class TCPClientSocket

{

public:

SOCKET hSocket;

SOCKADDR\_IN servAddr;

public:

TCPClientSocket(char \* arg[]);

~TCPClientSocket();

bool Connect();

int Send(char \* message, int bufsize);

int Receive(char \* message, int len);

};

클라이언트 소켓을 생성합니다.

초기화 하면서 socket()를 호출합니다.

Connect() 함수는 connect() 함수에 대한 래퍼 함수 입니다.

Send() 함수는 send() 함수에 대한 래퍼함수 입니다.

Receive() 함수는 recv() 함수에 대한 래퍼함수 입니다.

class TCPClientManager

{

public:

static const int NAME\_LEN = 20;

static const int BUF\_SIZE = 1024;

static char name[NAME\_LEN];

private:

WSADATA wsaData;

HANDLE thread\_Receive;

HANDLE thread\_Send;

public:

TCPClientManager(char \* n);

~TCPClientManager();

void StartReceiveThread(TCPClientSocket \* sock, unsigned (\_stdcall \* threadMain)(void \*));

void StartSendThread(TCPClientSocket \* sock, unsigned(\_stdcall \* threadMain)(void\*));

void WaitForUntilDone();

};

상수 및 WSADATA, 수신 스레드와 송신 스레드를 관리합니다.

Static 상수 목록은 캡슐화에 문제가 있을 수 있습니다.

name 변수는 다른 상용 프로그램을 만들 경우에는 사용되지 않을 것입니다.

StartReceiveThread() 함수는 수신 스레드를 등록합니다.

스레드는 직접 디자인 하여 등록 후 사용하도록 하십시오.

StartSendThread() 함수는 송신 스레드를 등록합니다.

스레드는 직접 디자인 하여 등록 후 사용하도록 하십시오.

등록하는 예제를 제시하겠습니다.

client->StartReceiveThread(socket, RecvMsg);

client->StartSendThread(socket, SendMsg);