# Cơ bản về lập trình mạng JavaSocket

## Tổng quan:

- Nó là sự kết hợp giữa Lập trình phần mềm( Java ) + Mạng máy tính + Cơ sở dữ liệu.

-

## Kiến thức về mạng

1. Địa chỉ IP : địa chỉ của máy mà nó kết nối đến máy khác để trung chuyển dữ liệu ( địa chỉ của một căn chung cư ) 192.168.255.255.
2. Cổng Port: địa chỉ của tiến trình trên máy mà nó đang chạy( giống số nhà của một căn chung cư). 8080.
3. Socket: là một giao diện để truyền thông,địa chỉ gồm địa chỉ IP của máy và Port của tiến trình đang chạy, được khởi tạo khi 1 tiến trình muốn truyền thông.
4. Mô hình lập trình mạng Client Server :

- Hiểu đơn giản là mỗi thằng sẽ cầm một src code của hệ thống trong đó:

+ Server là máy chủ mà Client muốn connect đến để lấy dữ liệu. Luôn chạy

+ Client có thể có nhiều Client khác nhau,các địa chỉ mạng khác nhau, chúng giao tiếp với Server qua đối tượng Socket để tiếp nhận dữ liệu và trung chuyển dữ liệu từ Server. Tg chạy hữu hạn

Trong ngữ cảnh lập trình ở mô hình đơn giản( theo nhóm chẳng hạn ) , Server được coi như phần chứa logic xử lý, Database. Client chứa các src code liên quan đến giao diện tương ứng ở các máy thành viên khác nhau, chúng connect với Server và thực hiện trung chuyển dữ liệu.

Trong những mô hình lớn, Client và Server có thể là các ứng dụng độc lập, Client chỉ truy cập vào dữ liệu của Server để thực hiện một chức năng nào đấy.

# Phần chính, lập trình với Socket.

1. Client/Server với TCP.

Cách tạo Socket với giao thức TCP:

Socket sk = new Socket( svh,svp)

Svh = String

Svp =int

Cách gửi nhân dữ liệu, tùy đề bài, nếu :

+ với luôngf dữ liệu BufferedWriter/ BufferedReader , ta tạo luồng gửi/ nhận như sau :

- BufferedWriter gui = new BufferedWriter( new OutputStreamWriter(sk.getOutputStream()));

- kiểu tạo đối tượng gửi nhận với In/out, sau đó ép kiểu sang đối tượng Buffered.

- Sau đó sử dụng phương thức write hoặc read với đối tượng gui/nhan để gửi chuỗi gì đấy lên.

- Nếu là String thì gửi UTF , int gửi Int

+ với luồng dữ liệu bình In/OUT THÌ cứ InputStream = sk.getIn..... thôi

TCP có 4 dạng :

Các dạng có chung phương pháp.

**Bước 1: Tạo Socket giao tiếp bằng cách:**

Khai báo chuỗi IP, Port ở bên ngoài như biến tạm của chương trình.

String ip =””

// có thể là localhost nếu test trên máy.

Int port = ;

Tạo Socket mới phải đặt trong trycatch.

Try{

Socket sk = new Socket(ip, port);

System.out.println(“oke”);

}

Catch(IOException e){

E.printStrack();

}

**Bước 2: oke sau đó định nghĩa luồng đọc ghi tương ứng .**

// Thằng InputStream/OutputStream nó sẽ là một đối tượng đọc ghi được trả về bởi một phương thức trong lớp Socket là getInputStream(), getOutputStream(); -> vì nó là luồng đọc ghi trên API Socket mà nên nó phải được cung cấp bởi Socket chứ.

// Thằng Socket

InputStream in = sk.getInputStream(),

OutputStream out = sk.getOutputStream(),

// sau đó ta có thể tự do đọc/ghi với luồng được định nghĩa này.

**Bước 3: Xử lý chuỗi gửi/ nhận ( chuỗi gửi/ nhận là tất cả đều phải đưa về String hết, sau đó getBytes() nó để đưa về Byte và chỉ dùng in.read() hoặc out.write() , sau đó in.flush();, out.flush();**

**Thôi.**

**Bước 4: Lúc gửi/ nhận phải đánh dấu out.flush()/in.flush()**

Luồng đọc ghi của

1. InputStream/OutputStream.

Đây là kiểu dữ liệu dạng Byte, nó sẽ gửi / nhận dữ liệu dạng Byte mà không quan tâm kiểu gì.

Luồng đọc ghi của thằng này sẽ được hỗ trợ bởi socket, qua phương thức getInputStream().

1. DataInputStream/ DataOutputStream.

Luồng đọc ghi của thằng này được hỗ trợ bởi Java, còn socket chỉ hỗ trợ luồng đọc ghi duy nhất là InputStream/OutputStream thôi ( tức là dạng byte)

Cho nên muốn định nghĩa luồng thằng này, phải bọc new DataInputStream ngoài socket.getInputStream() để mở rộng khả năng đọc của socket từ byte sang các dạng int, float, char

nó đọc dữ liệu có định dạng cơ bản như int, float, double, char,UTF

Khi đọc/ghi phải nói rõ là dữ liệu kiểu gì readInt, writeInt , UTF,.....

1. BufferedReader/BufferedWriter:

Tương tự , thằng này đọc dữ liệu dạng String Và đọc nhiều dòng kiểu readLine();

Cho nên luông đọc như sau: khởi tạo socket.getInputStream() để đọc ghi byte

Sau đó bọc new InputStreamReader(.....) để mở rộng khả năng chuyển thành đọc được ký tự,

Sau đó bọc new BufferedReader() để mở rộng khả năng chuyển thành đọc được chuỗi theo dòng.

Lúc đọc thì cứ read()

write() thôi

Đọc dữ liệu dạng chuỗi String, đặc biệt đọc dữ liệu theo kiểu từng dòng. Đọc với tốc độ cao hơn.