**Chương cuối thì thi cuối kì: Lập trình mạng Thi cuối kì JPA + networking. Bài tập lớn cũng vậy.**

ODP dùng socket chỉ cần TCP dùng Để kiểm tra 2 máy thông nhau: bin lấy địa chỉ ip.

Cách lấy địa chỉ ip của máy: Lệnh ipconfig option all.

Gói java.net ServerSocket và socket

**Chương mới buổi cuối Socket.io**

**1.Dữ liệu**

byte ( ảnh âm thanh)

**Sử dụng**: Inputstream/ output stream -> bufferedInputStream/ bufferedOutputStream

**Luồng byte:** bufferedOutputStream Kiểm tra đọc hết file: read --> -1 thì end file.

**Nguyên tắc đọc file:** read 1 byte và write 1 byte.

Hoặc available--> trả về số byte còn có khả năng đọc tiếp.

**2.Luồng kí tự**

Reader/ writer -->>> BufferredRead/Writer

Tools: Scanner đọc 1 line( dòng ) / PrintWriter

**3. Luồng data Primitype**

byte,short,long,float,double,char,boolean,string

**Sử dụng**: DataInputStream/OutputStream

**4.Luồng Object( Đối tượng)**

Các lớp đối tượng thì phải implement interface Serializable

**Sử dụng**: ObjectInputStream/OutputStream.

**Wrapper**: Byte, Short,...

*Muốn dùng socket thì 2 máy được kết nối với nhau dạng mạng local.*

**Dùng hàm bin.**

**Tạo phía server của tcp: SeverSocket.**

Ping vào 1 cái port rỗng và sẵn sàng chờ tín hiệu connect. 4 số cuối của mình ở mạng địa chỉ

**Client: Bên phía client gửi connect -> server connect.**

Proxy socket 1 ứng dụng chạy thì luôn ping vô 1 port.

Port ở client là ngẫu nhiên.

**Muốn truy cập vào server: phải biết port, ip, path 1 client chạy 1 thread.**

Những lớp - object phải implement Serialable thì mới chuyển dữ liệu class đi đc.

*Chú ý các class con nằm trong class cha( có implement) thì cx phải khai báo implement.*

**Đóng gói. Tạo folder trực tiếp trong project Export**

-> java -> chọn ổ T --> đặt tên --> chỉ đóng gói entity, Hạn chế đóng gói.

*Đi thi add trực tiếp project.*

**Phần UDP:**

Tốc độ nhanh.

Chỉ gửi đi không cần quan tâm nhận đc chưa.

**Ứng dụng:** video stream.

DatagramSocket Server bin vô 1 port.

Ko quan tâm client nhận đc không.

Cấu trúc 1 package: IP, port, 1 array bytes

Có 2 phương thức:

Void send

Void recievie

**Port phạm vi:** 0 đến 2 mũ 16.

Bộ đệm đầy ->> đẩy về đích Bộ đệm lần cuối: chưa đầy nên dùng đóng luồng để đẩy tất cả về đích.

**Java RMI Kiến trúc**

*Tạo 1 app rmi server cung cấp services.*

*Tuần 10 kiểm tra java RMI Thi cuối kì là socket*

*Thi cuối kì Mô hình lớp -> ánh xạ sang đb.*

*Dùng maria hoặc sql server.*

*Thi cuối kì socket Đc dùng copilot*

SOCKET CORE