**Creating the remote interface**

Calc.java

import java.rmi.\*;

public interface Adder extends Remote{

public int add(int x,int y)throws RemoteException;

public int sub(int x, int y) throws RemoteException;

public int mul(int x, int y) throws RemoteException;

}

**Providing implementation of remote interface**

CalcRemote.java

import java.rmi.\*;

import java.rmi.server.\*;

public class CalcRemote extends UnicastRemoteObject implements Calc{

CalcRemote()throws RemoteException{

super();

}

public int add(int x,int y){return x+y;}

public int sub(int x,int y){return x-y;}

public int mul(int x,int y){return x\*y;}

}

**Creating Client file**

MyClient.java

import java.rmi.\*;

public class MyClient{

public static void main(String args[]){

try{

Calc stub=(Calc)Naming.lookup("rmi://localhost:5000/anish");

System.out.println(stub.add(32,4));

}catch(Exception e){}

}

}

**Creating Server file**

MyServer.java

import java.rmi.\*;

import java.rmi.registry.\*;

public class MyServer{

public static void main(String args[]){

try{

Calc stub=new CalcRemote();

Naming.rebind("rmi://localhost:5000/anish",stub);

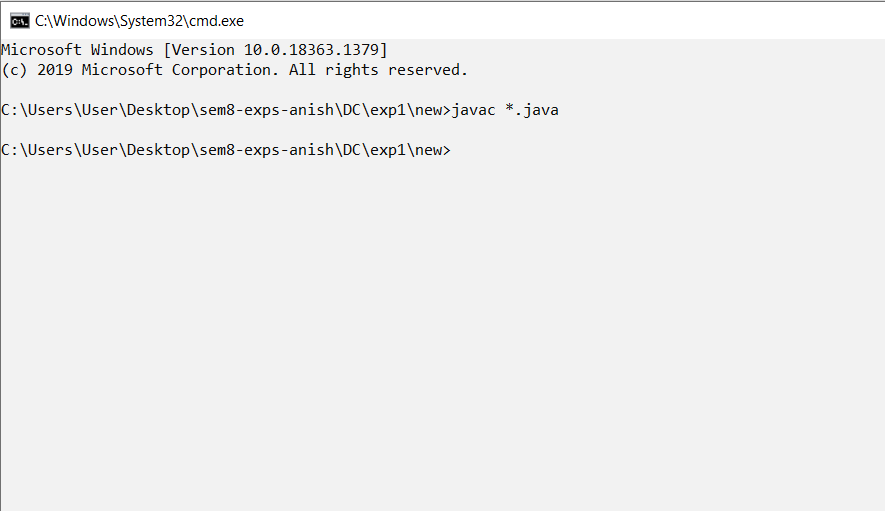
}catch(Exception e){System.out.println(e);}

}

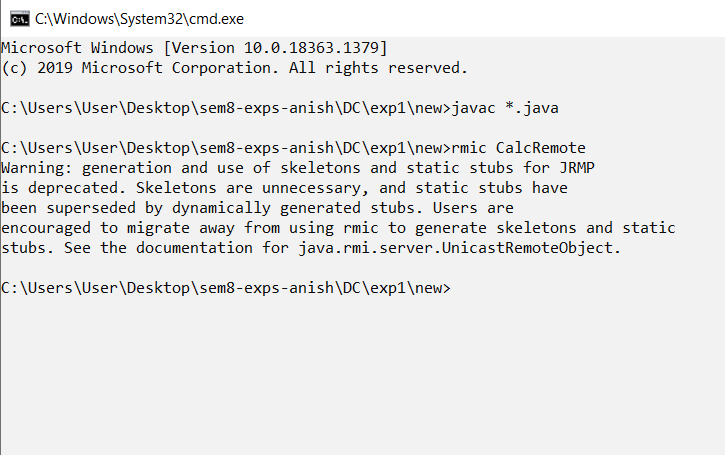
}

**Implementation Steps**

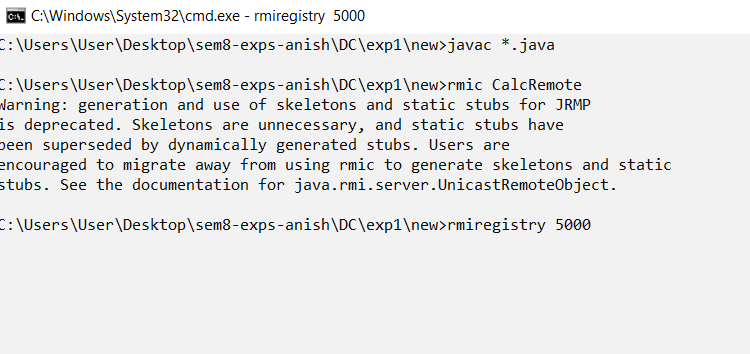
**Step1: Compile all java files**



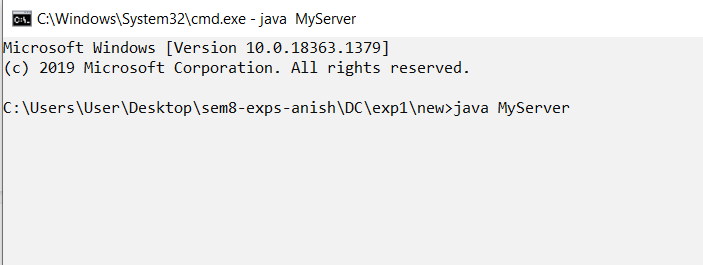
**Step2: create stub and skeleton object by using rmic tool**



**Step3: start rmi registry in one command prompt**



**Step4: Start the server in another command prompt**



**Step5: Start Client in another command prompt**

