**SOCKET( Lập trình mạng )**

**A diagram of a computer network

Description automatically generated**

**Để kết nối**

Cần protocal và các port , address của server

**Cách làm:**

**Tạo package dao;**

Có các class là interface. VD CourseDao

* **Lưu ý** : trong class interface chỉ được chứa các phương thức trừu tượng (là phương thức không có phần thân)

**Tạo package dao.impl;**

Có các class triển khai interface .VD CourseImpl

* **Lưu ý :** class interface không thể tạo đối tượng nên cần có class triển khai

**Các bước tiếp theo :**

* Phải implement CourseDao
* private EntityManager em;
* Giải thích :
* Tạo constructor rỗng : trong phương thức này gõ

|  |
| --- |
| em=Persistence.createEntityManagerFactory(“truyentenper”).createEntityManager(); |

**Tạo package Server**

Có class Server

**Các bước tiếp theo :**

* Tạo main
* Tạo try-with-resources
* Trong try tạo port : cách tạo ServerSocket server= new ServerSocket(*port*)
* **Lưu ý :** port là số nguyên có 4 số
* Viết câu lệnh system.out
* Chạy thử nếu không được đổi port khác

Lúc này socket sẽ chờ client gửi tín hiệu tới

* Gõ while(true)
* Tức là socket sẽ chờ mãi mãi
* Trong vòng while gõ Socket socket = server.accept();
* Phương thức accept tức là chấp nhận cho Client gọi tới
* Gõ tiếp : Server temp = new Server();
* Gõ tiếp : new Thread(temp.new Handler(socket)).start();
* Do là có thể nhiều Client gửi tới nên cần phải có Thread để quản lí
* **Lưu ý :** Handler chưa có nên cần phải viết phương thức này

**Viết class Handle trong class Server**

Tạo class Handle implements Runable

Private Socket socket;

Tạo constructor đầy đủ tham số

**Tạo project Client**

**Tạo package Client**

Có class Client

**Các bước tiếp theo :**

* Tạo main
* Tạo try-with-resources
* Trong try tạo port : cách tạo Socket socket= new Socket(*“diaChiIp hoặc hostname của server”,port*)

Cách xem địa chỉ IP: vào cmd gõ ipconfig -all tìm hàng IPv4 Address (hạn chế dùng)

* **Lưu ý :** port là số nguyên có 4 số. Dùng IOException
* Chạy thử
* Tạo menu

**Cách làm bài :**

**Bên client:**

**Gửi kiểu int**

DataOutputStream out = new DataOutputStream(socket.getOutputStream());

out.writeInt(soMuonGui);

**Gửi kiểu String**

Sử dụng out ở trên

out.writeUTF(chuoiMuonGui)

out.ffush();

**Nhận kiểu Object**

ObjectInputStream in = new ObjectInputStream(socket.getInputStream());

in.readObject()

Bên server :

Viết trong phương thức run

Tạo try catch

Copy hàm while bên client bỏ qua

**Nhận kiểu int**

DataInputStream in = new DataInputStream(socket.getInputStream());

in.readInt();

**Nhận kiểu String**

Sử dung lại in

in.readUTF();

**Gửi kiểu Object**

ObjectOutputStream out =new ObjectOutputStream(socket.getOutputStream());

out.writeObject(truyenVoObject)

**Cách viết phương thức :**

**Trả về boolean**

**A computer screen shot of code

Description automatically generated**

**A computer screen shot of a code

Description automatically generated**

**Trả về kiểu String (chỉ dùng cho trường hợp thêm)**

**A computer screen shot of a code

Description automatically generated**

**Trả về đối tượng nếu không có trả về null**

**A screen shot of a computer program

Description automatically generated**

**Trả về danh sách**

**A computer screen shot of a code

Description automatically generated**

**Trả về danh sách nếu không có trả về null**

**A computer screen shot of a code

Description automatically generated**

**Trả về kiểu Map**

**A computer screen with white text

Description automatically generated**

**A computer screen shot of a computer code

Description automatically generated**

**Trả về kiểu Map xắp xếp theo key**

**A computer screen shot of a program

Description automatically generated**

**Trả về Map sắp xếp theo key nhưng với key là đối tượng**

**A computer screen with white text

Description automatically generated**

**A close-up of a computer code

Description automatically generated**

**Trả về Map nhưng giữ nguyên thứ tự truyền vào**

**A computer screen with text

Description automatically generated**

**Lộ trình viết câu truy vấn**

Select

From

Where

Group by

Having

Order by