13\_Nguyễn Bảo Linh 18073011\_DHCNTT14

**Báo cáo các bài thực hành tuần 8 RMI**

Tạo lớp Interface là một thông điệp cho cả Server và Client

Cả client và server đều phải tuân thủ nó, chỉ được triển khai và gọi các phương thức cho phép. Các phương thức đều kèm theo throws RemoteException

**HelloInterface.java**

import java.rmi.Remote;

import java.rmi.RemoteException;

public interface HelloInterface extends Remote {

    String printHello() throws RemoteException;

    String daonguoc(String ten)throws RemoteException;

    int countWords(String ten)throws RemoteException;

}

Tạo một lớp thực thi Implement cho Interface

**HelloImpl.java**

import java.rmi.RemoteException;

import java.rmi.server.UnicastRemoteObject;

public class HelloImpl extends UnicastRemoteObject implements HelloInterface {

    public HelloImpl() throws RemoteException {

        super();

    }

    public String printHello() {

        return "Hello RMI ....";

    }

    public String daonguoc(String ten){

        return ten;

    }

    public static final char SPACE = ' ';

    public static final char TAB = '\t';

    public static final char BREAK\_LINE = '\n';

    public  int countWords(String ten) {

        if (ten == null) {

            return -1;

        }

        int count = 0;

        int size = ten.length();

        boolean notCounted = true;

        for (int i = 0; i < size; i++) {

            if (ten.charAt(i) != SPACE && ten.charAt(i) != TAB

                    && ten.charAt(i) != BREAK\_LINE) {

                if(notCounted) {

                    count++;

                    notCounted = false;

                }

            } else {

                notCounted = true;

            }

        }

        return count;

    }

}

Cài đặt cho đối tượng trên RMIServer:

**Server.java**

import java.rmi.registry.LocateRegistry;

import java.rmi.registry.Registry;

public class Server {

    public static void main(String[] args) {

        try {

            HelloImpl obj = new HelloImpl();

            System.out.println("Exporting .....");

            LocateRegistry.createRegistry(6789);

            Registry reg = LocateRegistry.getRegistry(6789);

            reg.bind("HelloObj", obj);

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

        } catch (Exception e) {

            // TODO: handle exception

        }

    }

}

Cài đặt đối tượng trên RMIclient:

**Client.java**

import java.rmi.registry.LocateRegistry;

import java.rmi.registry.Registry;

import java.util.Scanner;

public class Client {

    private static Scanner nhapvao;

    public static void main(String[] args) {

        try {

            Registry reg = LocateRegistry.getRegistry("localhost", 6789);

            HelloInterface obj = (HelloInterface) reg.lookup("HelloObj");

            System.out.println(obj.printHello());

            nhapvao=new Scanner(System.in);

            System.out.print("Nhap vao chuoi: ");

            String ten;

            ten=nhapvao.nextLine();

            System.out.println("\t---------------------Menu-------------------------");

            System.out.println("1.dem so luong tu \n2.xuat chuoi nguoc \n3.thoat \n");

            while (true) {

                System.out.print("chon  = ");

                int chon;

                chon=nhapvao.nextInt();

                switch (chon) {

                    case 1:

                            System.out.print("so luong tu trong chuoi la: ");

                            System.out.println(obj.countWords(ten));

                            System.out.println("\n");

                        break;

                        case 2:

                             StringBuilder str = new StringBuilder(ten);

                             System.out.print("chuoi duoc dao nguoc la:   ");

                             System.out.println(str.reverse().toString());

                             System.out.println("\n");

                        break;

                        case 3:

                        System.exit(1);

                        break;

                    default:

                    System.out.println("nhap sai");

                        break;

                }

            }

        } catch (Exception e) {

            // TODO: handle exception

        }

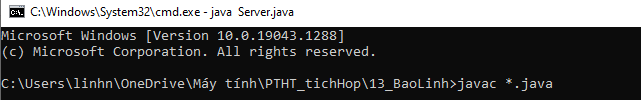
    }

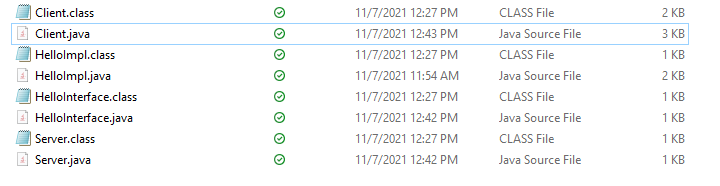
}

**Kết quả chương trình**

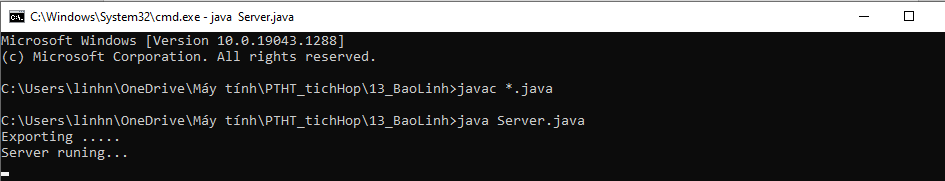
**Thực thi ứng dụng**

Chạy cmd gõ lệnh javc \*.java 🡪biên dịch các file có đuôi .java thành .class

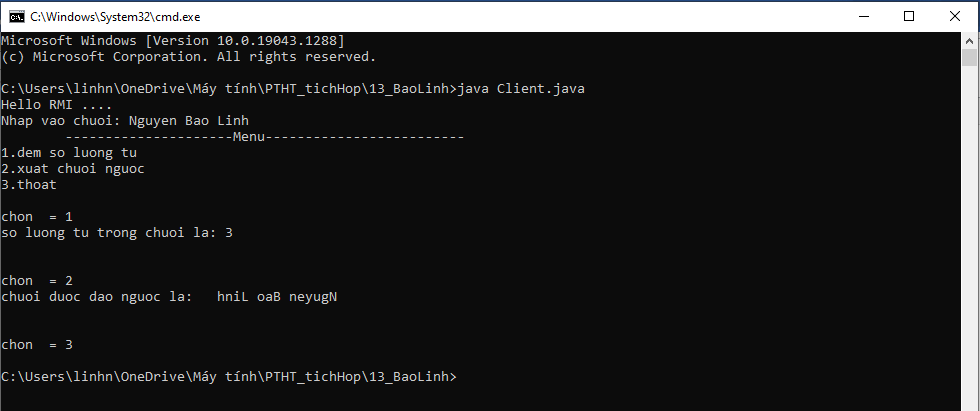




Tiếp tục gõ lệnh java Server.java để khởi động RMI Server



Mở 1 cmd mới gõ java Client.java để thực thi client

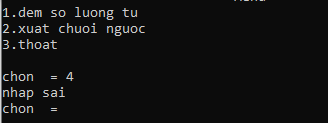


Client chọn 1 . Server trả về cho client số lượng từ trong chuổi.

Client chọn 2 . Server trả về cho client chuổi được đảo ngược.

Client chọn 3 .thoát chương trình.

Client chọn ngoài 1 2 3 in ra nhap sai và yêu cầu chọn lại từ 1->3



Link github :

https://github.com/NguyenLinh2000/PhatTrienHeThongTichHop/tree/RMI