

## □ 예제 프로그램 2: LabProject01(윈도우 크기 변경하기)

프로그램이 실행될 때 주 윈도우(Main Window)의 클라이언트 영역의 크기를 원하는 크기(예를 들어, 640x480)가 되도록 하자. 또한 주 윈도우의 크기를 변경하거나 최대화를 할 수 없도록 윈도우의 스타일을 변경하자.

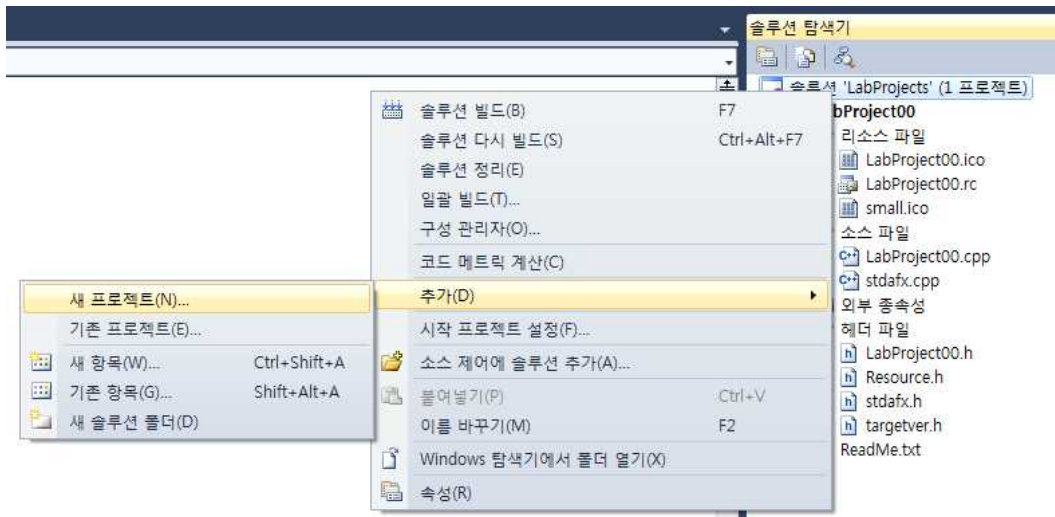


그림 1. 새 프로젝트 추가

### ① 새로운 프로젝트 추가

“LabProjects” 솔루션을 열고 솔루션 탐색기에서 <그림 1>과 같이 마우스 오른쪽 버튼으로 『솔루션 LabProjects』를 선택하고 메뉴에서 『추가』, 『새 프로젝트』를 차례로 선택한다. 그러면 『새 프로젝트 대화상자』가 나타난다.

### ② 새 프로젝트 대화상자

프로젝트 이름 “LabProject01”을 입력하고 『확인』을 선택한다.

### ③ “LabProject01.cpp” 파일을 수정

이제 “LabProject01.cpp” 파일을 수정하도록 하자.

#### ❶ InitInstance() 함수 변경

“LabProject01.cpp” 파일의 InitInstance 함수를 다음과 같이 변경하여 보자. 다음 코드는 주

윈도우의 크기를 조절하지 못하도록 윈도우 스타일을 설정한다. 그리고 최대화 버튼 (Maximize Button)을 사용할 수 없도록 윈도우 스타일을 설정한다. 주 윈도우의 클라이언트 영역의 크기가 (640x480)이 되도록 윈도우의 크기를 조절하여 주 윈도우를 생성한다.

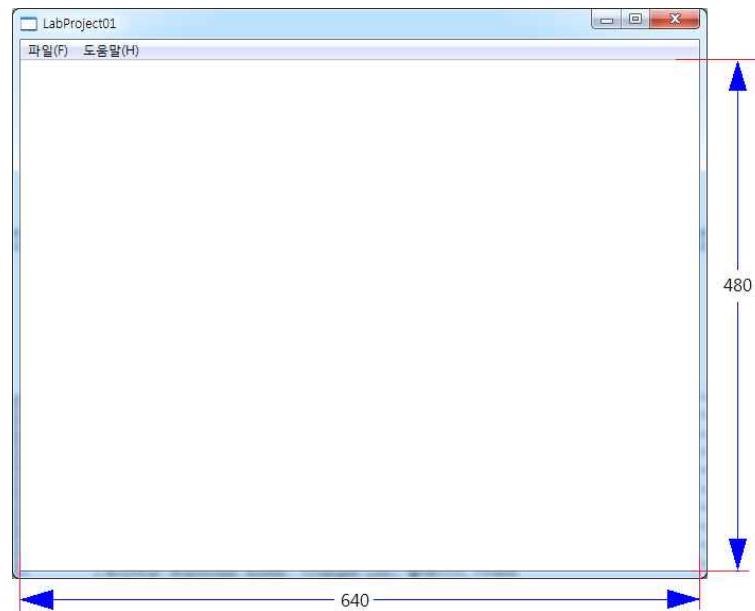


그림 2. 클라이언트 영역 크기

```

BOOL InitInstance(HINSTANCE hInstance, int nCmdShow)
{
    DWORD dwStyle = WS_OVERLAPPED | WS_CAPTION | WS_MINIMIZEBOX | WS_SYSMENU |
    WS_BORDER;
    RECT rc = { 0, 0, 640, 480 };
    AdjustWindowRect(&rc, dwStyle, FALSE);
    HWND hwnd = CreateWindow(szWindowClass, szTitle, dwStyle, CW_USEDEFAULT,
    CW_USEDEFAULT, rc.right - rc.left, rc.bottom - rc.top, NULL, NULL, hInstance,
    NULL);
    if (!hwnd) return FALSE;

    ShowWindow(hwnd, nCmdShow);
    UpdateWindow(hwnd);

    return TRUE;
}

```

## ■ AdjustWindowRect() 윈도우 API 함수

```

BOOL WINAPI AdjustWindowRect(LPRECT lpRect, DWORD dwStyle, BOOL bMenu);

```

윈도우가 원하는 클라이언트 영역의 크기를 가지도록 윈도우의 크기를 계산하여 준다.

- lpRect  
원하는 클라이언트 영역의 크기를 나타내는 사각형을 지정한다. 윈도우의 새로운 크기를

lpRect를 통하여 반환한다.

- dwStyle

윈도우의 스타일을 지정한다.

- bMenu

윈도우가 메뉴를 가지는가의 여부를 지정한다. bMenu가 참이면 윈도우가 메뉴를 가지고 있음을 의미한다.

위의 코드를 실행하면 <그림 2>와 같이 주 윈도우의 클라이언트 영역의 실제 크기가 640x480 화소 크기가 된다.