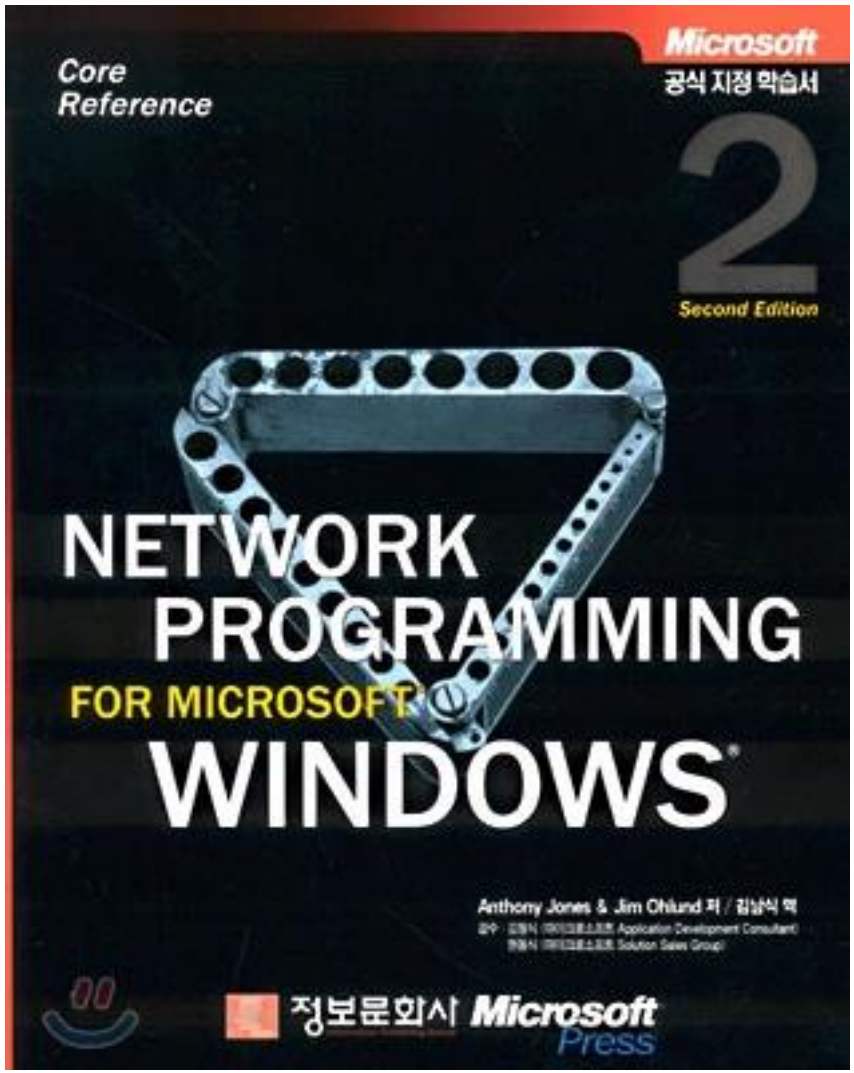
Network Programming for Windows 00:

# Introduction to Course

jintaeks@dongseo.ac.kr
Division of Digital Contents, DSU
March 2018

# Outline

- ✓ We will cover 5 chapters of the text book.
- ✓ 01 Introduction to Winsock
- ✓ 02 Winsock Design
- ✓ 03 Internet Protocol
- ✓ 04 Other Supported Protocols
- ✓ 05 Winsock IO Methods



# IOCP Server

- ✓ We will develop a simple application using IOCP and Winsock.
- ✓ We will develop a real game server.

# Prepare development environment

- ✓ Download and install Visual Studio community 2013
- ✓ Download boost library 1.59
- ✓ Bookmark
  - github site for source
  - Textbook site

# Windows Event Objects

- ✓ An *event object* is a synchronization object whose state can be explicitly set to **signaled** by use of the **SetEvent** function.
- ✓ Manual-reset event
  - An event object whose state remains **signaled** until it is explicitly reset to **nonsignaled** by the **ResetEvent** function.
- ✓ Auto-reset event
  - An event object whose state changed to signaled by SetEvent() automatically change to nonsignaled state(at which time **the system automatically sets the state to nonsignaled**).

```
DWORD WINAPI WaitForMultipleObjects(
    _In_       DWORD  nCount,
    _In_ const HANDLE *lpHandles,
    _In_       BOOL   bWaitAll,
    _In_       DWORD  dwMilliseconds
    );
```

- ✓ ***nCount*** [in]: The number of object handles in the array pointed to by *lpHandles.*
- ✓ Return Value
  - **WAIT_OBJECT_0** to (**WAIT_OBJECT_0** + ***nCount***– 1)
  - If *bWaitAll* is **TRUE**, the return value indicates that the state of all specified objects is signaled.
  - If *bWaitAll* is **FALSE**, the return value minus **WAIT_OBJECT_0** indicates the *lpHandles* array index of the object that satisfied the wait

# Practice

- ✓ Samples → prerequisite → Win32Thread projects
- ✓ Write a program which prints from 1 to 10, the main thread prints the odd numbers, the second thread prints the even numbers.
- ✓ Deadlock in threads.

# Prevent copy action: boost::noncopyable

```
class KGlobalSingleton
{
private:
        KGlobalSingleton( const KGlobalSingleton& );
        KGlobalSingleton& operator=( const KGlobalSingleton& );
...
```

# Singleton: KGlobalSingleton<T>

```
template<typename T>
class KGlobalSingleton
{
public:
    typedef T           Type;
    inline static Type& Singleton() { return KSingletonCreator::Access(); }
    inline              KGlobalSingleton() {}

private:
                        KGlobalSingleton( const KGlobalSingleton& );
                        KGlobalSingleton& operator=( const KGlobalSingleton& );

protected:
    class KSingletonCreator
    {
    public:
        inline KSingletonCreator()
        {
            Access();
        }//KSingletonCreator()
```

```
class KSingletonCreator
    {
    public:
        inline KSingletonCreator()
        {
            Access();
        }//KSingletonCreator()

        __declspec(noinline) static Type& Access( )
        {
            static Type ref;
            return ref;
        }//Access()
    };//struct KSingletonCreator

private:
    static KSingletonCreator     ms_kSingletonCreator;
};//template<typename T>

template <typename T>
/*static*/ typename KGlobalSingleton<T>::KSingletonCreator  KGlobalSingleton<T>::ms_kSingletonCreator;
```

# KBaseSingleton<T>

```
template<typename T>
class KBaseSingleton
{
public:
    template<typename U>
    static void         CreateInstance( DWORD dwParam ) { ms_instance.reset( new U() ); }

protected:
    static boost::shared_ptr<T>
                        ms_instance;

public:
    static boost::shared_ptr<T>
                        Instance() { return ms_instance; }
};

template<typename T>
boost::shared_ptr<T> KBaseSingleton<T>::ms_instance = nullptr;
```

# References

- ✓ http://www.winsocketdotnetworkprogramming.com/winsock2programming/
- ✓ https://github.com/GP101/DiconServer
- ✓ https://sourceforge.net/projects/boost/files/boost-binaries/
- ✓ https://imagine.microsoft.com/ko-kr/Catalog/Product/530

# MY **BRIGHT** FUTURE

DSU Dongseo University 동서대학교

DSU Dongseo Univ 동서대학교