

개정3판

Visual
Studio
2017

쉽게 풀어쓴

C언어
EXPRESS

천인국 지음

제2장 프로그램 작성 과정





프로그램 개발 과정

요구사항 분석

무엇을
만들 것인가를
결정한다.



설계

알고리즘을
설계한다.



구현

개발 도구를
사용하여
소스 코드를
작성한다.



테스팅

여러가지
경우에 대하여
실행하여 본다.



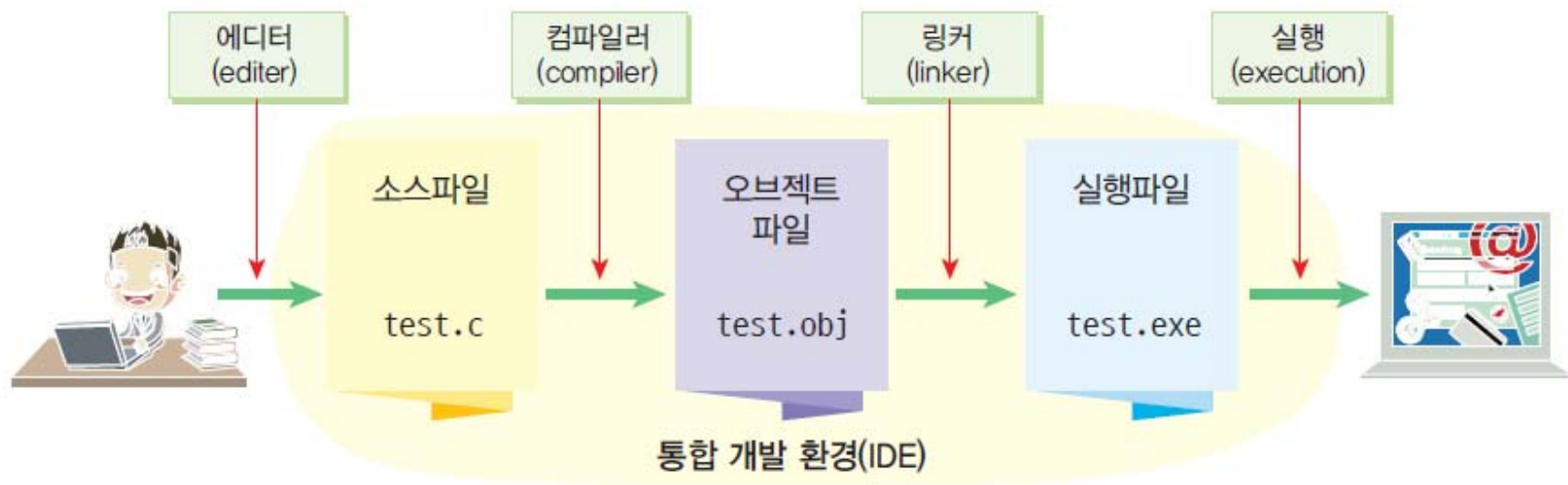
유지보수

사용자의 추가
요구사항을
반영한다.





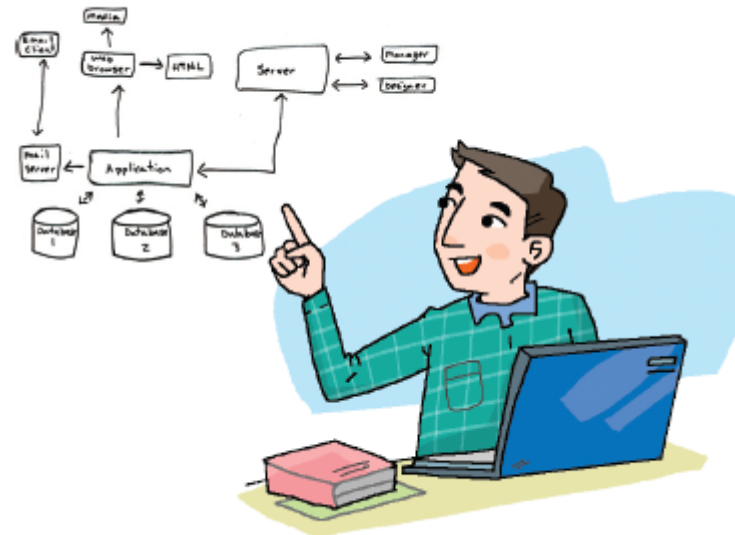
프로그램 개발 과정





설계

- 문제를 해결하는 알고리즘을 개발하는 단계
 - 순서도와 의사 코드를 도구로 사용
- 알고리즘은 프로그래밍 언어와는 무관
- 알고리즘은 원하는 결과를 얻기 위하여 밟아야 하는 단계에 집중적으로 초점을 맞추는 것





소스 작성

- 알고리즘의 각 단계를 프로그래밍 언어를 이용하여 기술
- *소스 프로그램(source program)*: 알고리즘을 프로그래밍 언어의 문법에 맞추어 기술한 것
 - 소스 프로그램은 주로 텍스트 에디터나 통합 개발 환경을 이용하여 작성
 - 소스 파일 이름 예: test.c

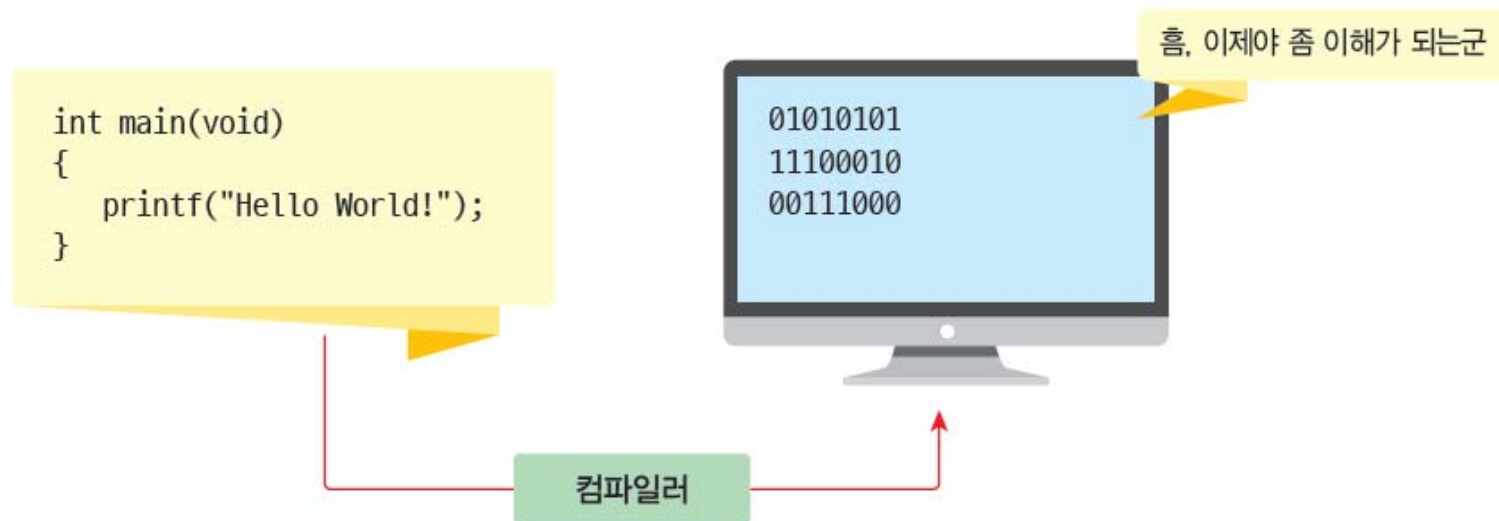


```
int main(void)
{
    printf("Hello World!");
}
```



컴파일

- 소스 프로그램을 오브젝트 파일로 변환하는 작업
- 오브젝트 파일 이름 예: test.obj





컴파일 오류

- 컴파일 오류(compile error): 문법 오류
(예) He go to school;

The goes to
school.



오류가 있어요. 수정해주세요!

오류발생!!

```
int main(void)
{
    printf>Hello World!</pre>
```



오류가 있어요. 수정해주세요!

오류발생!!



링크

- 컴파일된 오브젝트(obj) 프로그램을 라이브러리와 연결하여 실행 프로그램을 작성하는 것

실행 파일 이름 예: test.exe

라이브러리(library): 프로그램에서 많이 사용되는 기능을 미리 작성해 놓은 것

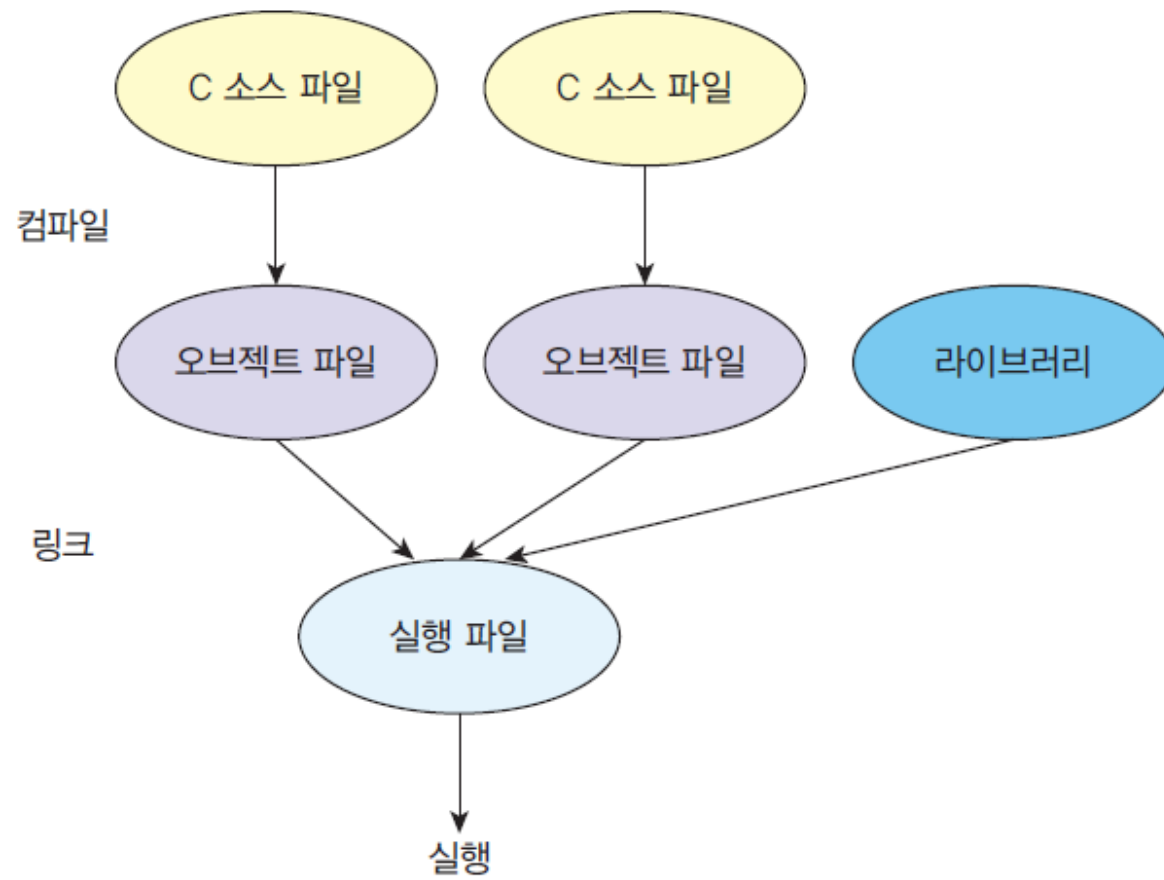
- (예) 입출력 기능, 파일 처리, 수학 함수 계산

링커(linker): 링크를 수행하는 프로그램





링크





실행 및 디버깅

□ 실행 시간 오류(runtime error):

예

- 0으로 나누는 것
- 잘못된 메모리 주소에 접근하는 것

□ 논리 오류(logical error):

문법은 틀리지 않았으나 논리적으로 정확하지 않는 것

예

- ① 그릇1과 그릇2를 준비한다.
- ② 그릇1에 밀가루, 우유, 계란을 넣고 잘 섞는다.
- ③ 그릇2를 오븐에 넣고 30분 동안 350도로 굽는다.





디버깅

- 소스에 존재하는 오류를 수정하는 것





디버깅의 유래

- 1945년 마크 II 컴퓨터가 릴레이 장치에 낱아든 나방 때문에 고장을 일으켰고 이것을 “컴퓨터 버그(bug: 벌레)”라고 불렀다. 여성 컴퓨터 과학자인 그레이스 호퍼가 나방을 채집해 기록에 남기고 이를 “디버깅(debugging)”작업이라고 보고하였다.



9/2
9/9

0800 Autom started
1000 stopped - autom ✓
1500 HP - MC 2.13047695
PRO 2 2.13047695
conv 2.13047695
Relays 6-2 in 0.24 full speed speed test
1m relay
Relays changed
1100 Started Cosine Tape (Sine check)
1525 Started Multi-Angle Test.
1545 Relay #70 Panel F
(motion relay).
1630 Autom started.
1700 closed down.

First actual case of bug being found.

Relay #70
Sine
Cosine



소프트웨어의 유지 보수

- 소프트웨어(Software: SW)의 유지 보수가 필요한 이유
 - 디버깅 후에도 버그가 남아 있을 수 있기 때문
 - 소프트웨어가 개발된 다음에 사용자의 요구가 추가될 수 있기 때문
- 유지 보수 비용이 전체 SW 개발 비용의 50% 이상을 차지

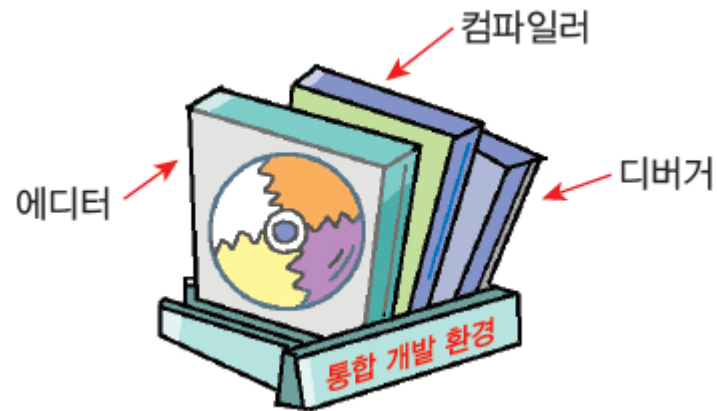




통합 개발 환경

□ 통합 개발 환경

- Integrated Development Environment (IDE)
- 에디터 + 컴파일러 + 디버거





통합 개발 환경의 예

- 비주얼 스튜디오: 마이크로소프트
- 이클립스(eclipse): 오픈 소스 프로젝트
- Dev-C++: 오픈 소스 프로젝트





비주얼 스튜디오 버전

- 커뮤니티 버전(Visual Studio Community)
 - 강력한 IDE, 학생, 오픈 소스 제공자 및 개인을 위해 무료로 제공
- 프로페셔널 버전(Visual Studio Professional)
 - 소규모 팀에 적합한 Professional IDE
- 엔터프라이즈 버전(Visual Studio Enterprise)
 - 모든 규모의 팀을 위한 확장성 뛰어난 통합형 솔루션

- 다운로드 링크

<https://visualstudio.microsoft.com/ko/downloads/>

- 교재는 2018년 개정판으로 Visual Studio 2017 버전을 사용하는 것으로 되어 있으나 현재 2019 버전을 다운로드 받아서 사용할 수 있음
- 프로그래밍 하는 것은 차이가 없으므로 Visual Studio 2019을 받아서 설치하면 됨



비주얼 스튜디오 설치

Windows 및 Mac용 Visual Studi x +

visualstudio.microsoft.com/ko/downloads/

Microsoft | Visual Studio 제품 다운로드 구입 지원 구독자 액세스 무료 Visual Studio Microsoft 전체 로그인

다운로드

Visual Studio 2019

버전 16.4

[릴리스 정보 >](#)

Android, iOS, Windows, 웹 및 클라우드에 대해 모든 기능을 갖춘 IDE(통합 개발 환경)

[버전 비교 >](#)
[오프라인 설치 방법 >](#)

커뮤니티

강력한 IDE, 학생, 오픈 소스 제공자 및 개인을 위해 무료로 제공

[무료 다운로드](#) ↓

[Preview 다운로드](#) ↓

Professional

소규모 팀에 적합한 Professional IDE

[평가판](#) ↓

[Preview 다운로드](#) ↓

Enterprise

모든 규모의 팀을 위한 확장성 뛰어난 통합형 솔루션

[평가판](#) ↓

[Preview 다운로드](#) ↓

클릭

Visual Studio Code

[릴리스 정보 >](#)

필요에 맞게 조정할 수 있고 무료로 제공되는 빠른 오픈 소스 코드 편집기

[무료 다운로드](#) ↓

Mac용 Visual Studio

버전 8.4

[릴리스 정보 >](#)

.NET을 사용하여 iOS, Android 및 웹 앱 및 게임 개발

[무료 다운로드](#) ↓


피드백



비주얼 스튜디오 설치

Visual Studio를 다운로드해 주셔서 감사합니다.
다운로드가 곧 시작됩니다. 다운로드가 시작되지 않은 경우에는 [여기를 클릭하여 다시 시도하세요](#)


Visual Studio를 사용하여 개발 시작



설치 지침

설치에 대한 도움이 필요하세요? Visual Studio 설치의 도움말 설명서를 확인하세요.

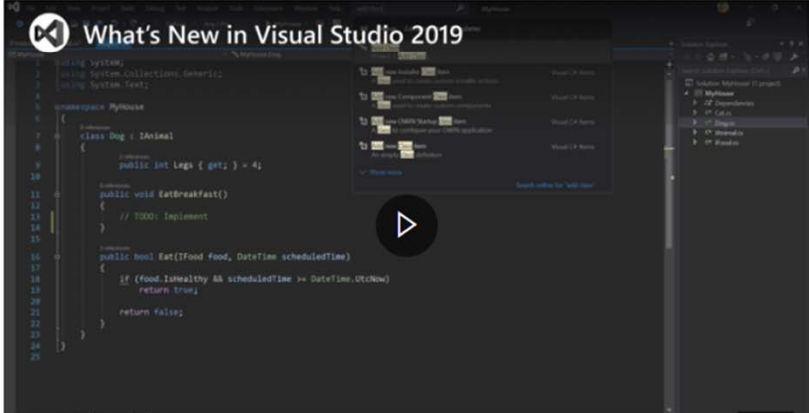
[지침 >](#)




빠른 시작 가이드

C++ 콘솔을 사용하여 첫 번째 애플리케이션을 개발하는 방법에 대한 자습서를 사용해 보세요.

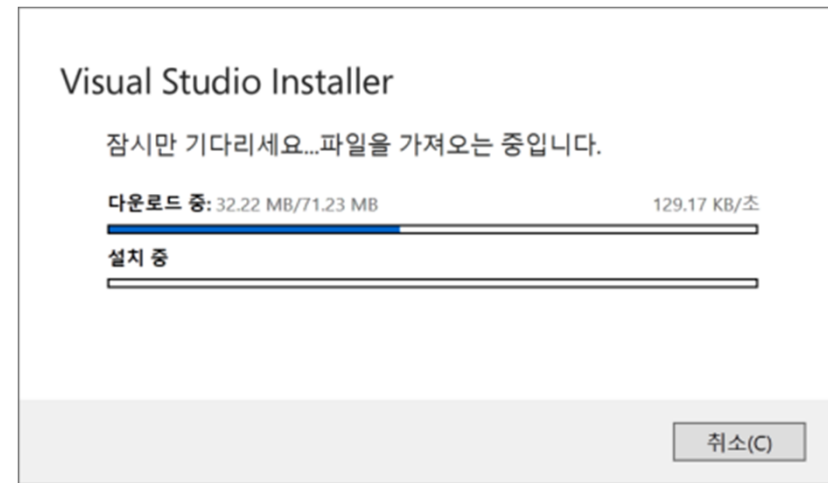
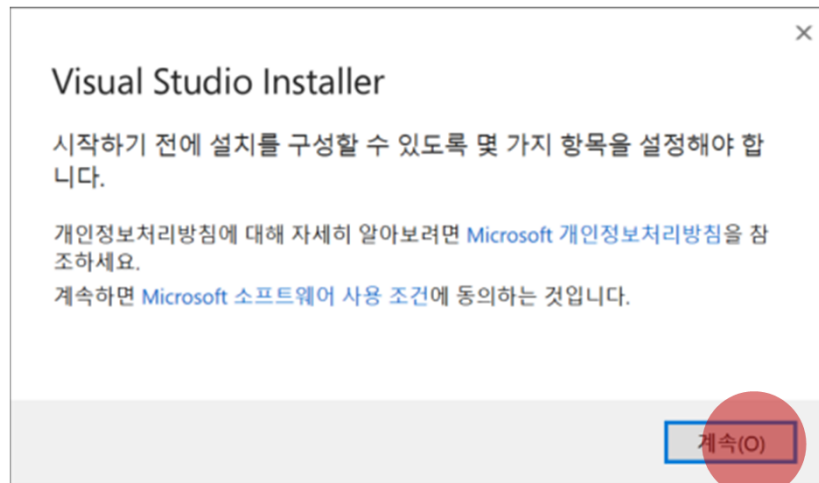
[계산기 앱 자습서 >](#)



**피드백**



비주얼 스튜디오 설치



클릭



비주얼 스튜디오 설치

설치 중 — Visual Studio Community 2019 — 16.4.6

워크로드 개별 구성 요소 언어 팩 설치 위치

웹 및 클라우드 (4)

- ☐ ASP.NET 및 웹 개발
Docker 지원이 포함된 ASP.NET Core, ASP.NET, HTML/JavaScript 및 컨테이너를 사용하여 웹 애플리케이션을...
- ☐ Python 개발
Python에 대한 편집, 디버깅, 대화형 개발 및 소스 제어입니다.
- ☐ Azure 개발
.NET Core 및 .NET Framework를 사용하여 클라우드 앱을 개발하고 리소스를 만들기 위한 Azure SDK, 도구 및 프로젝트...
- ☐ Node.js 개발
비동기 이벤트 구동 JavaScript 런타임인 Node.js를 사용하여 확장 가능한 네트워크 애플리케이션을 빌드합니다.

데스크톱 및 모바일 (5)

- ☐ .NET 데스크톱 개발
.NET Core and .NET Framework와 함께 C#, Visual Basic 및 F#를 사용하여 WPF, Windows Forms 및 콘솔 애플리케이션을...
- ☒ C++를 사용한 데스크톱 개발
MSVC, Clang, CMake 또는 MSBuild 등 선택한 도구를 사용하여 Windows용 최신 C++ 앱을 빌드합니다.
- ☐ 유니버설 Windows 플랫폼 개발
C#, VB 또는 C++(선택 사항)를 사용하여 유니버설 Windows 플랫폼용 애플리케이션을 만듭니다.
- ☐ .NET을 사용한 모바일 개발
Xamarin을 사용하여 iOS, Android 또는 Windows용 플랫폼 간 애플리케이션을 빌드합니다.

설치 세부 정보

> Visual Studio 핵심 편집기

✓ C++를 사용한 데스크톱 개발 포함됨

- ✓ C++ 핵심 데스크톱 기능
- ✓ IntelliCode

옵션

- ☒ MSVC v142 - VS 2019 C++ x64/x86 빌드 도구(v...
- ☒ Windows 10 SDK(10.0.18362.0)
- ☒ Just-In-Time 디버거
- ☒ C++ 프로파일링 도구
- ☒ Windows용 C++ CMake 도구
- ☒ 최신 v142 빌드 도구용 C++ ATL(x86 및 x64)
- ☒ Test Adapter for Boost.Test
- ☒ Test Adapter for Google Test
- ☒ Live Share
- ☒ C++ AddressSanitizer(실험적)
- ☐ 최신 v142 빌드 도구용 C++ MFC(x86 및 x64)
- ☐ v142 빌드 도구용 C++/CLI 지원(14.24)
- ☐ v142 빌드 도구용 C++ 모듈(x64/x86 - 실험적)
- ☐ Windows용 C++ Clang 도구(9.0.0 - x64/x86)

위치
D:\Program Files (x86)\Microsoft Visual Studio\2019\Community 변경...

계속하면 선택한 Visual Studio 버전에 대한 [라이선스](#)에 동의하는 것입니다. Microsoft는 Visual Studio와 함께 다른 소프트웨어를 다운로드할 수 있는 기능도 제공합니다. 이 소프트웨어는 [타사 고지 사항](#) 또는 해당 라이선스에 명시된 대로 별도로 사용이 허가됩니다. 계속하면 이러한 라이선스에도 동의하는 것입니다.

시스템 드라이브(C) 2.32GB
다른 드라이브 6.56GB
필요한 총 공간 8.88GB

다운로드하는 동안 설치 설치

1. 체크

2. 클릭




비주얼 스튜디오 설치

Visual Studio Installer

설치됨

사용 가능

 Visual Studio Community 2019

다운로드 및 확인 중: 25MB/1.81GB
1% (0KB/초)

설치 중: 패키지 9/336
0%

Microsoft.VisualStudio.NativeImageSupport

☒ 설치 후 시작

[릴리스 정보](#)

일시 중지

개발자 뉴스

[Every Xamarin.Forms Layout is a Repeater Control](#)

When you need to display a lot of data...

2020년 3월 13일 금요일

[How to write a Roslyn Analyzer](#)

Roslyn analyzers inspect your code for style,...

2020년 3월 13일 금요일

[Binding iOS Swift Libraries \(Xamarin\)](#)

The iOS platform, along with its native languages...

2020년 3월 13일 금요일

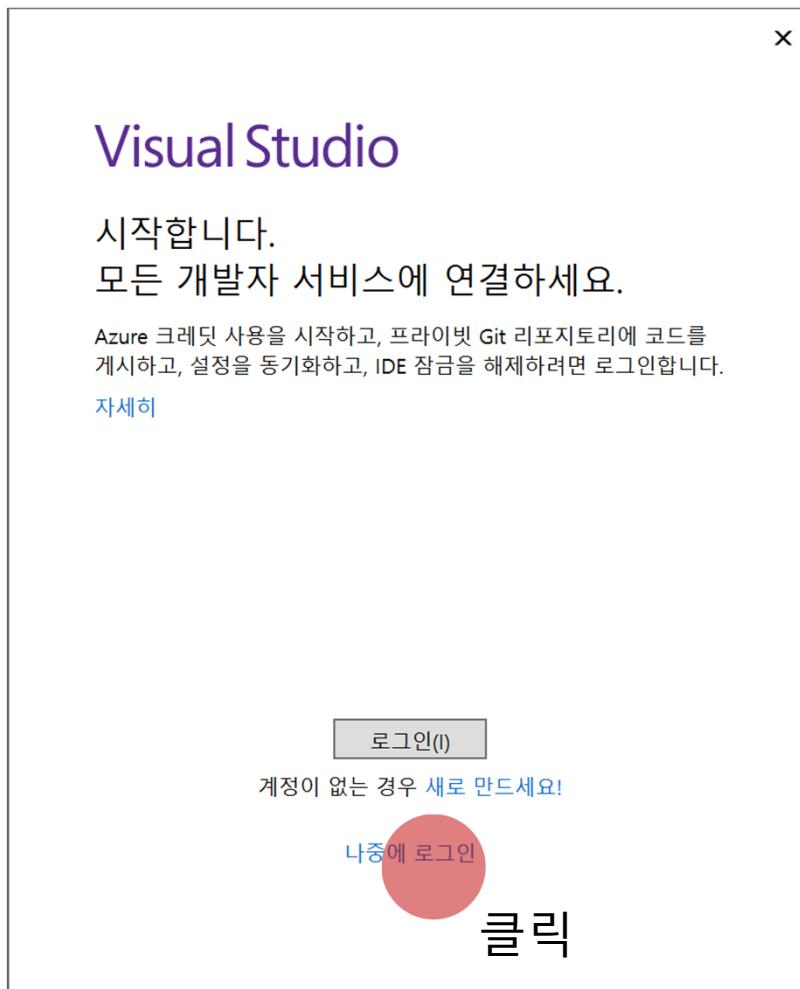
[온라인에서 뉴스 더 보기...](#)

도움이 필요하세요? [Microsoft Developer Community](#)를 확인하거나 [Visual Studio 지원](#)을 통해 문의하세요.

설치 관리자 버전 2.4.1083.303



비주얼 스튜디오 시작



나중에 로그인 클릭 후 다음 화면에서
비주얼 스튜디오 시작 버튼 클릭



비주얼 스튜디오 시작

Visual Studio 2019

최근 파일 열기(R)

Visual Studio를 사용할 때 여는 프로젝트, 폴더 또는 파일은 빠른 액세스를 위해 여기에 표시됩니다.

항상 목록의 맨 위에 표시되도록 자주 여는 항목을 고정할 수 있습니다.

시작



코드 복제 또는 체크 아웃(C)

GitHub 또는 Azure DevOps 같은 온라인 리포지토리에서 코드 가져오기



프로젝트 또는 솔루션 열기(P)

로컬 Visual Studio 프로젝트 또는 .sln 파일 열기



로컬 폴더 열기(F)

폴더 내에서 탐색 및 코드 편집



새 프로젝트 만들기(N)

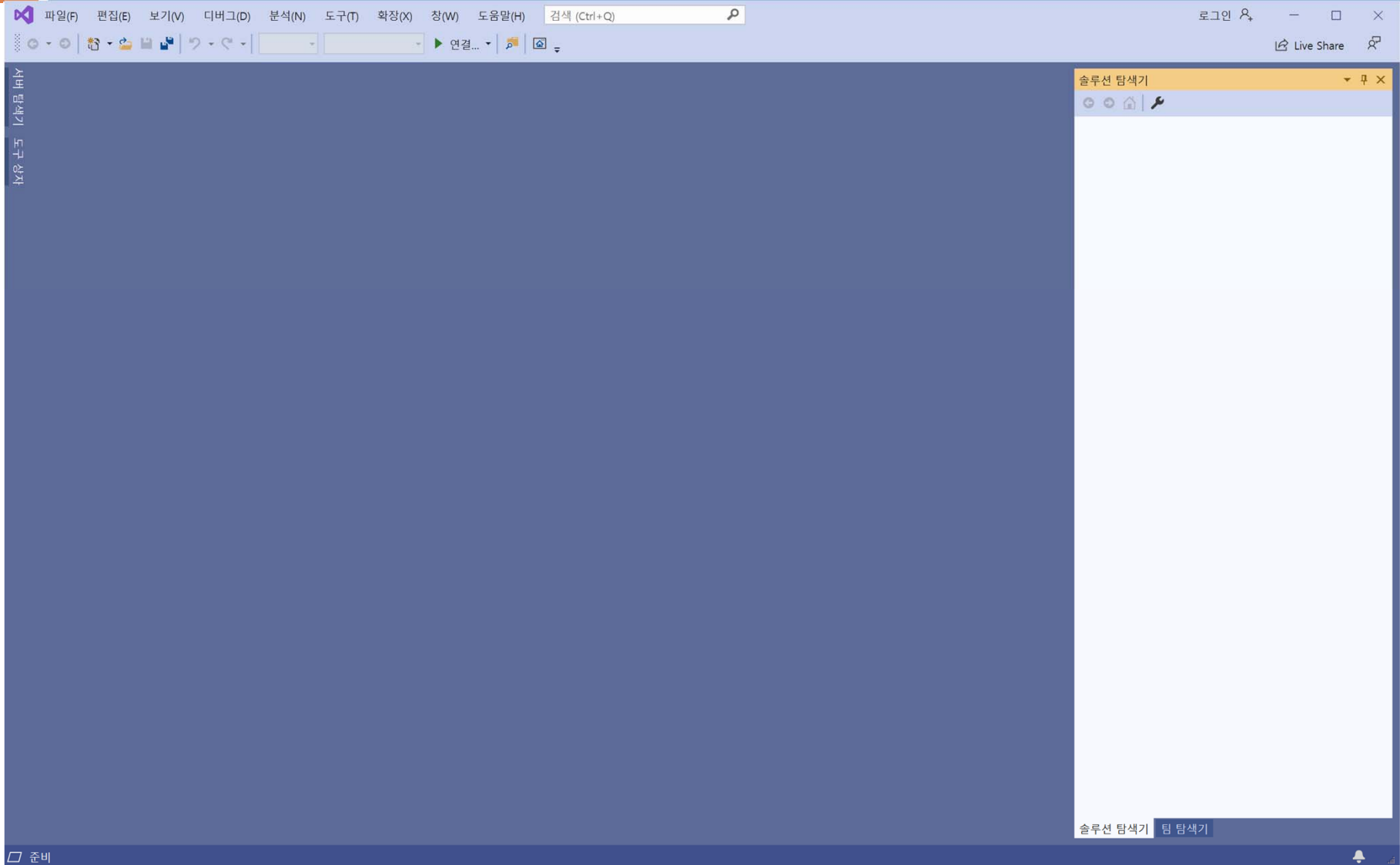
시작하려면 코드 스캐폴딩과 함께 프로젝트 템플릿을 선택하세요.

[코드를 사용하지 않고 계속\(W\) →](#)

클릭

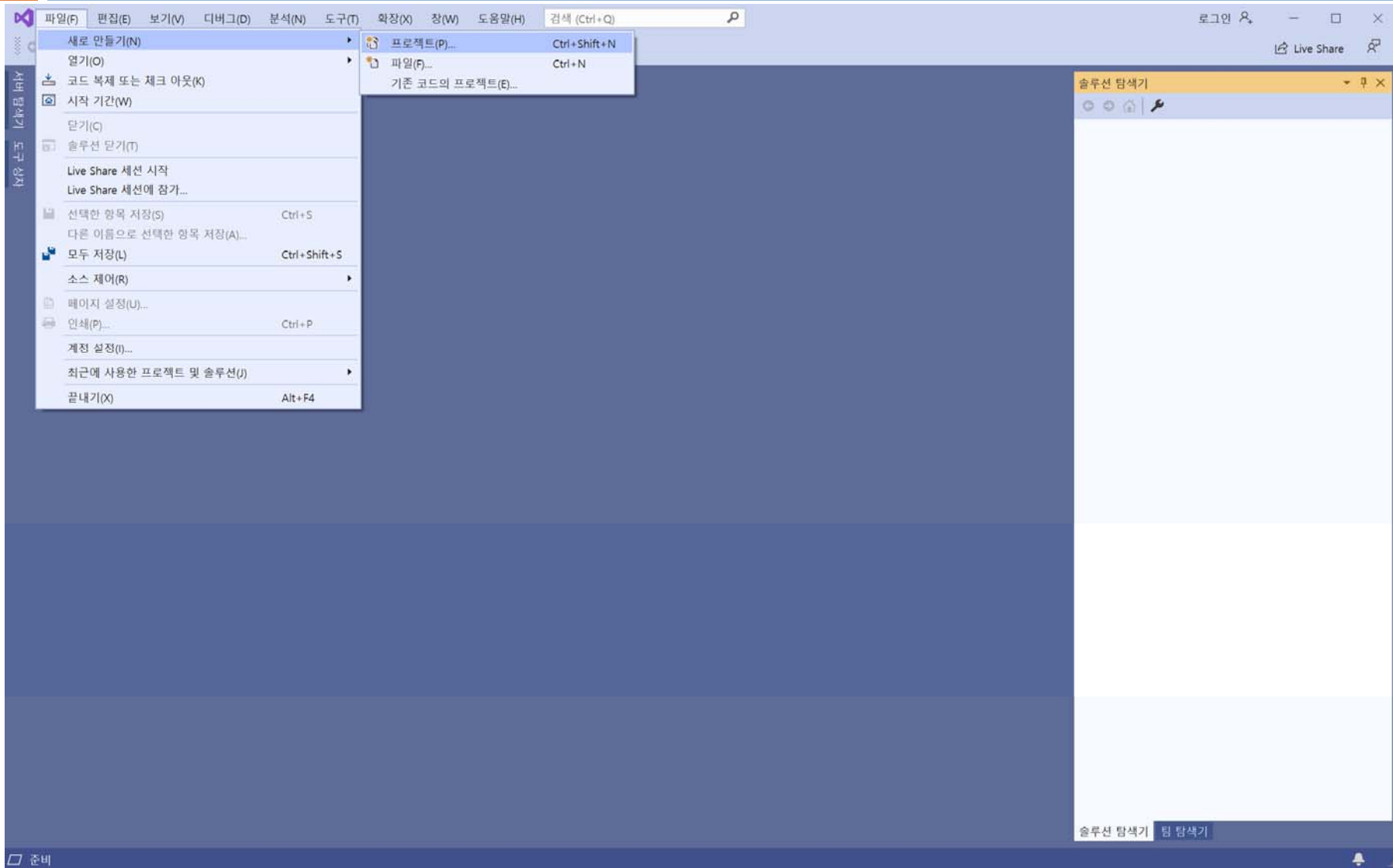


비주얼 스튜디오 시작





프로젝트 생성하기





프로젝트 생성하기

새 프로젝트 만들기

최근 프로젝트 템플릿(R)

최근에 액세스한 템플릿 목록이 여기에 표시됩니다.

템플릿 검색(Alt+S)(S)

모든 언어(L) | 모든 플랫폼(P) | 모든 프로젝트 형식(T)

- 빈 프로젝트**
Windows용 C++를 사용하여 처음부터 시작합니다. 시작 파일을 제공하지 않습니다.
C++ | Windows | 콘솔
- 콘솔 앱**
Windows 터미널에서 코드를 실행합니다. 기본적으로 "Hello World"를 출력합니다.
C++ | Windows | 콘솔
- Windows 데스크톱 마법사**
마법사를 사용하여 고유한 Windows 앱을 만드세요.
C++ | Windows | 데스크톱 | 콘솔 | 라이브러리
- Windows 데스크톱 애플리케이션**
Windows에서 실행되는 그래픽 사용자 인터페이스를 사용하는 애플리케이션용 프로젝트입니다.
C++ | Windows | 데스크톱
- 공유 항목 프로젝트**
공유 항목 프로젝트는 여러 프로젝트 간에 파일을 공유하는 데 사용됩니다.
C++ | Windows | Android | iOS | Linux | 데스크톱 | 콘솔

다음(N)

클릭

솔루션 탐색기 | 도구 상자

준비



프로젝트 생성하기

파일(F) 편집(E) 보기(V) 디버그(D) 분석(N) 도구(T) 확장(X) 창(W) 도움말(H) 검색 (Ctrl+Q)

로그인 Live Share

솔루션 탐색기 도구 상자

새 프로젝트 구성

Windows 데스크톱 마법사 C++ Windows 데스크톱 콘솔 라이브러리

프로젝트 이름(N)

FirstProgram **입력**

위치(L)

C:\Users\swkang\source\repos ...

솔루션 이름(M) ⓘ

FirstProgram

☐ 솔루션 및 프로젝트를 같은 디렉터리에 배치(D)

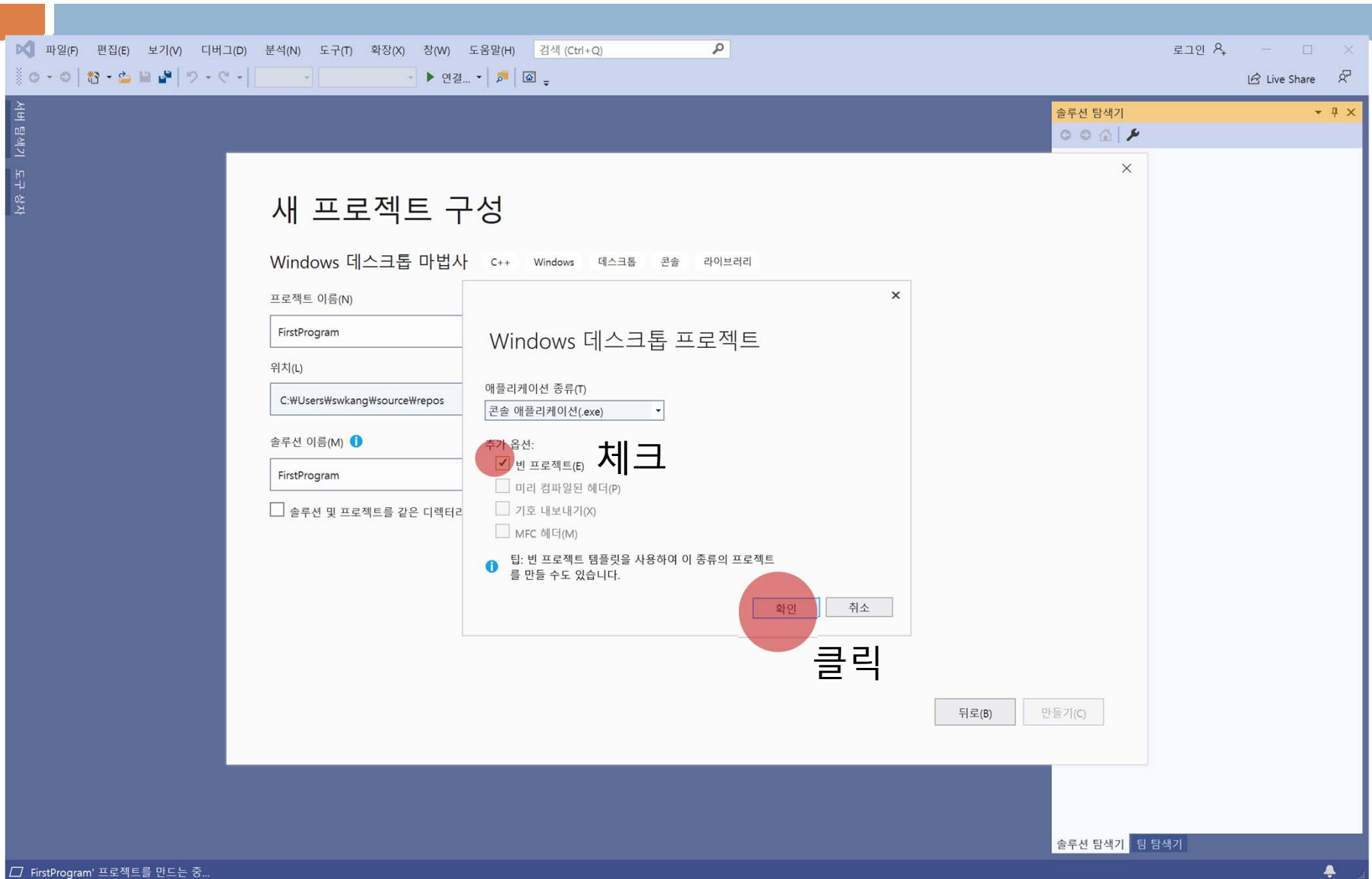
뒤로(B) **만들기(C)** **클릭**

솔루션 탐색기 팀 탐색기

준비

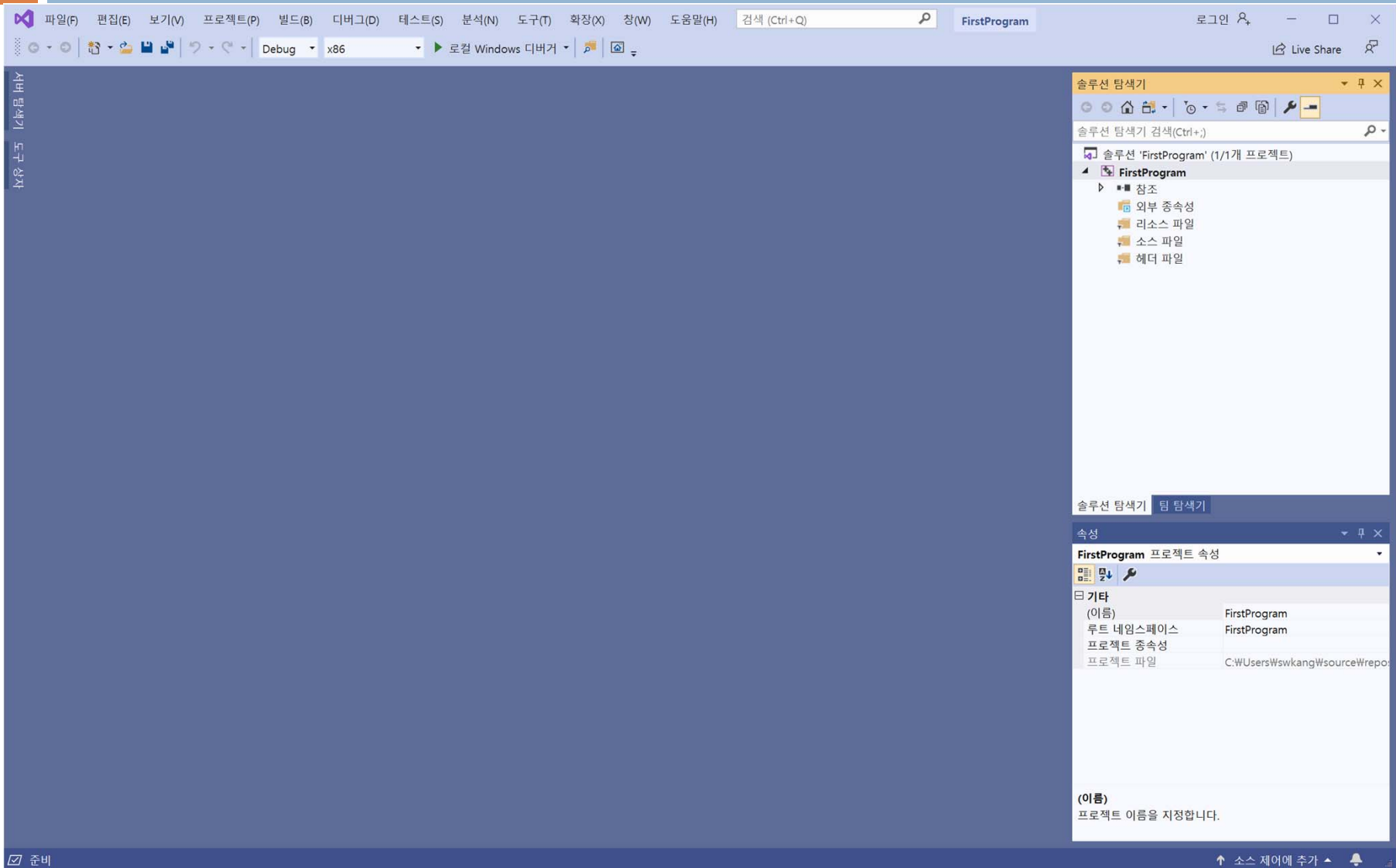


프로젝트 생성하기





프로젝트 생성하기





소스 파일 생성하기

Visual Studio IDE showing the 'FirstProgram' project. The 'Solution Explorer' on the right shows the project structure with '소스 파일' (Source Files) highlighted. A red circle and the text '우클릭' (Right-click) indicate where to click to open the context menu. The context menu is open, showing options like 'Add (A)...', 'Class (C)...', 'New Solution Explorer View (N)', etc. The 'Properties' window at the bottom right shows the '소스 파일 필터 속성' (Source File Filter Properties) for the selected file.

소스 파일 필터 속성	
고급	
SCC 파일	True
일반	
(이름)	소스 파일
고유 식별자	{4FC737F1-C7A5-4376-A066-2A...}
필터	cpp;c;cc;cxx;def;idl;hpj;bat;asr...

(이름)
필터 이름을 지정합니다.



소스 파일 생성하기

The screenshot shows the Visual Studio IDE interface. The main window displays a dark blue background. On the right side, the 'Solution Explorer' (솔루션 탐색기) is open, showing a project named 'FirstProgram'. A right-click context menu is displayed over the 'FirstProgram' project. The menu includes options like 'Add' (추가), 'Class' (클래스), 'File' (파일), 'Folder' (폴더), 'Item' (항목), 'New' (새), 'Open' (열기), 'Remove' (제거), 'Rename' (이름 바꾸기), and 'Settings' (속성). A red circle highlights the 'Add' option, and a red arrow points to it with the Korean word '클릭' (Click). Below the 'Add' option, a sub-menu is visible, showing 'New Class' (새 클래스), 'New File' (새 파일), 'New Folder' (새 폴더), 'New Item' (새 항목), and 'New Resource' (새 리소스). The 'New File' option is highlighted. At the bottom of the screen, the 'Properties' (속성) window is open, showing the 'Source File Filter' (소스 파일 필터) property set to 'True'. The 'General' (일반) tab is selected, and the 'Filter' (필터) section shows the 'Filter Name' (필터 이름) set to '소스 파일'.

클릭



소스 파일 생성하기

새 항목 추가 - FirstProgram

정렬 기준: 기본값

검색(Ctrl+E)

Visual C++

- C++ 파일(.cpp)
- 헤더 파일(.h)
- C++ 클래스

형식: Visual C++

C++ 소스 코드를 포함하는 파일을 만듭니다.

이름(N): hello.cpp

위치(L): C:\Users\Wswkang\source\repos\FirstProgram\FirstProgram\

찾아보기(B)...

추가(A) 취소

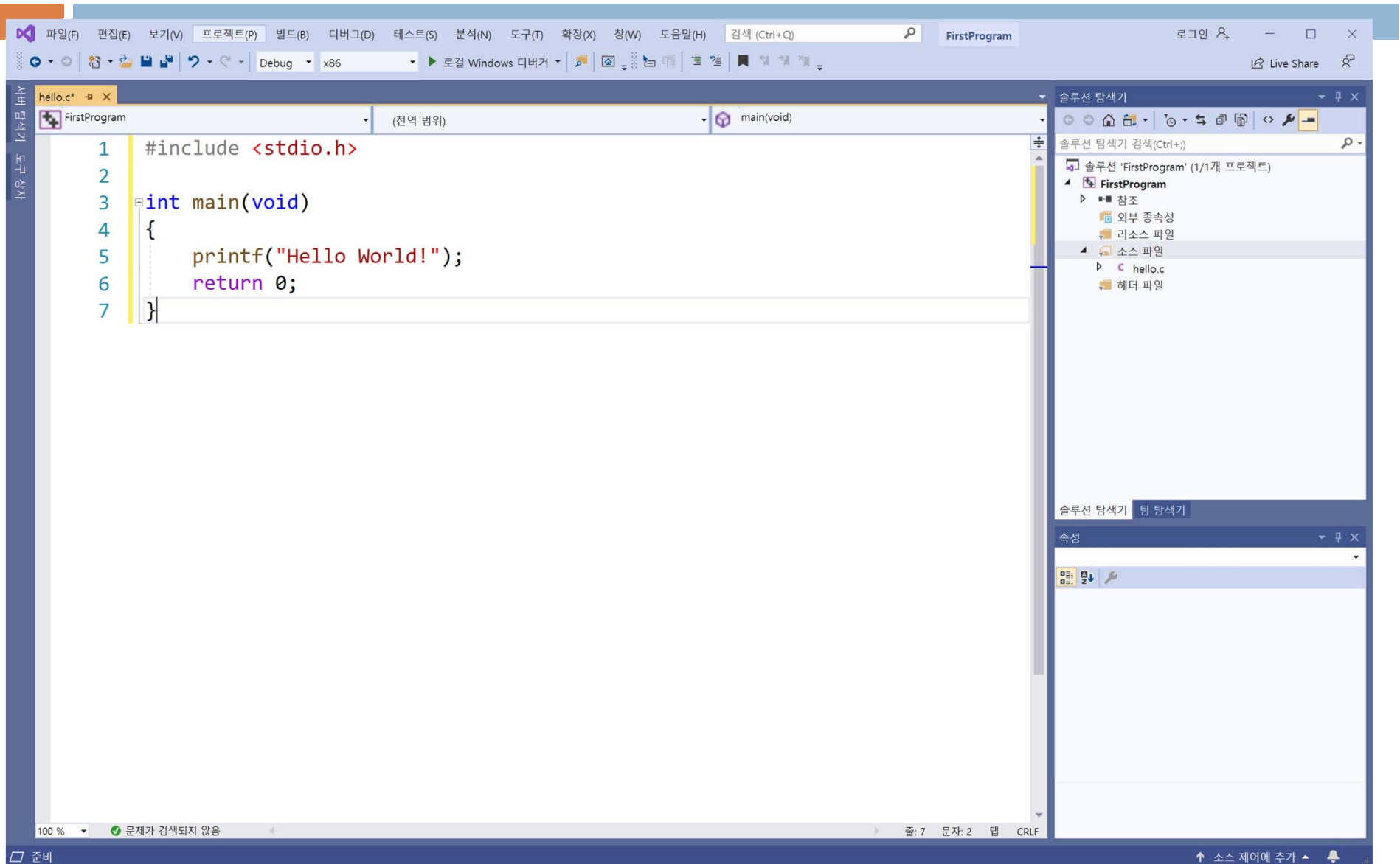
클릭

파일 이름 입력

클릭



프로그램 코드 입력





기호는 정확하게 입력

문장의 끝에는 세미콜론



컴파일하기

클릭

Build (B) | Debug (D) | Tests (S) | Analysis (N) | Tools (T) | Extensions (X) | Windows (W) | Help (H) | Search (Ctrl+Q) | FirstProgram | 로그인 | Live Share

솔루션 빌드(B) Ctrl+Shift+B
솔루션 다시 빌드(R)
솔루션 정리(C)
솔루션의 전체 프로그램 데이터베이스 파일 빌드
솔루션에서 코드 분석 실행(Y) Alt+F11
FirstProgram 빌드(U) Ctrl+B
FirstProgram 다시 빌드(E)
FirstProgram 정리(N)
FirstProgram에서 코드 분석 실행(A)
프로젝트만(J)
알림 빌드(T) ...
구성 관리자(O) ...
컴파일(M) Ctrl+F7
파일에서 코드 분석 실행(F) Ctrl+Shift+Alt+F7

```
1 #include <stdio.h>
2
3 int main(void)
4 {
5     printf("Hello, World!\n");
6     return 0;
7 }
```

출력

출력 보기 선택(S): 빌드

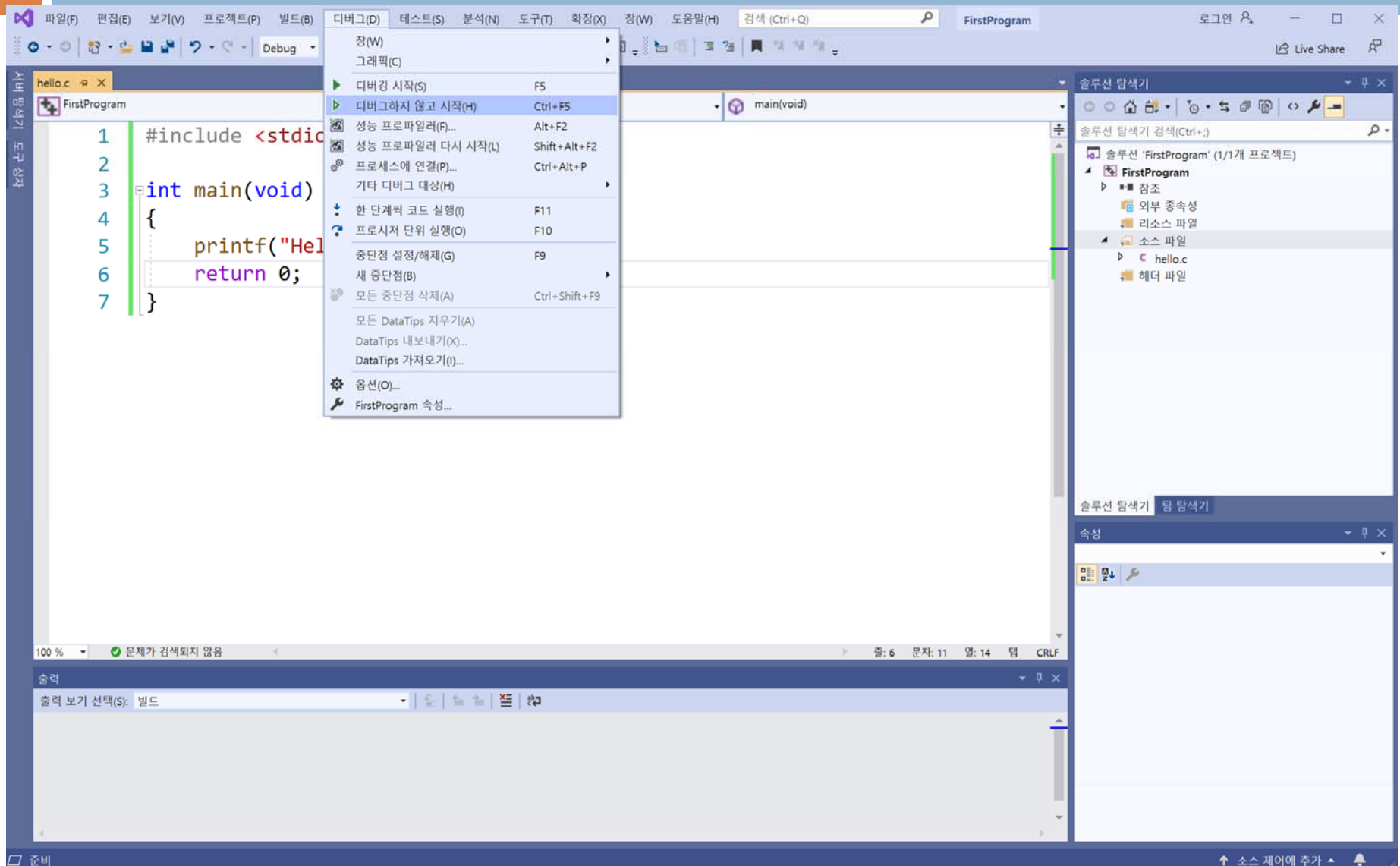
```
1>----- 빌드 시작: 프로젝트: FirstProgram, 구성: Debug Win32 -----
1>hello.c
1>FirstProgram.vcxproj -> C:\Users\sewkang\source\repos\FirstProgram\Debug\FirstProgram.exe
===== 빌드: 성공 1, 실패 0, 최신 0, 생략 0 =====
```

빌드에 성공했습니다.

소스 제어에 추가

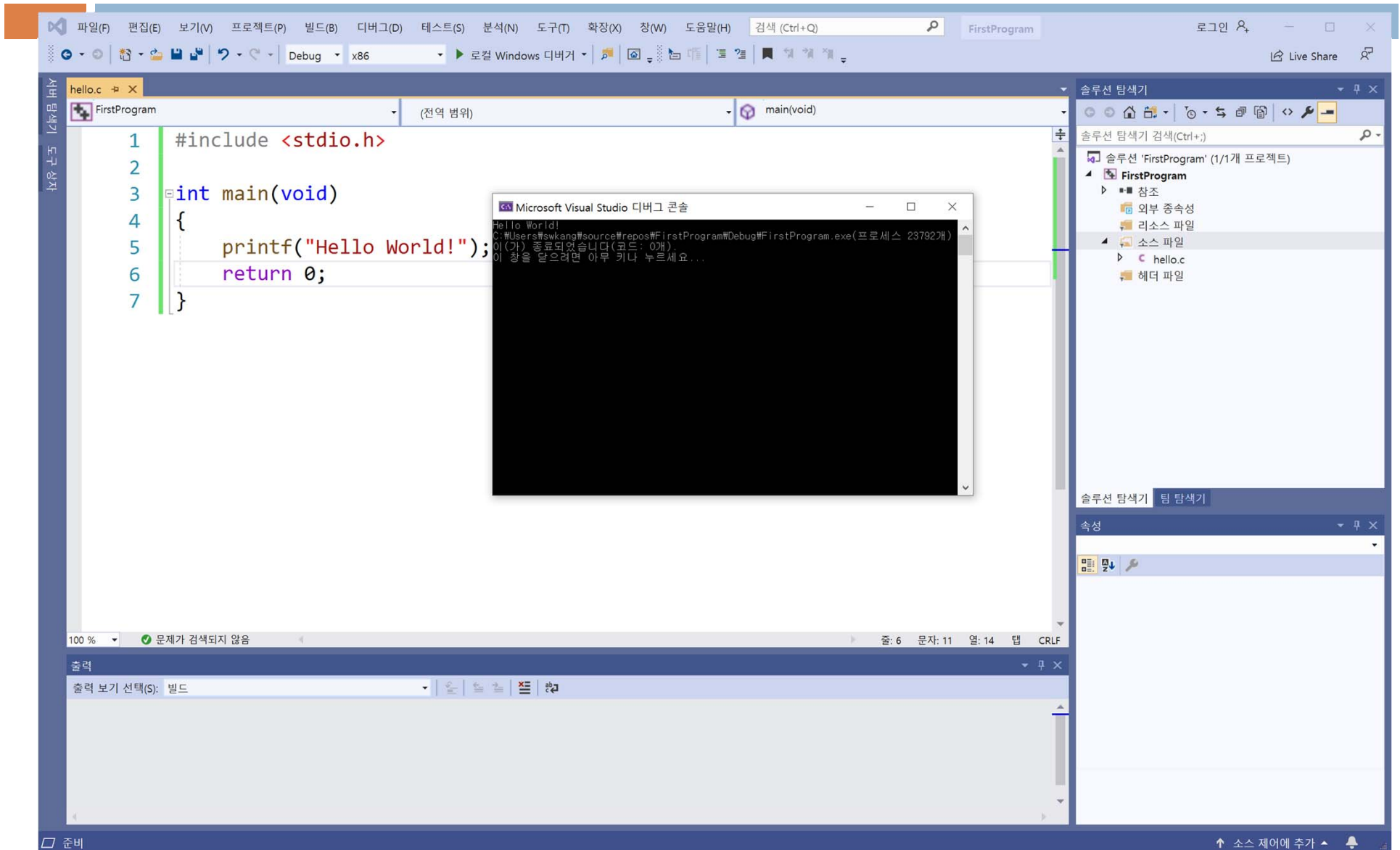


프로그램 실행 하기





프로그램 실행 하기





중간 점검



- 에디터, 컴파일러, 링커, 실행, 디버깅 등의 기능이 하나의 프로그램 안에 들어 있는 것을 무엇이라고 하는가?
- 비주얼 스튜디오에서 새로운 프로젝트를 생성하는 메뉴는 무엇인가?
- 비주얼 스튜디오에서 프로젝트에 속하는 소스 파일을 컴파일하여 실행 파일을 생성하는 메뉴는 무엇인가?
- C언어에서는 대문자와 소문자를 구별하는가?
- 비주얼 스튜디오를 이용하여서 sample.c라는 소스 파일을 컴파일하였을 때 생성되는 파일들은 무엇인가?
- 비주얼 스튜디오를 사용하여 소스 프로그램을 편집하는 경우, 메모장 같은 다른 텍스트 에디터를 사용하여도 되는가?



첫번째 프로그램의 설명

```
#include <stdio.h>
```

```
int main(void)
```

```
{
```

```
    printf("Hello World!");
```

```
    return 0;
```

```
}
```





프로그램 == 작업 지시서

*화면에 "Hello World!"를
표시한다.

작업 지시서

```
#include <stdio.h>
```

```
int main(void)
```

```
{
```

```
    printf("Hello World!");
```

```
    return 0;
```

```
}
```

프로그램



작업을 적어주는 위치

```
#include <stdio.h>
```

```
int main(void)
```

```
{
```

```
    return 0;
```

```
}
```

여기다 원하는 작업을
수행하는 문장을 적어준다.

프로그램



간략한 소스 설명

```
#include <stdio.h>
```

헤더파일을 포함한다.

```
int main(void)
```

메인 함수 시작

```
{
```

```
    printf("Hello World!");
```

화면에 "Hello World!"를 출력

```
    return 0;
```

외부로 0값을 반환

```
}
```

메인 함수 종료

프로그램



헤더 파일 포함

- `#include`는 소스 코드 안에 특정 파일을 현재의 위치에 포함

- 주의!: 전처리기 지시자 문장 끝에는 세미콜론(;)을 붙이면 안 된다.

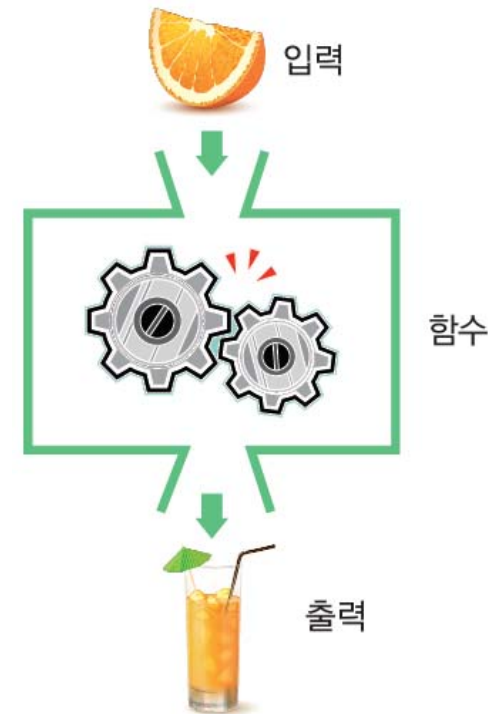
`#include <stdio.h>`

- 헤더 파일(header file): 컴파일러가 필요로 하는 정보를 가지고 있는 파일
- stdio.h: standard input output header file



함수

- 함수(function): 특정한 작업을 수행하기 위하여 작성된 독립적인 코드
- (참고) 수학적 함수 $y = x^2 + 1$
- 프로그램 = 함수의 집합





main() 함수

- 함수의 반환값

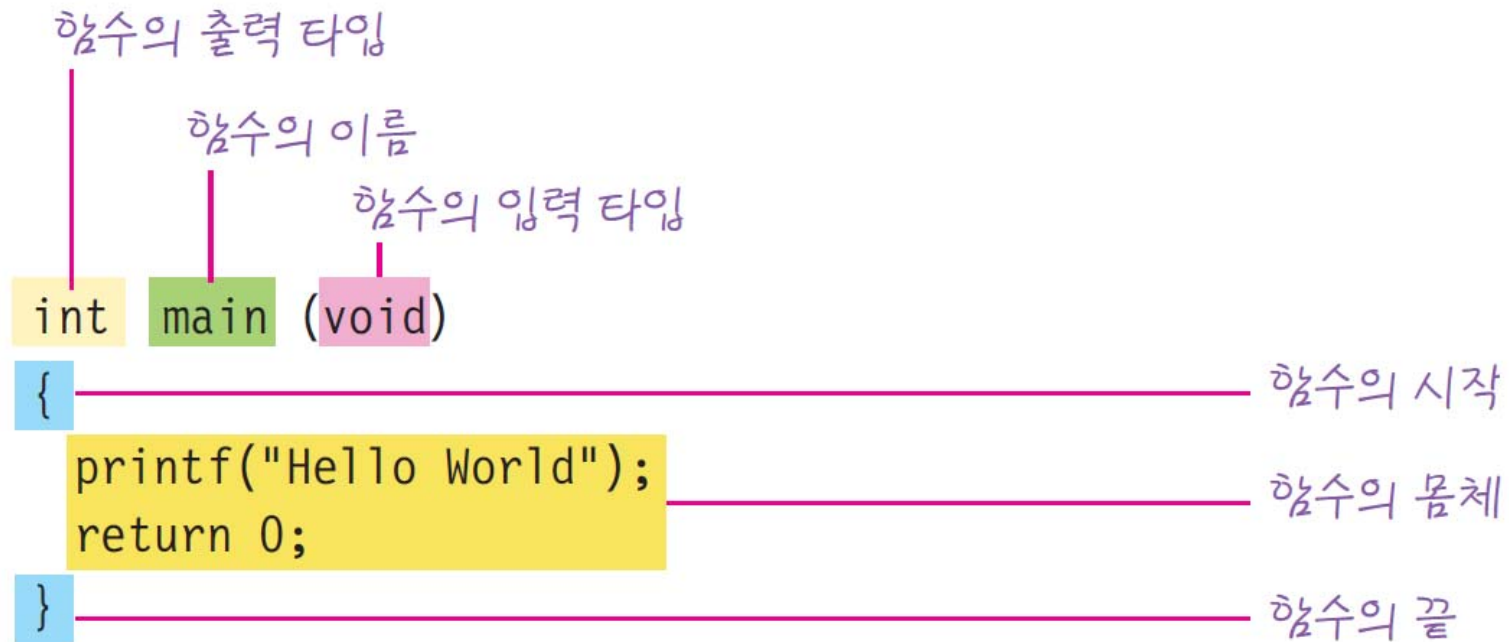
- 함수의 입력은 없음!

int main(void)

- 함수의 이름
- main()은 가장 먼저 수행되는 함수



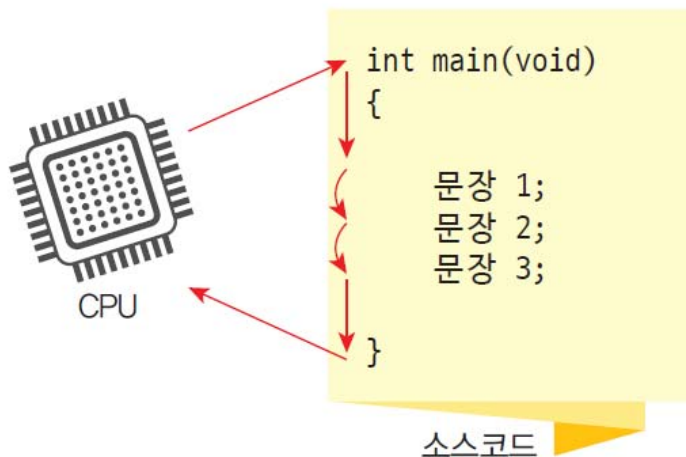
함수의 간략한 설명





문장 (statement)

- 함수는 여러 개의 문장으로 이루어진다.
- 문장들은 순차적으로 실행된다.
- 문장의 끝에는 반드시 ;이 있어야 한다.



소스 코드의 문장들은
기본적으로 차례대로
실행됩니다.



printf() 호출

- printf()는 컴파일러가 제공하는 함수로서 출력을 담당한다

printf("Hello World!");

- 큰따옴표 안의 문자열이 화면에 출력된다.





함수의 반환값

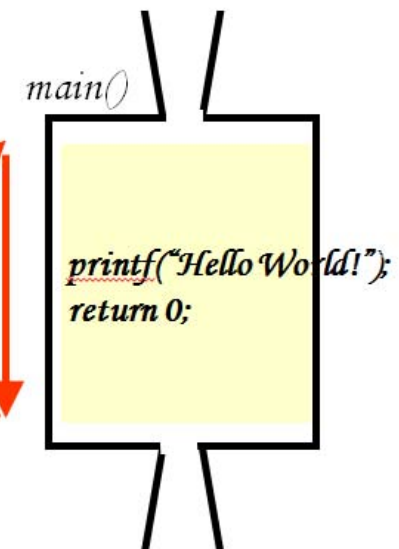
- return은 함수의 결과값을 외부로 반환

return 0;

- 반환값은 0

운영 체제

운영 체제





중간 점검

- 문장의 끝에 추가하여야 하는 기호는?
- C프로그램에 반드시 있어야 하는 함수는?
- printf()가 하는 기능은 무엇인가?

