**Fibonacci  
Client**

package fibonacci;

import java.rmi.registry.LocateRegistry;

import java.rmi.registry.Registry;

import java.util.Scanner;

public class Client {

public static void main(String[] args) {

try {

Registry registry = LocateRegistry.getRegistry("localhost", 1098);

FibonacciCalculator calculator = (FibonacciCalculator) registry.lookup("FibonacciCalculator");

Scanner scanner = new Scanner(System.in);

System.out.print("Nhập số Fibonacci cần tính: ");

int n = scanner.nextInt();

int result = calculator.calculateFibonacci(n);

System.out.println("Fibonacci(" + n + ") = " + result);

scanner.close();

} catch (Exception e) {

System.err.println("Client exception: " + e.toString());

e.printStackTrace();

}

}

}

**interface FibonacciCalculator**

package fibonacci;

import java.rmi.Remote;

import java.rmi.RemoteException;

public interface FibonacciCalculator extends Remote {

int calculateFibonacci(int n) throws RemoteException;

}

**FibonacciCalculatorImpl**package fibonacci;

import java.rmi.RemoteException;

import java.rmi.server.UnicastRemoteObject;

public class FibonacciCalculatorImpl extends UnicastRemoteObject implements FibonacciCalculator {

protected FibonacciCalculatorImpl() throws RemoteException {

super();

}

@Override

public int calculateFibonacci(int n) throws RemoteException {

if (n <= 1) {

return n;

} else {

return calculateFibonacci(n - 1) + calculateFibonacci(n - 2);

}

}

}

**Server**package fibonacci;

import java.rmi.registry.LocateRegistry;

import java.rmi.registry.Registry;

public class Server {

public static void main(String[] args) {

try {

FibonacciCalculator calculator = new FibonacciCalculatorImpl();

Registry registry = LocateRegistry.createRegistry(1098);

registry.rebind("FibonacciCalculator", calculator);

System.out.println("Server is running...");

} catch (Exception e) {

System.err.println("Server exception: " + e.toString());

e.printStackTrace();

}

}

}