PrintStream(s.getOutputStream());
ServerSocket ss = new ServerSocket(7001);
Socket s1 = ss.accept();
BufferedReader in1 = new BufferedReader(new
InputStreamReader(s1.getInputStream()));
PrintStream out1 = new PrintStream(s1.getOutputStream()); BufferedReader br = new
BufferedReader(new InputStreamReader(System.in));
String str="Token";
while(true)
{
if(str.equalsIgnoreCase("Token"))
{
System.out.println("Do you want to send some data");
System.out.println("Enter Yes or No"); str=br.readLine();
if(str.equalsIgnoreCase("Yes"))
{System.out.println("Enter the data");
str=br.readLine();
out.println(str);
}
out1.println("Token");
}
System.out.println("Waiting for Token");
str=in1.readLine();
}
}
}
```

```
//ClientTwo.java
import java.io.*;
import java.net.*;
public class ClientTwo
{
```

```
public static void main(String args[])throws IOException
{
Socket s=new Socket("localhost",7000);
PrintStream out = new PrintStream(s.getOutputStream()); Socket
s2=new Socket("localhost",7001); BufferedReader in2 = new
BufferedReader(new InputStreamReader(s2.getInputStream()));
PrintStream out2 = new PrintStream(s2.getOutputStream()); BufferedReader br = new
BufferedReader(new InputStreamReader(System.in));
String str;
while(true)
{
System.out.println("Waiting for Token");
str=in2.readLine();
if(str.equalsIgnoreCase("Token"))
{
System.out.println("Do you want to send some data");
System.out.println("Enter Yes or No"); str=br.readLine();
if(str.equalsIgnoreCase("Yes")){
System.out.println("Enter the data"); str=br.readLine();
out.println(str);
}
out2.println("Token");
}
}
}
}
```



```
import java.io.*;
import java.net.*;
public class MutualServer implements Runnable
{
Socket socket=null; static
ServerSocket ss;
MutualServer(Socket newSocket)
{
this.socket=newSocket;
}
public static void main(String args[]) throws IOException
{
ss=new ServerSocket(port:7000);
System.out.println(x:"Server Started");
while(true)
{
Socket s = ss.accept();
```

```
PS C:\Users\I5\Desktop\writeup\DS code\assign5> java MutualServer
Server Started
1234
9876
```

```
PS C:\Users\I5\Desktop\writeup\DS code\assign5> java ClientOne
Do you want to send some data
Enter Yes or No
Yes
Enter the data
1234
Waiting for Token
Do you want to send some data
Enter Yes or No
No
Waiting for Token
Do you want to send some data
Enter Yes or No
```

```
PS C:\Users\I5\Desktop\writeup\DS code\assign5>
        java ClientTwo
Waiting for Token
Do you want to send some data
Enter Yes or No
Yes
Enter the data
9876
Waiting for Token
Do you want to send some data
Enter Yes or No
No
Waiting for Token
```

Assign2

// ReverseServer.java

```
import ReverseModule.ReverseHelper; // Add this if missing
import ReverseModule.Reverse;
import org.omg.CosNaming.*;
import org.omg.CosNaming.NamingContextPackage.*;
import org.omg.CORBA.*;
import org.omg.PortableServer.*;

class ReverseServer {
    public static void main(String[] args) {
        try {
            // Initialize the ORB
            ORB orb = ORB.init(args, null);

            // Initialize the POA
            POA rootPOA = POAHelper.narrow(orb.resolve_initial_references("RootPOA"));
            rootPOA.the_POAManager().activate();

            // Create an instance of ReverseImpl
            ReverseImpl rvr = new ReverseImpl();
            org.omg.CORBA.Object ref = rootPOA.servant_to_reference(rvr);
            Reverse h_ref = ReverseHelper.narrow(ref);

            // Register with Naming Service
            org.omg.CORBA.Object objRef = orb.resolve_initial_references("NameService");
            NamingContextExt ncRef = NamingContextExtHelper.narrow(objRef);

            String name = "Reverse";
            NameComponent path[] = ncRef.to_name(name);
            ncRef.rebind(path, h_ref);

            System.out.println("Reverse Server is running...");
            orb.run();
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

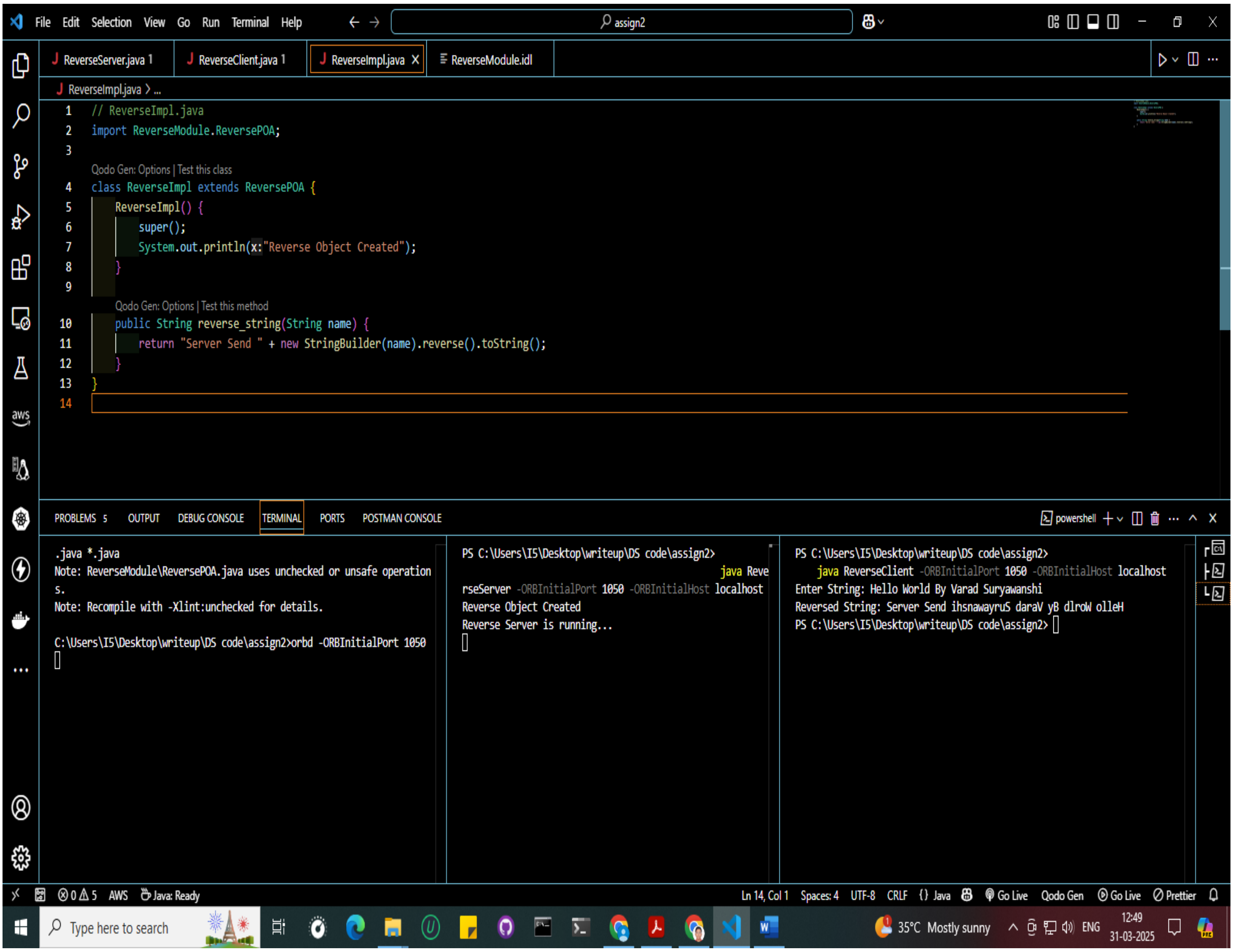
```
// ReverseClient.java
import ReverseModule.*;
import org.omg.CosNaming.*;
import org.omg.CosNaming.NamingContextPackage.*;
import org.omg.CORBA.*;
import java.io.*;

class ReverseClient {
    public static void main(String args[]) {
        Reverse ReverseImpl = null;
        try {
            // Initialize the ORB
            ORB orb = ORB.init(args, null);
            org.omg.CORBA.Object objRef = orb.resolve_initial_references("NameService");
            NamingContextExt ncRef = NamingContextExtHelper.narrow(objRef);

            String name = "Reverse";
            ReverseImpl = ReverseHelper.narrow(ncRef.resolve_str(name));

            System.out.print("Enter String: ");
            BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
            String str = br.readLine();

            String tempStr = ReverseImpl.reverse_string(str);
            System.out.println("Reversed String: " + tempStr);
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

```
// ReverseImpl.java
import ReverseModule.ReversePOA;

class ReverseImpl extends ReversePOA {
    ReverseImpl() {
        super();
        System.out.println("Reverse Object Created");
    }
    public String reverse_string(String name) {
        return "Server Send " + new StringBuilder(name).reverse().toString();
    }
}
```

// ReverseModule.idl

```
module ReverseModule {
    interface Reverse {
        string reverse_string(in string str);
    };
};
```

1) idlj -fall ReverseModule.idl
2) javac -d . ReverseModule/*.java *.java
3) orbd -ORBInitialPort 1050
4) java ReverseServer -ORBInitialPort 1050 -ORBInitialHost localhost
5) java ReverseClient -ORBInitialPort 1050 -ORBInitialHost localhost



```
// ReverseImpl.java
import ReverseModule.ReversePOA;

Qodo Gen: Options | Test this class
class ReverseImpl extends ReversePOA {
    ReverseImpl() {
        super();
        System.out.println(x:"Reverse Object Created");
    }

    Qodo Gen: Options | Test this method
    public String reverse_string(String name) {
        return "Server Send " + new StringBuilder(name).reverse().toString();
    }
}
```

```
.java *.java
Note: ReverseModule\ReversePOA.java uses unchecked or unsafe operation
s.
Note: Recompile with -Xlint:unchecked for details.

C:\Users\I5\Desktop\writeup\DS code\assign2>orbd -ORBInitialPort 1050
```

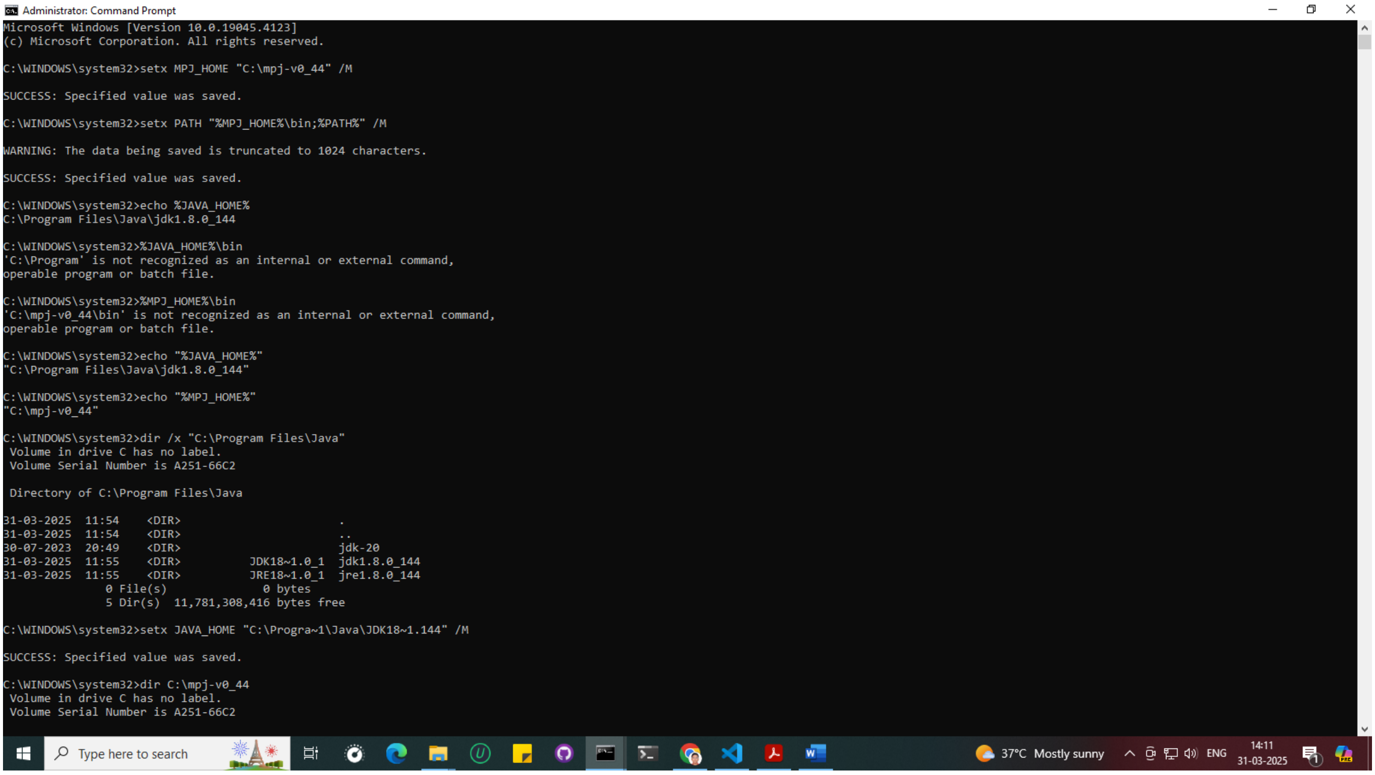
```
PS C:\Users\I5\Desktop\writeup\DS code\assign2>
                                                java Reve
rseServer -ORBInitialPort 1050 -ORBInitialHost localhost
Reverse Object Created
Reverse Server is running...
```

```
PS C:\Users\I5\Desktop\writeup\DS code\assign2>
    java ReverseClient -ORBInitialPort 1050 -ORBInitialHost localhost
Enter String: Hello World By Varad Suryawanshi
Reversed String: Server Send ihsnawayruS daraV yB dlroW olleH
PS C:\Users\I5\Desktop\writeup\DS code\assign2>
```

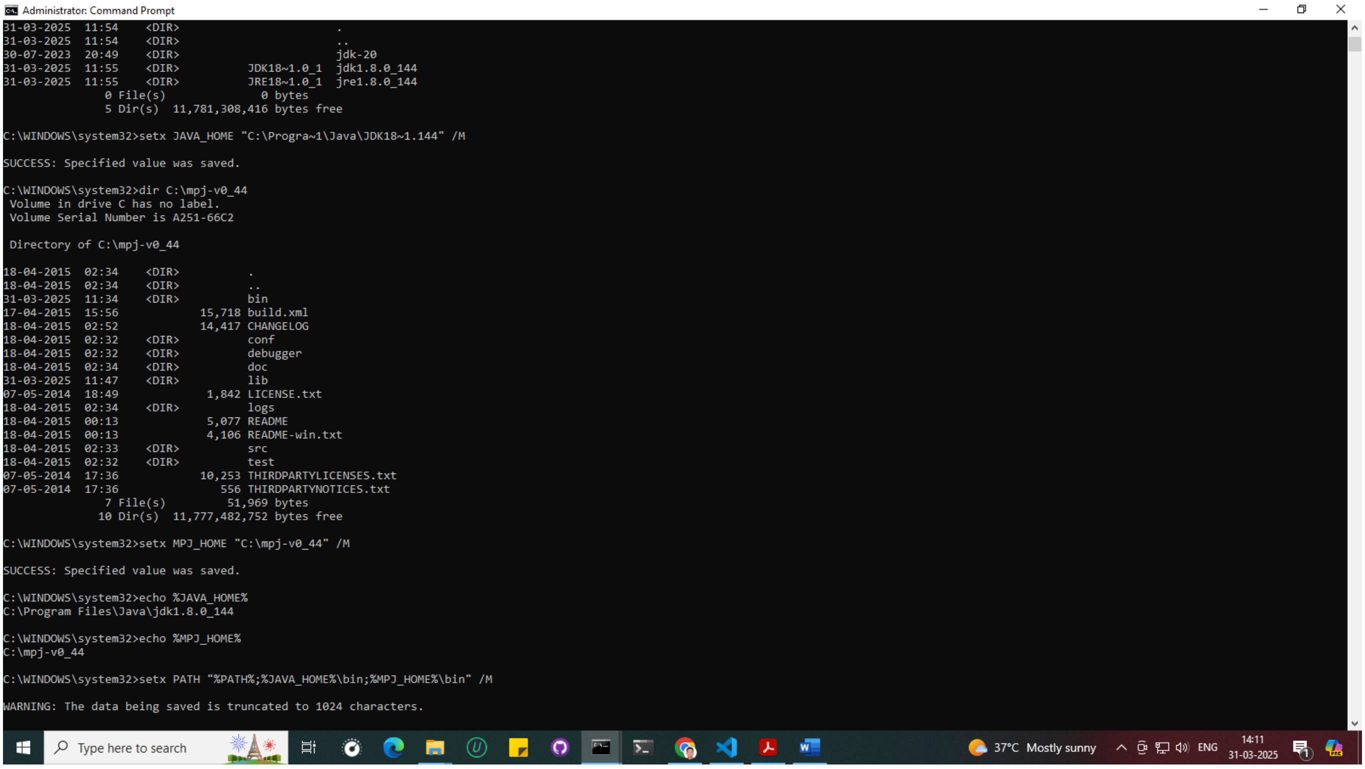
Assign3

//ScatterGather.java

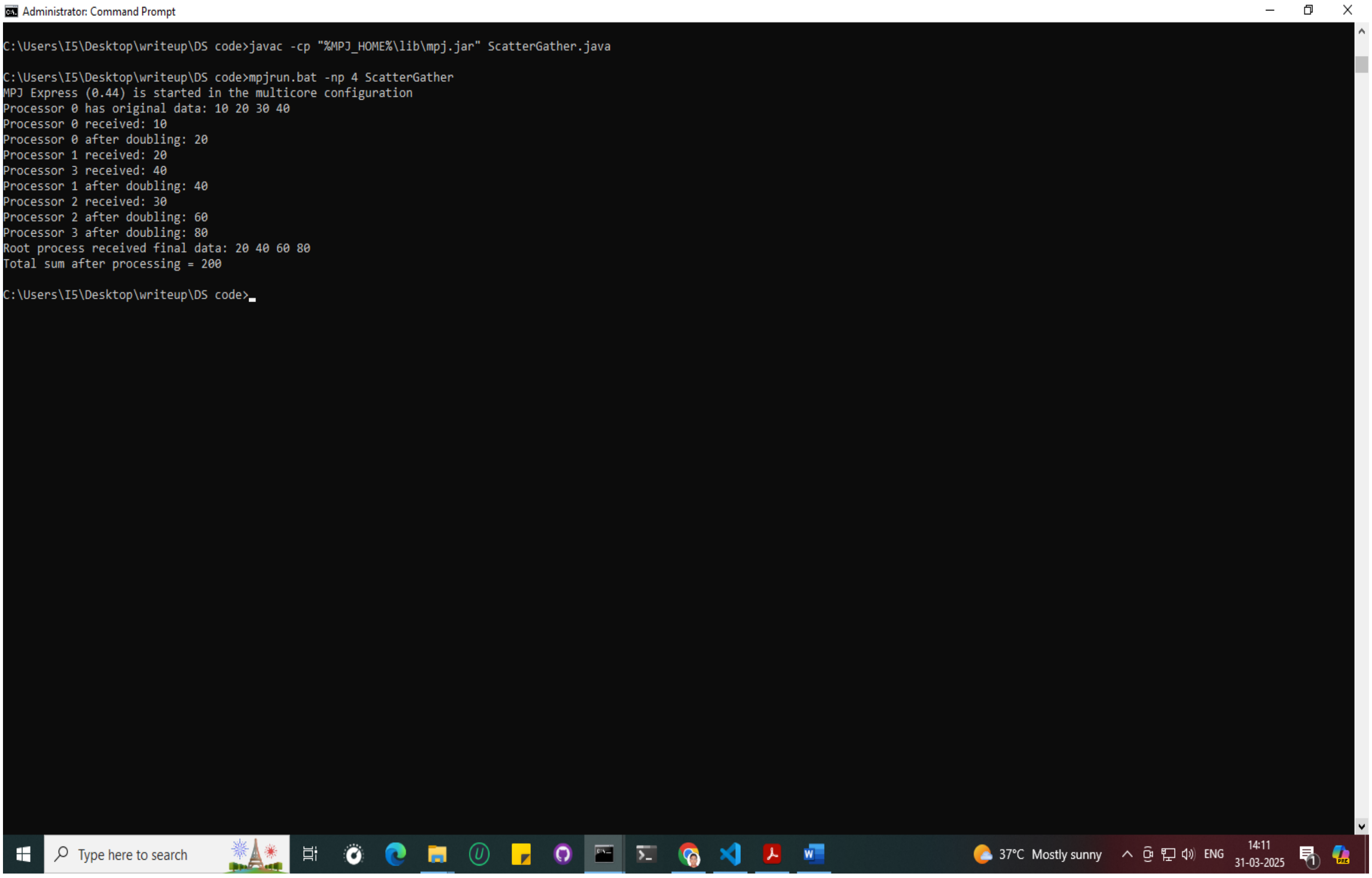
```
import mpi.MPI;

public class ScatterGather {
    public static void main(String args[]) {
        // Initialize MPI execution environment
        MPI.Init(args);

        // Get the ID of the current process
        int rank = MPI.COMM_WORLD.Rank();
        int size = MPI.COMM_WORLD.Size();
        int root = 0;

        int totalElements = size;
        int[] sendbuf = new int[totalElements];  //  Ensure sendbuf is not null for all ranks

        // Only the root process initializes the full array
        if (rank == root) {
            sendbuf[0] = 10;
            sendbuf[1] = 20;
            sendbuf[2] = 30;
            sendbuf[3] = 40;

            System.out.print("Processor " + rank + " has original data: ");
            for (int i = 0; i < totalElements; i++) {
                System.out.print(sendbuf[i] + " ");
            }
            System.out.println();
        }

        // Each process will receive 1 element
        int[] recvbuf = new int[1];

        // Scatter the data to all processes
        MPI.COMM_WORLD.Scatter(sendbuf, 0, 1, MPI.INT, recvbuf, 0, 1, MPI.INT, root);

        // Display the data received by each processor
        System.out.println("Processor " + rank + " received: " + recvbuf[0]);
        // Each processor processes its data (e.g., doubles it)
        recvbuf[0] = recvbuf[0] * 2;
        System.out.println("Processor " + rank + " after doubling: " + recvbuf[0]);

        //  Ensure sendbuf is properly initialized for `Gather`
        if (rank == root) {
            sendbuf = new int[totalElements];  // Root needs to have an allocated array
        }

        // Gather the processed data back at the root
        MPI.COMM_WORLD.Gather(recvbuf, 0, 1, MPI.INT, sendbuf, 0, 1, MPI.INT, root);

        // Root displays final gathered data and computes the sum
        if (rank == root) {
            System.out.print("Root process received final data: ");
            int totalSum = 0;
            for (int i = 0; i < totalElements; i++) {
                System.out.print(sendbuf[i] + " ");
                totalSum += sendbuf[i];
            }
            System.out.println("\nTotal sum after processing = " + totalSum);
        }

        // Finalize MPI environment
        MPI.Finalize();
    }
}
```



```
Microsoft Windows [Version 10.0.19045.4123]
(c) Microsoft Corporation. All rights reserved.

C:\WINDOWS\system32>setx MPJ_HOME "C:\mpj-v0_44" /M

SUCCESS: Specified value was saved.

C:\WINDOWS\system32>setx PATH "%MPJ_HOME%\bin;%PATH%" /M

WARNING: The data being saved is truncated to 1024 characters.

SUCCESS: Specified value was saved.

C:\WINDOWS\system32>echo %JAVA_HOME%
C:\Program Files\Java\jdk1.8.0_144

C:\WINDOWS\system32>%JAVA_HOME%\bin
'C:\Program' is not recognized as an internal or external command,
operable program or batch file.

C:\WINDOWS\system32>%MPJ_HOME%\bin
'C:\mpj-v0_44\bin' is not recognized as an internal or external command,
operable program or batch file.

C:\WINDOWS\system32>echo "%JAVA_HOME%"
"C:\Program Files\Java\jdk1.8.0_144"

C:\WINDOWS\system32>echo "%MPJ_HOME%"
"C:\mpj-v0_44"

C:\WINDOWS\system32>dir /x "C:\Program Files\Java"
 Volume in drive C has no label.
 Volume Serial Number is A251-66C2

 Directory of C:\Program Files\Java

31-03-2025  11:54    <DIR>                       .
31-03-2025  11:54    <DIR>                       ..
30-07-2023  20:49    <DIR>                       jdk-20
31-03-2025  11:55    <DIR>          JDK18~1.0_1  jdk1.8.0_144
31-03-2025  11:55    <DIR>          JRE18~1.0_1  jre1.8.0_144
               0 File(s)              0 bytes
               5 Dir(s)  11,781,308,416 bytes free

C:\WINDOWS\system32>setx JAVA_HOME "C:\Progra~1\Java\JDK18~1.144" /M

SUCCESS: Specified value was saved.

C:\WINDOWS\system32>dir C:\mpj-v0_44
 Volume in drive C has no label.
 Volume Serial Number is A251-66C2
```



```
31-03-2025  11:54    <DIR>                          .
31-03-2025  11:54    <DIR>                          ..
30-07-2023  20:49    <DIR>                          jdk-20
31-03-2025  11:55    <DIR>          JDK18~1.0_1  jdk1.8.0_144
31-03-2025  11:55    <DIR>          JRE18~1.0_1  jre1.8.0_144
               0 File(s)              0 bytes
               5 Dir(s)  11,781,308,416 bytes free

C:\WINDOWS\system32>setx JAVA_HOME "C:\Progra~1\Java\JDK18~1.144" /M

SUCCESS: Specified value was saved.

C:\WINDOWS\system32>dir C:\mpj-v0_44
 Volume in drive C has no label.
 Volume Serial Number is A251-66C2

 Directory of C:\mpj-v0_44

18-04-2015  02:34    <DIR>          .
18-04-2015  02:34    <DIR>          ..
31-03-2025  11:34    <DIR>          bin
17-04-2015  15:56            15,718 build.xml
18-04-2015  02:52            14,417 CHANGELOG
18-04-2015  02:32    <DIR>          conf
18-04-2015  02:32    <DIR>          debugger
18-04-2015  02:34    <DIR>          doc
31-03-2025  11:47    <DIR>          lib
07-05-2014  18:49             1,842 LICENSE.txt
18-04-2015  02:34    <DIR>          logs
18-04-2015  00:13             5,077 README
18-04-2015  00:13             4,106 README-win.txt
18-04-2015  02:33    <DIR>          src
18-04-2015  02:32    <DIR>          test
07-05-2014  17:36            10,253 THIRDPARTYLICENSES.txt
07-05-2014  17:36               556 THIRDPARTYNOTICES.txt
               7 File(s)         51,969 bytes
              10 Dir(s)  11,777,482,752 bytes free

C:\WINDOWS\system32>setx MPJ_HOME "C:\mpj-v0_44" /M

SUCCESS: Specified value was saved.

C:\WINDOWS\system32>echo %JAVA_HOME%
C:\Program Files\Java\jdk1.8.0_144

C:\WINDOWS\system32>echo %MPJ_HOME%
C:\mpj-v0_44

C:\WINDOWS\system32>setx PATH "%PATH%;%JAVA_HOME%\bin;%MPJ_HOME%\bin" /M

WARNING: The data being saved is truncated to 1024 characters.
```



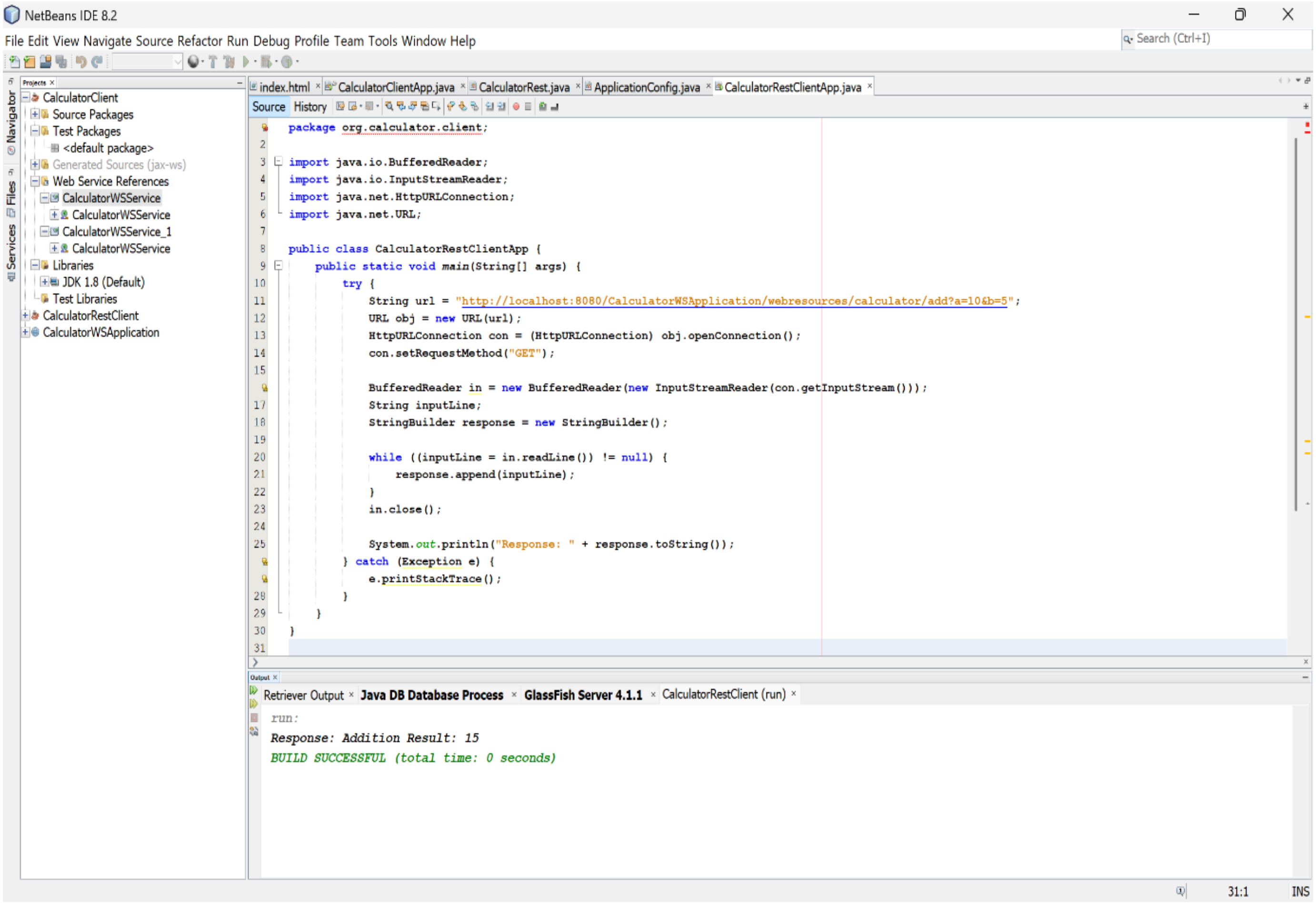
```
C:\Users\I5\Desktop\writeup\DS code>javac -cp "%MPJ_HOME%\lib\mpj.jar" ScatterGather.java

C:\Users\I5\Desktop\writeup\DS code>mpjrun.bat -np 4 ScatterGather
MPJ Express (0.44) is started in the multicore configuration
Processor 0 has original data: 10 20 30 40
Processor 0 received: 10
Processor 0 after doubling: 20
Processor 1 received: 20
Processor 3 received: 40
Processor 1 after doubling: 40
Processor 2 received: 30
Processor 2 after doubling: 60
Processor 3 after doubling: 80
Root process received final data: 20 40 60 80
Total sum after processing = 200

C:\Users\I5\Desktop\writeup\DS code>
```

Assign7th

```
//CalculatorWS.java
package org.calculator;
import javax.jws.WebMethod;
import javax.jws.WebService;
@WebService
public class CalculatorWS {
  @WebMethod
  public int add(int a, int b) {
    return a + b;
  }
  @WebMethod
  public int subtract(int a, int b) {
    return a - b;
  }
}

//CalculatorClientApp
package org.calculator.client;
import org.calculator.CalculatorWS;
import org.calculator.CalculatorWSService;
public class CalculatorClientApp {
  public static void main(String[] args) {
    CalculatorWSService service = new CalculatorWSService();
    CalculatorWS calculator = service.getCalculatorWSPort();
    int resultAdd = calculator.add(10, 5);
    int resultSub = calculator.subtract(10, 5);
    System.out.println("Addition Result: " + resultAdd);
    System.out.println("Subtraction Result: " + resultSub);
  }
}

//CalculatorRest.java
package org.calculator;

import javax.ws.rs.GET;
import javax.ws.rs.Path;
import javax.ws.rs.Produces;
import javax.ws.rs.QueryParam;
import javax.ws.rs.core.MediaType;
```

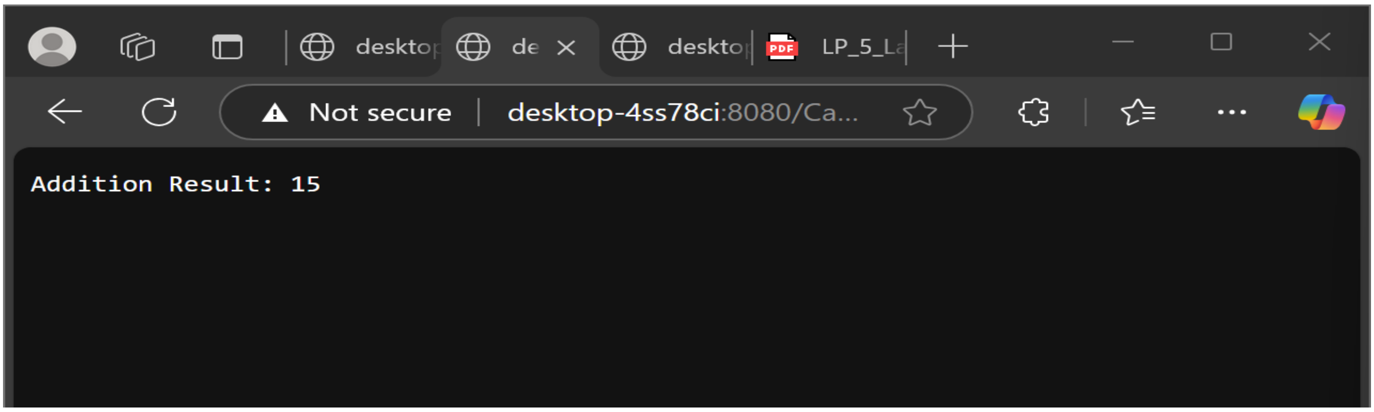
```
@Path("/calculator")
public class CalculatorRest {

   @GET
   @Path("/add")
   @Produces(MediaType.TEXT_PLAIN)
   public String add(@QueryParam("a") int a, @QueryParam("b") int b) {
      return "Addition Result: " + (a + b);
   }

   @GET
   @Path("/subtract")
   @Produces(MediaType.TEXT_PLAIN)
   public String subtract(@QueryParam("a") int a, @QueryParam("b") int b) {
      return "Subtraction Result: " + (a - b);
   }
}

//CalculatorRestClientApp
package org.calculator.client;

import java.io.BufferedReader;
import java.io.InputStreamReader;
import java.net.HttpURLConnection;
import java.net.URL;

public class CalculatorRestClientApp {
   public static void main(String[] args) {
      try {
         String url = "http://desktop-
4ss78ci:8080/CalculatorWSApplication/webresources/calculator/add?a=10&b=5";
         String url =
"http://localhost:8080/CalculatorWSApplication/webresources/calculator/add?a=10&b=5";
         URL obj = new URL(url);
         HttpURLConnection con = (HttpURLConnection) obj.openConnection();
         con.setRequestMethod("GET");

         BufferedReader in = new BufferedReader(new InputStreamReader(con.getInputStream()));
         String inputLine;
```

```
            StringBuilder response = new StringBuilder();

            while ((inputLine = in.readLine()) != null) {
                response.append(inputLine);
            }
            in.close();

            System.out.println("Response: " + response.toString());
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

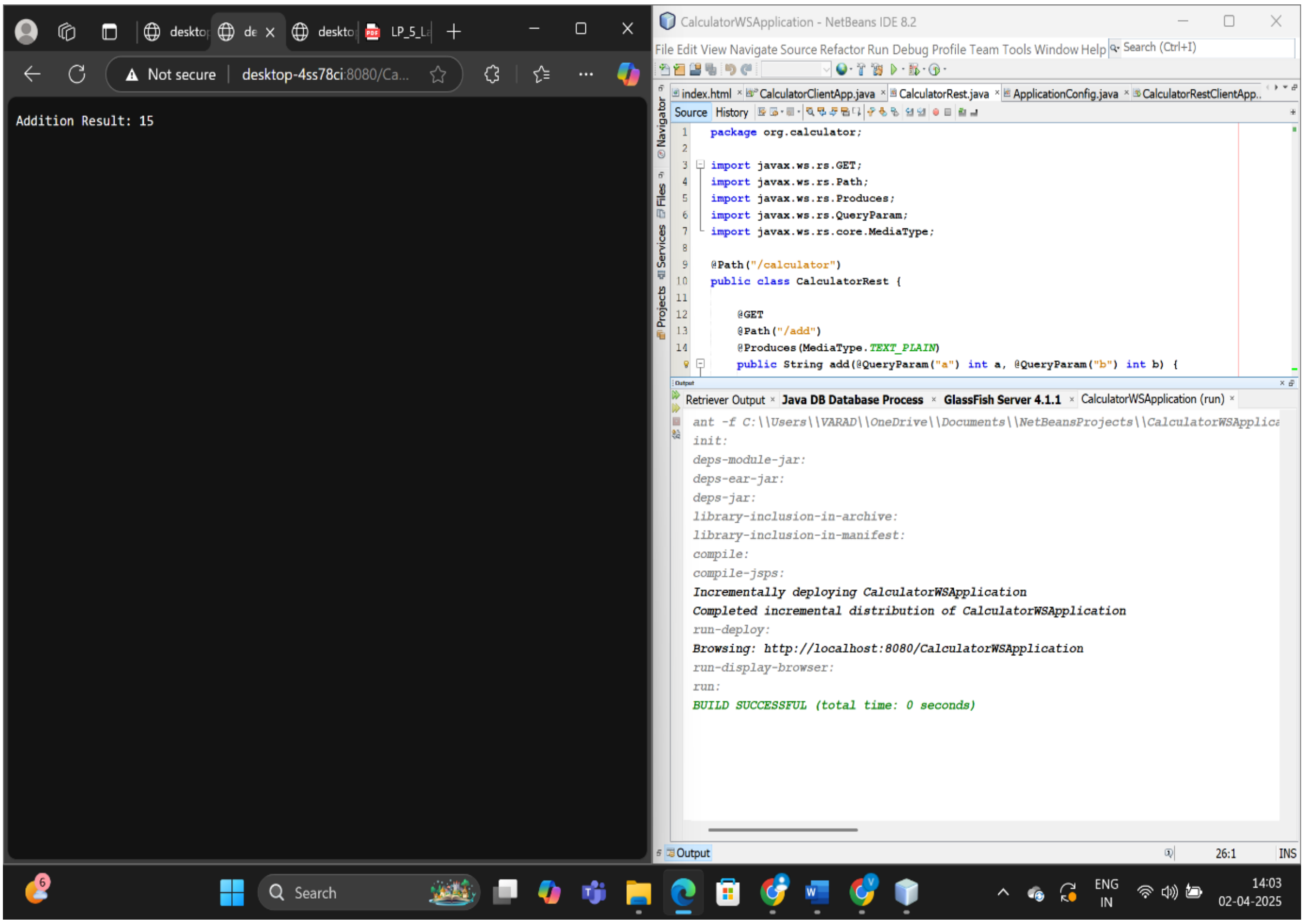


```
package org.calculator.client;

import java.io.BufferedReader;
import java.io.InputStreamReader;
import java.net.HttpURLConnection;
import java.net.URL;

public class CalculatorRestClientApp {
    public static void main(String[] args) {
        try {
            String url = "http://localhost:8080/CalculatorWSApplication/webresources/calculator/add?a=10&b=5";
            URL obj = new URL(url);
            HttpURLConnection con = (HttpURLConnection) obj.openConnection();
            con.setRequestMethod("GET");

            BufferedReader in = new BufferedReader(new InputStreamReader(con.getInputStream()));
            String inputLine;
            StringBuilder response = new StringBuilder();

            while ((inputLine = in.readLine()) != null) {
                response.append(inputLine);
            }
            in.close();

            System.out.println("Response: " + response.toString());
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

```
run:
Response: Addition Result: 15
BUILD SUCCESSFUL (total time: 0 seconds)
```



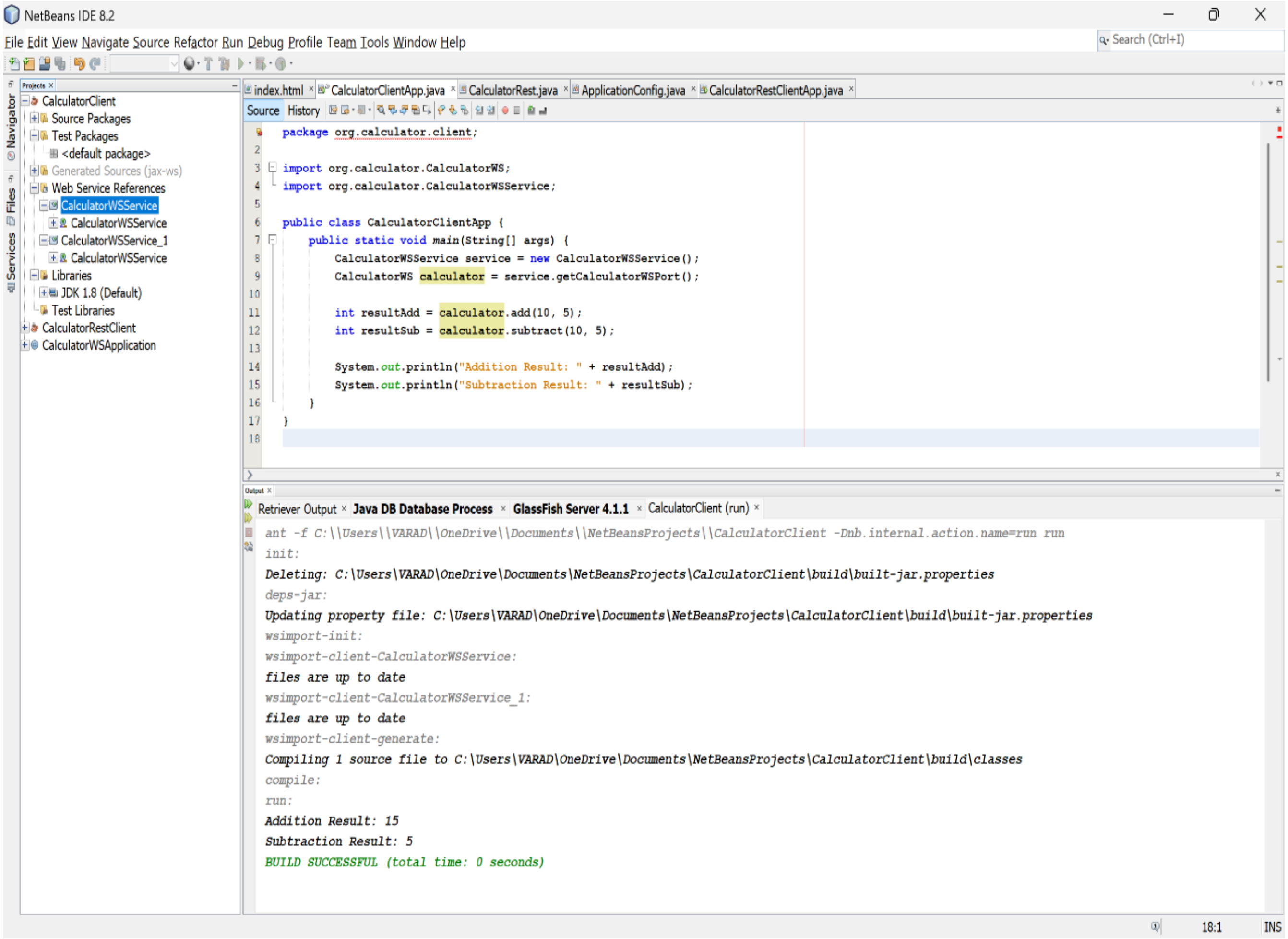
```
Addition Result: 15
```



```
Addition Result: 15
```

```java
package org.calculator;

import javax.ws.rs.GET;
import javax.ws.rs.Path;
import javax.ws.rs.Produces;
import javax.ws.rs.QueryParam;
import javax.ws.rs.core.MediaType;

@Path("/calculator")
public class CalculatorRest {

    @GET
    @Path("/add")
    @Produces(MediaType.TEXT_PLAIN)
    public String add(@QueryParam("a") int a, @QueryParam("b") int b) {
```

```
ant -f C:\\Users\\VARAD\\OneDrive\\Documents\\NetBeansProjects\\CalculatorWSApplica
init:
deps-module-jar:
deps-ear-jar:
deps-jar:
library-inclusion-in-archive:
library-inclusion-in-manifest:
compile:
compile-jsps:
Incrementally deploying CalculatorWSApplication
Completed incremental distribution of CalculatorWSApplication
run-deploy:
Browsing: http://localhost:8080/CalculatorWSApplication
run-display-browser:
run:
BUILD SUCCESSFUL (total time: 0 seconds)
```



```java
package org.calculator.client;

import org.calculator.CalculatorWS;
import org.calculator.CalculatorWSService;

public class CalculatorClientApp {
    public static void main(String[] args) {
        CalculatorWSService service = new CalculatorWSService();
        CalculatorWS calculator = service.getCalculatorWSPort();

        int resultAdd = calculator.add(10, 5);
        int resultSub = calculator.subtract(10, 5);

        System.out.println("Addition Result: " + resultAdd);
        System.out.println("Subtraction Result: " + resultSub);
    }
}
```

```
ant -f C:\\Users\\VARAD\\OneDrive\\Documents\\NetBeansProjects\\CalculatorClient -Dnb.internal.action.name=run run
init:
Deleting: C:\Users\VARAD\OneDrive\Documents\NetBeansProjects\CalculatorClient\build\built-jar.properties
deps-jar:
Updating property file: C:\Users\VARAD\OneDrive\Documents\NetBeansProjects\CalculatorClient\build\built-jar.properties
wsimport-init:
wsimport-client-CalculatorWSService:
files are up to date
wsimport-client-CalculatorWSService_1:
files are up to date
wsimport-client-generate:
Compiling 1 source file to C:\Users\VARAD\OneDrive\Documents\NetBeansProjects\CalculatorClient\build\classes
compile:
run:
Addition Result: 15
Subtraction Result: 5
BUILD SUCCESSFUL (total time: 0 seconds)
```