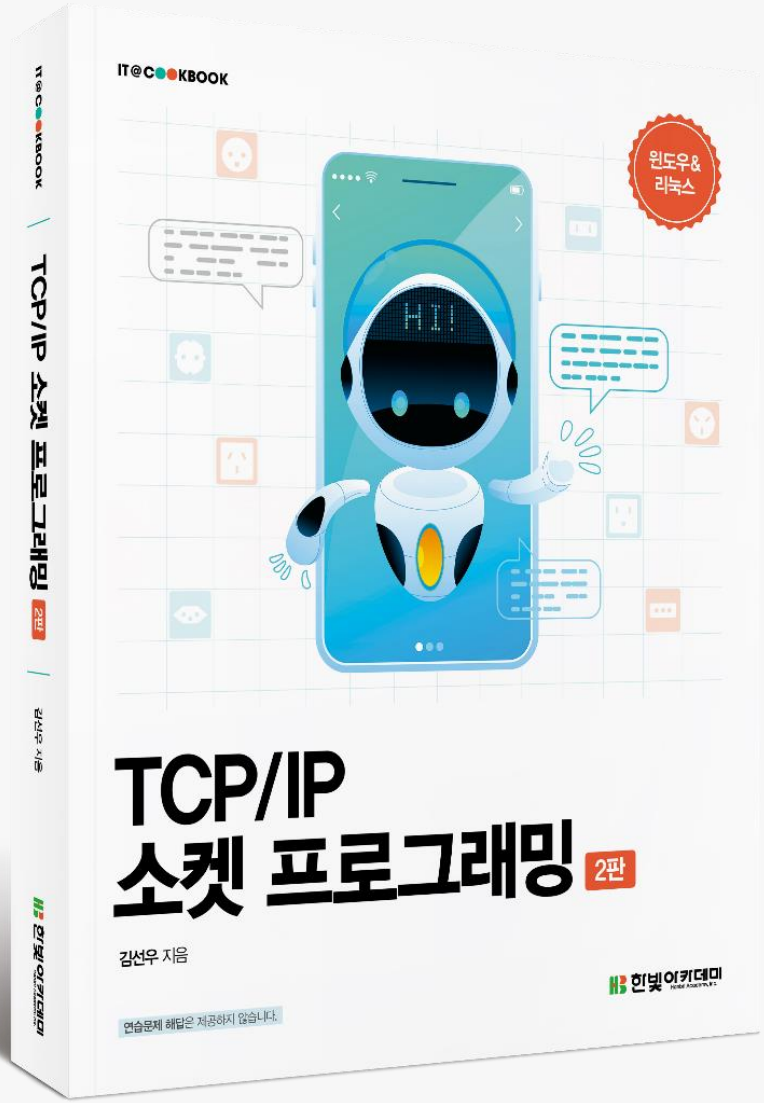


Chapter 02 소켓 시작하기

학습목표

- 소켓 함수의 오류 처리 방법을 익힌다.
- 소켓 초기화와 종료 방법을 익힌다.
- 소켓을 생성하고 닫는 방법을 익힌다.

목차

01 오류 처리

오류 처리 유형

① 오류를 처리할 필요가 없는 경우

- 리턴값이 없거나 호출 시 항상 성공하는 일부 소켓 함수

② 리턴값만으로 오류를 처리하는 경우

- WSAStartup() 함수

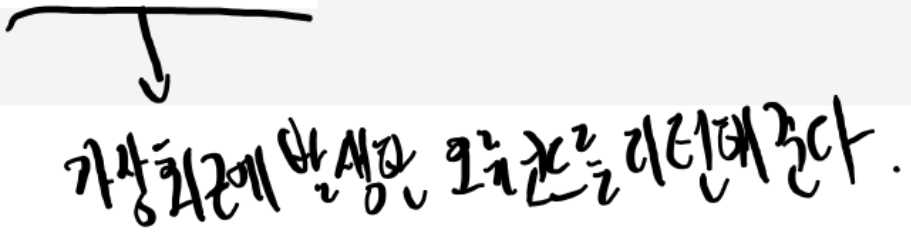
③ 리턴값으로 오류 발생을 확인하고, 구체적인 내용은 오류 코드로 확인하는 경우

- 대부분의 소켓 함수

윈도우 오류 처리 (1)

윈도우

```
#include <winsock2.h>  // WSAGetLastError() 함수가 선언되어 있다.
...
if (소켓 함수(...) == 실패) {
    printf(WSAGetLastError()에 해당하는 오류 메시지);
}
```



윈도우 오류 처리 (2): 오류 코드를 문자열로 바꾸기 (1)

FormatMessage() 함수

윈도우

```
#include <windows.h>
DWORD FormatMessage(
    ❶ DWORD dwFlags,
      LPCVOID lpSource,
    ❷ DWORD dwMessageId,
    ❸ DWORD dwLanguageId,
    ❹ LPTSTR lpBuffer,
      DWORD nSize,
      va_list *Arguments
);                                  성공: 오류 메시지의 길이, 실패: 0
```

윈도우 오류 처리 (3): 오류 코드를 문자열로 바꾸기 (2)

- **err_quit() 함수 정의**

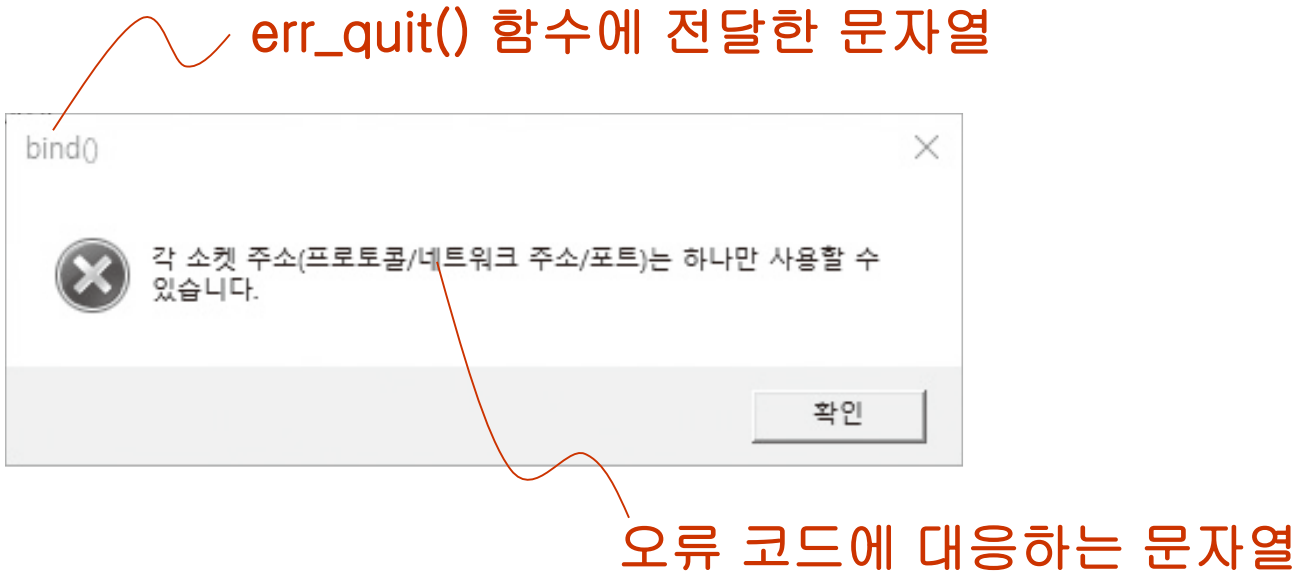
```
void err_quit(const char *msg)
{
    LPVOID lpMsgBuf;
    FormatMessageA(
        FORMAT_MESSAGE_ALLOCATE_BUFFER
        | FORMAT_MESSAGE_FROM_SYSTEM,
        NULL, WSAGetLastError(),
        MAKELANGID(LANG_NEUTRAL, SUBLANG_DEFAULT),
        (char *)&lpMsgBuf, 0, NULL);
    MessageBoxA(NULL, (const char *)lpMsgBuf, msg, MB_ICONERROR);
    LocalFree(lpMsgBuf);
    exit(1);
}
```

윈도우 오류 처리 (4): 오류 코드를 문자열로 바꾸기 (3)

- **err_quit() 함수 사용 예**

```
if (socket(...) == INVALID_SOCKET) err_quit("socket()");
if (bind(...) == SOCKET_ERROR) err_quit("bind()");
```

- **err_quit() 함수의 오류 메시지**



윈도우 오류 처리 (5): 오류 코드를 문자열로 바꾸기 (4)

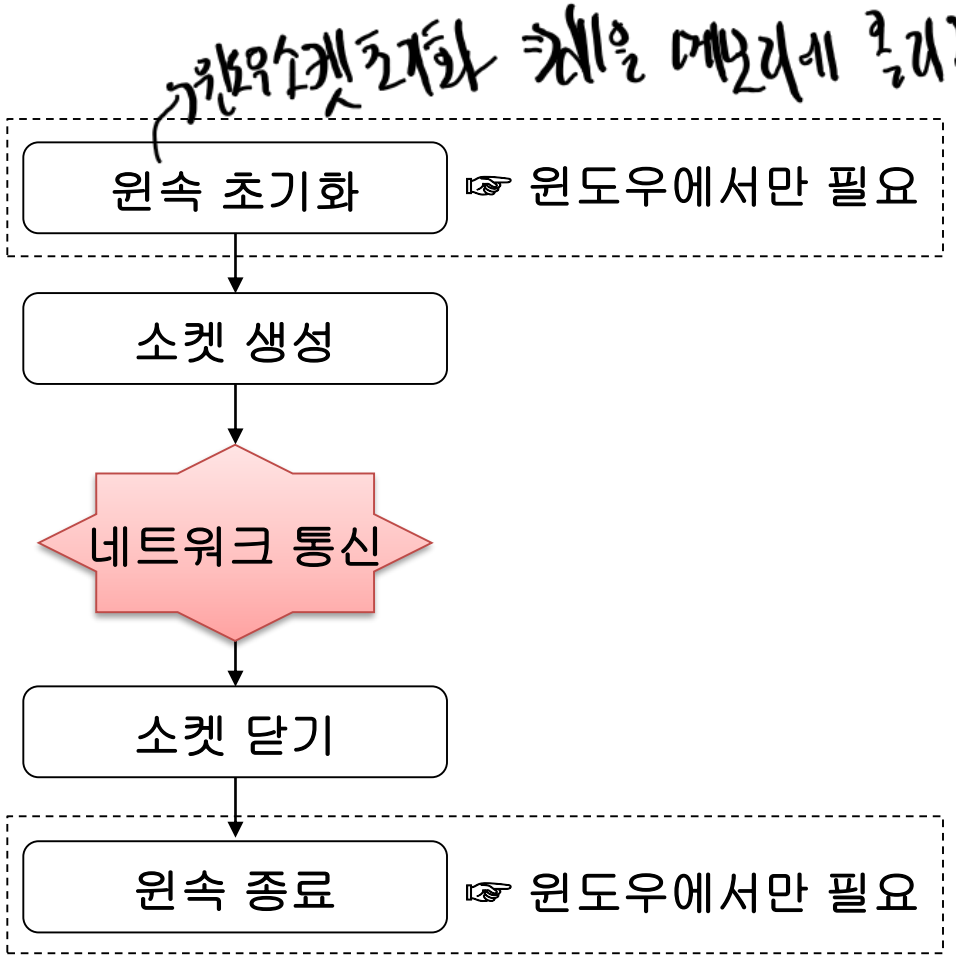
- **err_display() 함수 정의**

```
void err_display(const char *msg)
{
    LPVOID lpMsgBuf;
    FormatMessageA(
        FORMAT_MESSAGE_ALLOCATE_BUFFER
         | FORMAT_MESSAGE_FROM_SYSTEM,
        NULL, WSAGetLastError(),
        MAKELANGID(LANG_NEUTRAL, SUBLANG_DEFAULT),
        (char *)&lpMsgBuf, 0, NULL);
    printf("[%s] %s\n", msg, (char *)lpMsgBuf);
    LocalFree(lpMsgBuf);
}
```

02 소켓 초기화와 종료

윈속 초기화와 종료 (1)

- **윈속 응용 프로그램의 공통 구조**



윈속 초기화와 종료 (2)

윈속 초기화

```
int WSAStartup (
    WORD wVersionRequested,
    LPWSADATA lpWSAData
);
```

성공: 0, 실패: 오류 코드

- wVersionRequested
  - 프로그램이 요구하는 최상위 윈속 버전. 하위 8비트에 주 버전을, 상위 8비트에 부 버전을 넣어 전달
- lpWSAData
  - 윈도우 운영체제가 제공하는 윈속 구현에 관한 정보를 얻을 수 있음(거의 사용 안 함)

윈속 초기화와 종료 (3)

윈속 종료

int WSACleanup(void);

성공: 0, 실패: SOCKET_ERROR

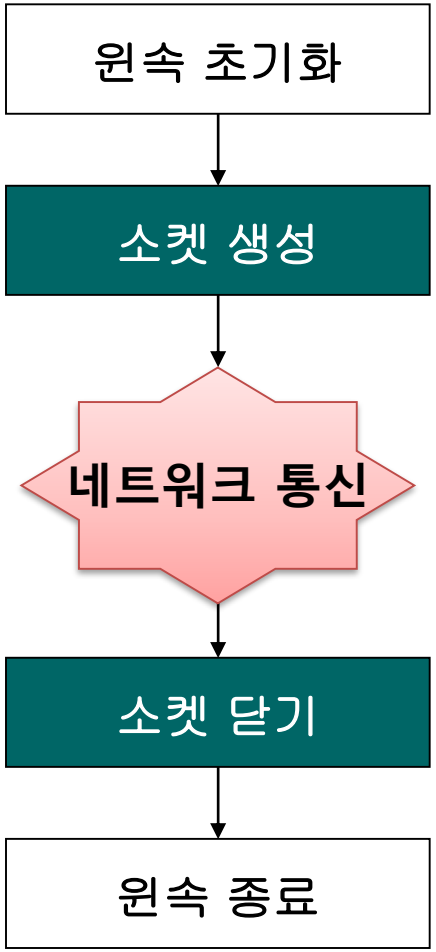
윈속 초기화와 종료 (4)

- **실습 2-1 윈속 초기화와 종료하기**
  - InitSocket.cpp
  - https://github.com/promche/TCP-IP-Socket-Prog-Book-2nd/blob/Source/Windows/Chapter02/InitSocket/InitSocket.cpp

03 소켓 생성과 닫기

소켓 생성과 닫기 (1)

- 윈속 응용 프로그램의 공통 구조



소켓 생성과 닫기 (2)

소켓 생성

```
#include <winsock2.h>    [윈도우]
SOCKET socket (
    int af,              // 주소 체계
    int type,            // 소켓 타입
    int protocol         // 프로토콜
);
                성공: 새로운 소켓, 실패: INVALID_SOCKET
```

```
#include <sys/types.h>    [리눅스]
#include <sys/socket.h>
int socket (
    int domain,      // 주소 체계
    int type,        // 소켓 타입
    int protocol     // 프로토콜
);
                                성공: 새로운 소켓, 실패: -1
```

- 사용자가 요청한 프로토콜을 사용해 통신할 수 있도록
  내부적으로 리소스를 할당하고,
  이에 접근할 수 있는
  일종의 핸들값인 소켓 디스크립터(socket descriptor)를 리턴

소켓 생성과 닫기 (3)

주소 체계

```
...
#define AF_INET 2       // Internetwork: UDP, TCP, etc.
...
#define AF_INET6 23     // Internetwork Version 6
...
#define AF_BTH 32       // Bluetooth RFCOMM/L2CAP protocols
...
```

소켓 생성과 닫기 (4)

소켓 타입

- 사용할 프로토콜의 특성

소켓 타입	특성
SOCK_STREAM	신뢰성 있는 데이터 전송 기능 제공, 연결형 프로토콜
SOCK_DGRAM	신뢰성 없는 데이터 전송 기능 제공, 비연결형 프로토콜

→ TCP

→ UDP

- TCP와 UDP 프로토콜 사용을 위한 설정 (1)

사용할 프로토콜	주소 체계	소켓 타입
TCP	AF_INET 또는 AF_INET6	SOCK_STREAM
UDP		SOCK_DGRAM

소켓 생성과 닫기 (5)

프로토콜

- 주소 체계와 소켓 타입이 같더라도 해당 프로토콜이 두 개 이상 존재할 경우 프로토콜을 명시적으로 지정
  - TCP와 UDP 프로토콜 사용을 위한 설정 (2)

사용할 프로토콜	주소 체계	소켓 타입	프로토콜
TCP	AF_INET 또는 AF_INET6	SOCK_STREAM	IPPROTO_TCP
UDP		SOCK_DGRAM	IPPROTO_UDP

  - TCP와 UDP 프로토콜 사용을 위한 설정 (3)

사용할 프로토콜	주소 체계	소켓 타입	프로토콜
TCP	AF_INET 또는 AF_INET6	SOCK_STREAM	0
UDP		SOCK_DGRAM	0

소켓 생성과 닫기 (6)

소켓 닫기

```
#include <winsock2.h>    [윈도우]
int closesocket (
    SOCKET s
);
```

성공: 0, 실패: SOCKET_ERROR

```
#include <unistd.h>        [리눅스]
int close (
    int fd
);
```

성공: 0, 실패: -1

- 소켓을 닫고 관련 리소스를 운영체제에 반환

소켓 생성과 닫기 (7)

- **실습 2-2** 소켓 생성과 닫기
  - InitSocket.cpp
  - [윈도우] https://github.com/promche/TCP-IP-Socket-Prog-Book-2nd/blob/Source/Windows/Chapter02/InitSocket/InitSocket.cpp
  - [리눅스] https://github.com/promche/TCP-IP-Socket-Prog-Book-2nd/blob/Source/Linux/Chapter02/InitSocket.cpp