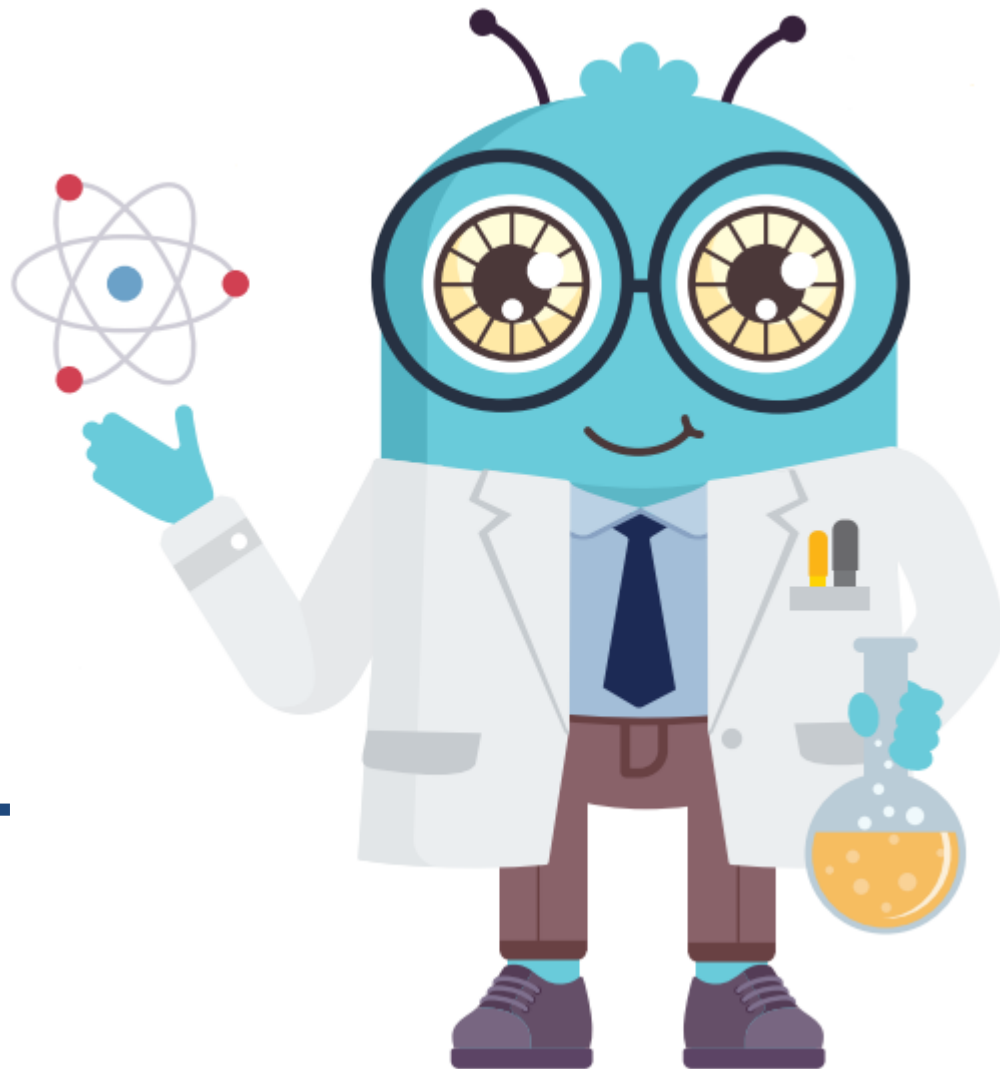


*Chapter 01*

# C 언어의 이해와 컴파일러 설치



# 목차

1. C 언어의 이해
2. C 프로그램 작성 방법
3. C 컴파일러 설치와 간단한 실습

01

C 언어의 이해

# 1. C 언어의 이해

## 1. 프로그래밍 개요

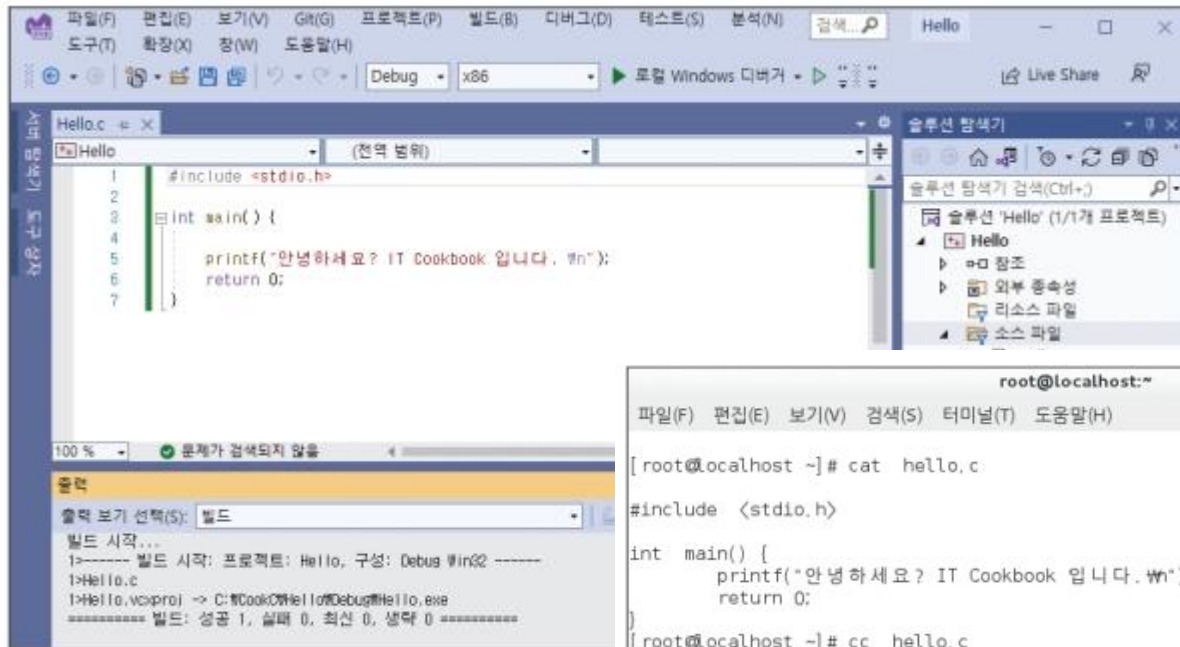
- 한글, 엑셀, 게임, 모바일 앱 등의 프로그램을 만들 때 사용하는 언어
- ‘컴퓨터 프로그래머’라는 사람들이 제작
  - 간단한 프로그램(계산기, 메모장)은 혼자서 만들기도 하지만 다양한 기능의 복잡한 프로그램
- C 언어(C++ 포함)를 가장 많이 사용함
  - 모바일용 프로그래밍을 작성할 때에는 주로 자바, C/C++, C#을 많이 사용
  - PC 운영체제나 PC 운영체제에서 작동되는 응용 프로그램을 작성할 경우에는 C나 C++ 등을 주로 사용

# 1. C 언어의 이해

## 2. C 의 특징

- **이식성이 좋다**

- 한번 만들어 놓으면 윈도우 운영체제 외에 유닉스나 기타 운영체제에서 사용 가능



윈도우 계열



유닉스 계열

그림 1-1 윈도우 계열과 유닉스 계열의 코딩과 컴파일 비교

# 1. C 언어의 이해

## 2. C 의 특징

- 하드웨어와 관련된 부분까지 접근할 수 있다
  - 시스템 프로그래밍이 가능



그림 1-2 하드웨어(왼쪽)를 제어하는 C(오른쪽)

```
37 architecture Behavioral of Binary_Counter is
38     signal Q : Integer range 0 to 255 := 0;
39     begin
        ~ 중략 ~
44     if(CLR = '0') then
45         Q <= 0;
46         LED <= not conv_std_logic_vector(Q, 8);
47     elsif(CLK'EVENT and CLK = '1' and CE = '1')
        then
48         if(Q=256) then
49             Q <= 0;
50             LED <= not conv_std_logic_vector(Q, 8);
51         else
52             Q <= Q+1;
53             LED <= not conv_std_logic_vector(Q, 8);
54         else if;
55     end if;
```

# 1. C 언어의 이해

## 2. C 의 특징

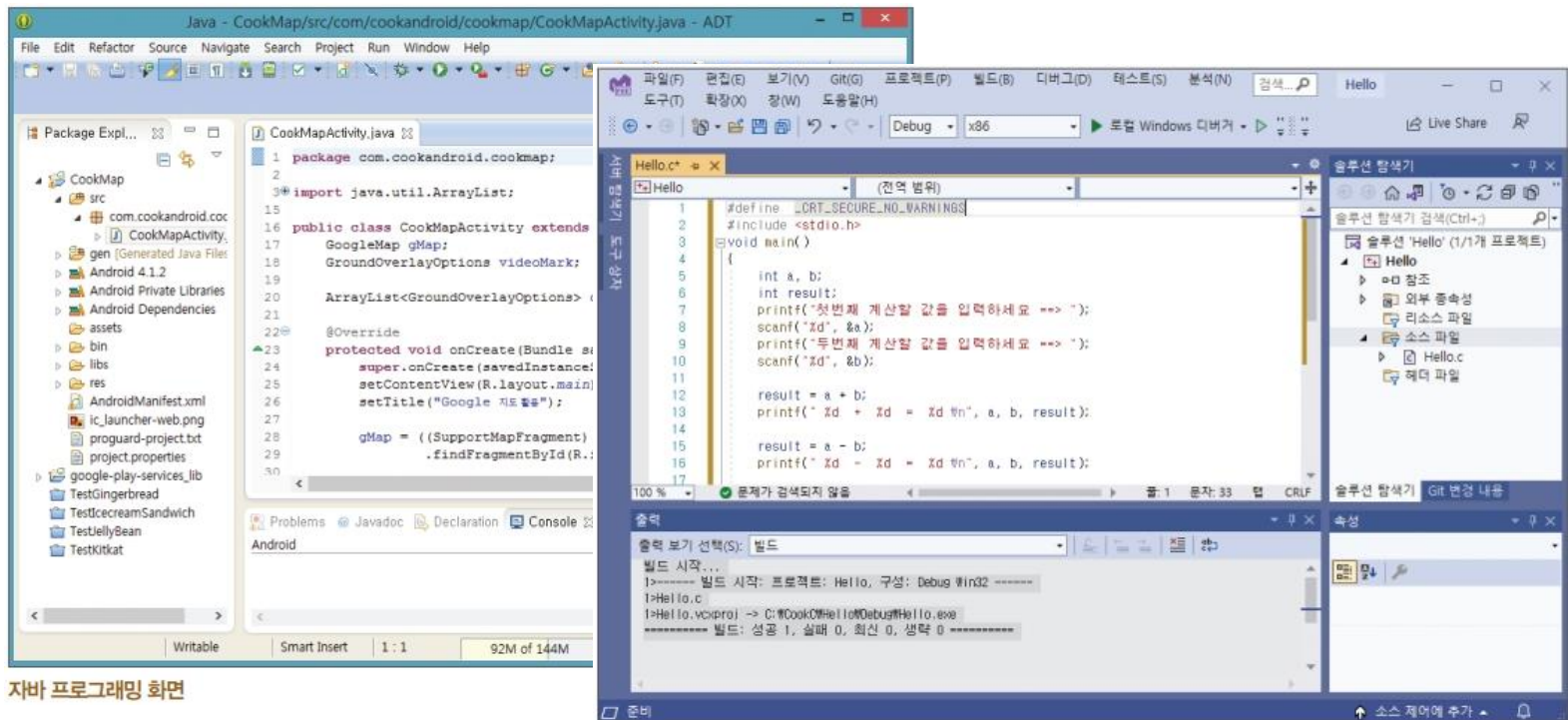
### 여기서 잠깐 C 탄생의 비밀

- C가 탄생하기 전인 1960년대 후반에는 프로그래머인 마틴 리처즈 (Martin Richards)가 개발한 BCPL이라는 프로그래밍 언어가 있었음
- 벨연구소에서 근무하던 켄 톰프슨(Ken Thompson)은 BCPL을 기반으로 B라는 프로그래밍 언어를 개발했는데 이것이 바로 C의 모태
- 1970년대 초에 벨연구소에서 근무하던 데니스 리치(Dennis Ritchie)가 유닉스 운영체제를 개발하려고 C를 개발
- 데니스는 B를 바탕으로 C에 많은 기능을 추가했는데, 이 때문에 B보다 한 단계 진보했다는 의미에서 알파벳의 B 다음인 'C'가 이름으로 붙었음
- C는 유닉스 운영체제 개발의 90% 이상을 담당하며 전 세계적으로 가장 많이 사용되는 프로그래밍 언어로 성장

# 1. C 언어의 이해

## 2. C 의 특징

- 구문이 간결하고 명확하다
- C를 익히면 다른 프로그래밍 언어도 쉽게 익힐 수 있다
  - C를 잘할 수 있다면 거의 대부분의 다른 프로그래밍 언어를 쉽게 익힐 수 있음



C 프로그래밍 화면

그림 1-3 자바와 C의 구문 비교



# 1. C 언어의 이해

---

## 3. C 컴파일러의 종류

- C 프로그래밍을 하려면 반드시 C 컴파일러가 필요함
- 종류 : Turbo C, Turbo C++, Borland C++, MS C++, Visual C++, GCC, G++ 등
- 요즘의 C 컴파일러는 C와 더불어 C++도 동시 지원

02

# C 프로그램 작성 방법

## 2. C 프로그램 작성 방법

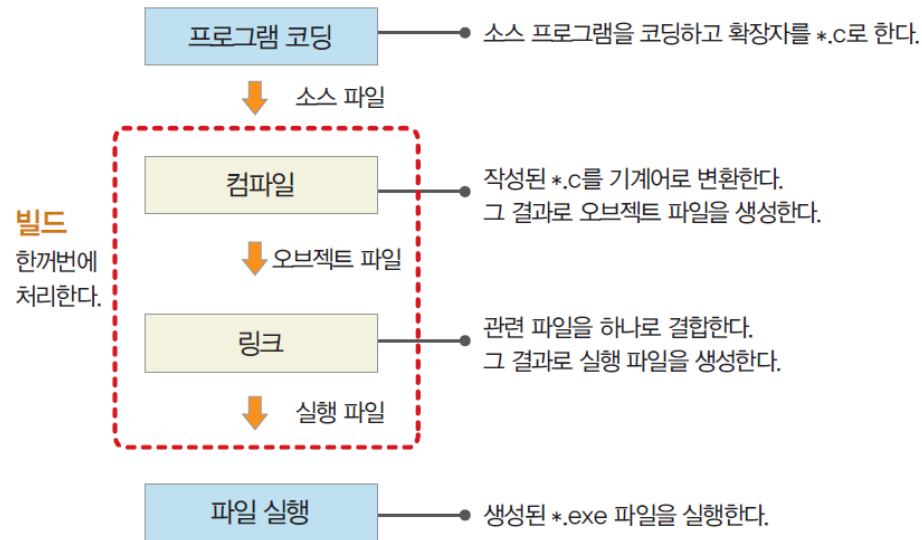


그림 1-4 C의 작성과 실행 순서

### 1. 프로그램 코딩

- '코딩(Coding)'이란 용어는 머릿속에 그려진 결과물을 만들기 위해 C 컴파일러가 알아들을 수 있는 형식으로 문서를 작성한다는 의미
- 사람만 이해하는 코딩

어이 컴퓨터~  
100에서 50을 뺀 결과가 뭐지?  
한번 모니터에 출력해봐~~

# [기본 1-1] C 프로그램 맛보기

## 1. 프로그램 코딩

기본 1-1 C 프로그램 맛보기

1-1.c

```
01 #include <stdio.h>
02
03 void main( )
04 {
05     int result;
06     result = 100 - 50;
07     printf(" %d ", result);
08 }
```

- 100에서 50을 뺀 결과를 모니터에 출력하는 전형적인 C 프로그램

### SELF STUDY

123과 456을 곱한 결과를 출력하도록 [기본 1-1]을 수정해보자.

## 2. C 프로그램 작성 방법

### 2. 컴파일과 링크

- 컴파일 : 소스 파일을 컴퓨터가 이해할 수 있는 기계어로 바꾸는 과정



그림 1-5 컴파일의 개념

## 2. C 프로그램 작성 방법

### 2. 컴파일과 링크

- 링크 : 여러 개의 '오브젝트 파일'을 하나의 '실행 파일'로 묶어주는 과정

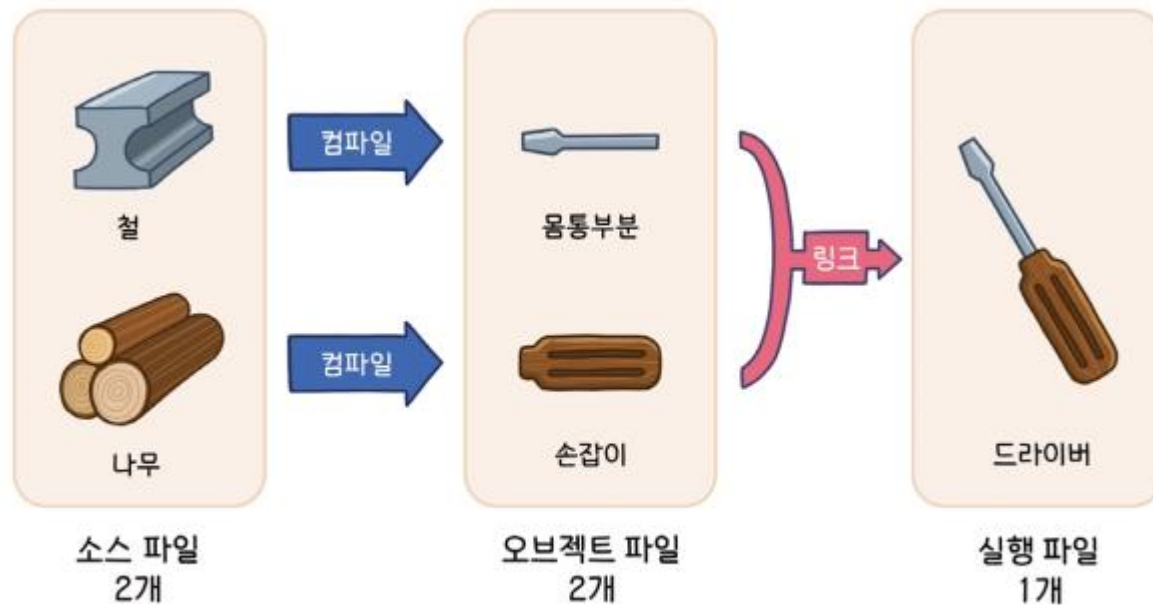
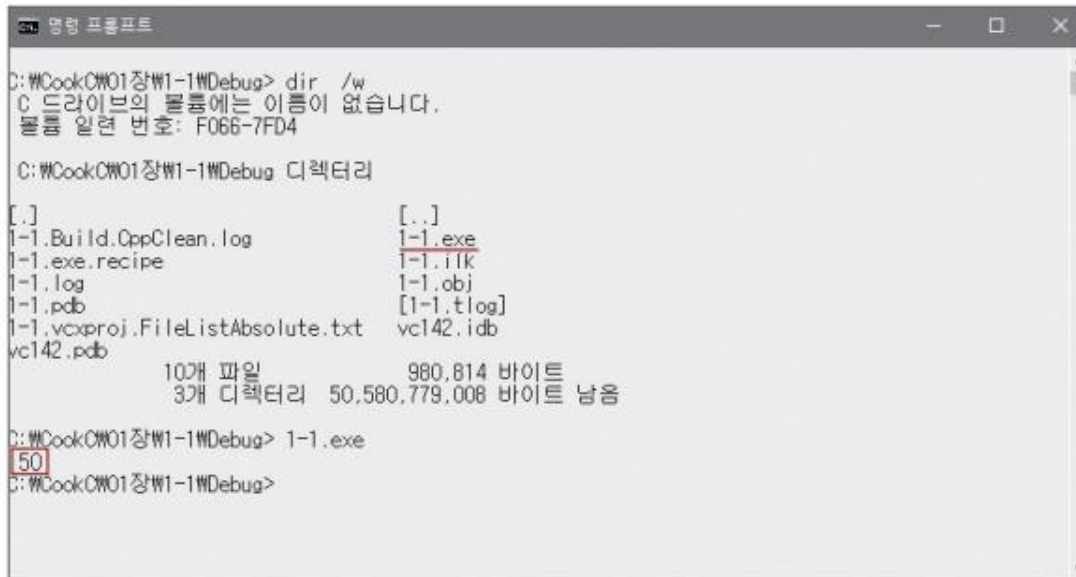


그림 1-6 컴파일과 링크의 개념

## 2. C 프로그램 작성 방법

### 3. 프로그램 실행



```
C:\CookCWO1장\1-1\Debug> dir /w
C 드라이브의 볼륨에는 이름이 없습니다.
볼륨 일련 번호: F066-7FD4

C:\CookCWO1장\1-1\Debug 디렉터리

[.]
1-1.Build.OppClean.log          1-1.exe
1-1.exe.recipe                 1-1.ilink
1-1.log                        1-1.obj
1-1.pdb                       [1-1.tlog]
1-1.vcxproj.FileListAbsolute.txt vc142.idb
vc142.pdb
                               10개 파일          980,814 바이트
                               3개 디렉터리  50,580,779,008 바이트 남음

C:\CookCWO1장\1-1\Debug> 1-1.exe
50
C:\CookCWO1장\1-1\Debug>
```

그림 1-7 컴파일과 링크가 완료된 1-1.exe 파일

#### 여기서 잠깐 빌드=컴파일+링크

- 컴파일과 링크는 소스 파일이 하나뿐이더라도 반드시 수행해야 함
- 대부분의 컴파일러는 컴파일과 링크 과정을 별도로 처리하지 않음
- 그래서 컴파일과 링크를 합쳐서 컴파일 또는 빌드(build)라고 함
- 필요에 따라 컴파일과 링크를 별도로 수행할 수도 있음

03

C 컴파일러 설치와 간단한 실습



### 3. C 컴파일러 설치와 간단한 실습

#### 1. 실습할 수 있는 Visual Studio 버전

- 이 책에서는 Visual Studio 2012 Professional 버전을 사용

표 1-1 윈도우에 설치 가능한 Visual Studio

운영체제	설치 가능한 유료 버전	설치 가능한 무료 버전
Windows 7(sp1) Windows 8 Windows 8.1	Visual Studio 2008, 2010, 2012, 2013, 2015, 2017, 2019	Visual Studio Express 2010 Visual Studio Express 2012, 2013, 2015 for Windows Desktop Visual Studio Community 2013, 2015, 2017, 2019
Windows 10 Windows 11	Visual Studio 2012, 2013, 2015, 2017, 2019, 2022	Visual Studio Express 2012, 2013, 2015 for Windows Desktop Visual Studio Community 2013, 2015, 2017, 2019, 2022

- 유료 버전의 Visual Studio는 몇 개의 에디션이 있는데 기능이 많은 순서로 Enterprise, Professional 등이 있음
- 이 책에서는 무료이며 최신버전인 Visual Studio Community 2022 버전을 사용하지만 [표 1-1]의 어떤 버전을 사용해도 동일하게 실습할 수 있음

### 3. C 컴파일러 설치와 간단한 실습

## 2. Visual Studio 2012 Professional 설치

- Visual Studio 2012 Professional 설치 방법

01 <https://visualstudio.com>에서 Visual Studio Community 설치 파일을 다운로드



그림 1-8 설치 파일 다운로드

### 3. C 컴파일러 설치와 간단한 실습

#### 02 다운로드한 파일을 실행한 뒤 <계속>을 클릭



그림 1-9 설치 초기 화면

#### 03 잠시 다운로드 및 설치가 진행

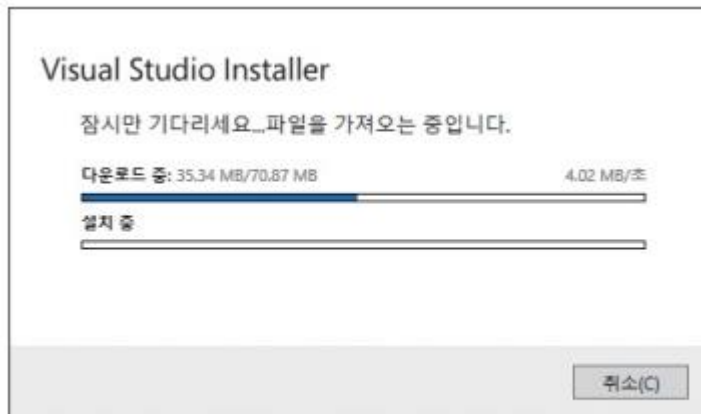


그림 1-10 설치 초기 진행

### 3. C 컴파일러 설치와 간단한 실습

**04** [워크로드]에서는 'C++를 사용한 데스크톱 개발'만 체크한 뒤 <설치>를 클릭, 프로그램을 설치하려면 10GB 정도의 하드디스크 여유 공간이 필요

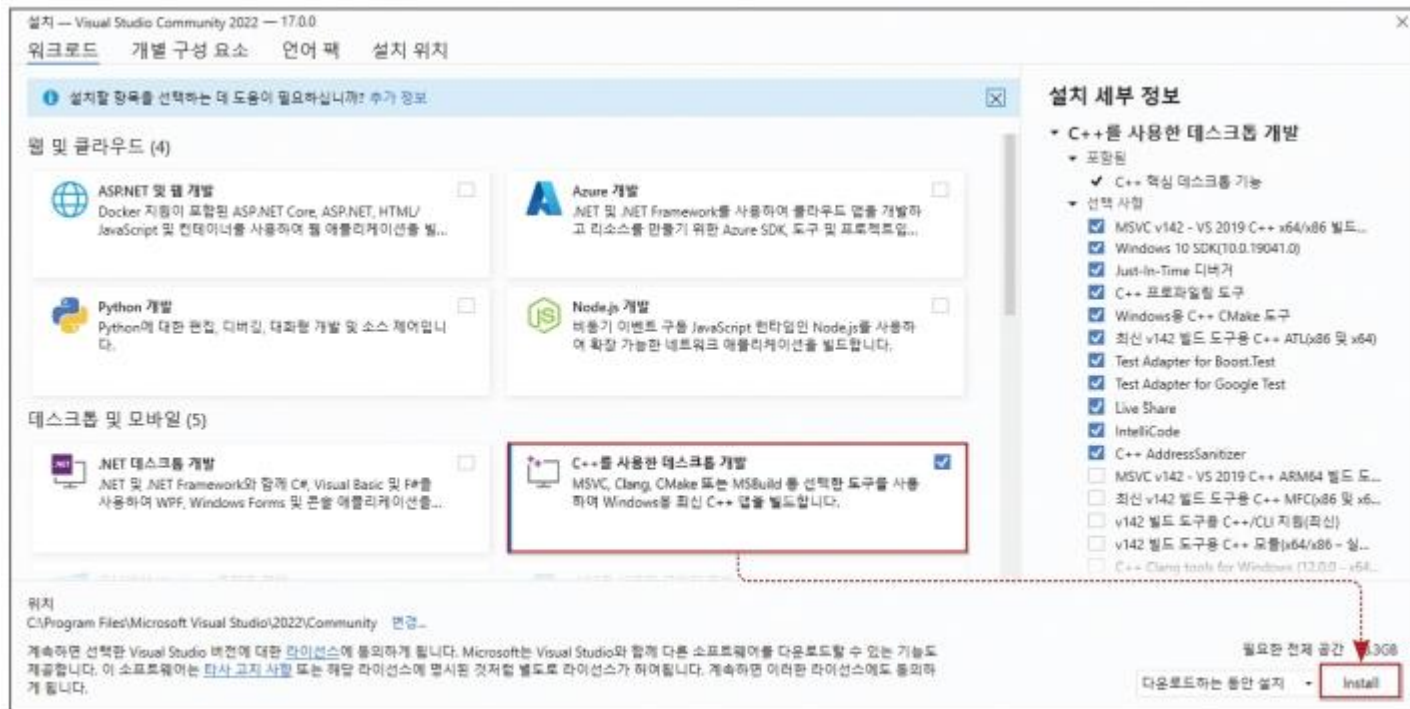


그림 1-11 설치할 기능 선택

**TIP |** 'C++을 사용한 데스크톱 개발' 항목이 보이지 않으면 화면을 아래로 스크롤

### 3. C 컴파일러 설치와 간단한 실습

**05** '설치 후 시작'이 체크되어 있으면 설치 완료 후에 Visual Studio가 자동으로 시작



그림 1-12 설치 후 자동 시작 체크

**TIP |** 설치하는 시점의 Visual Studio 세부 버전이 달라질 수 있으며 [Visual Studio Community 2022 Current] 등으로 약간 다르게 나올 수도 있음

### 3. C 컴파일러 설치와 간단한 실습

#### 06 설치가 진행(컴퓨터 성능에 따라서 수 분에서 수십 분이 걸림)

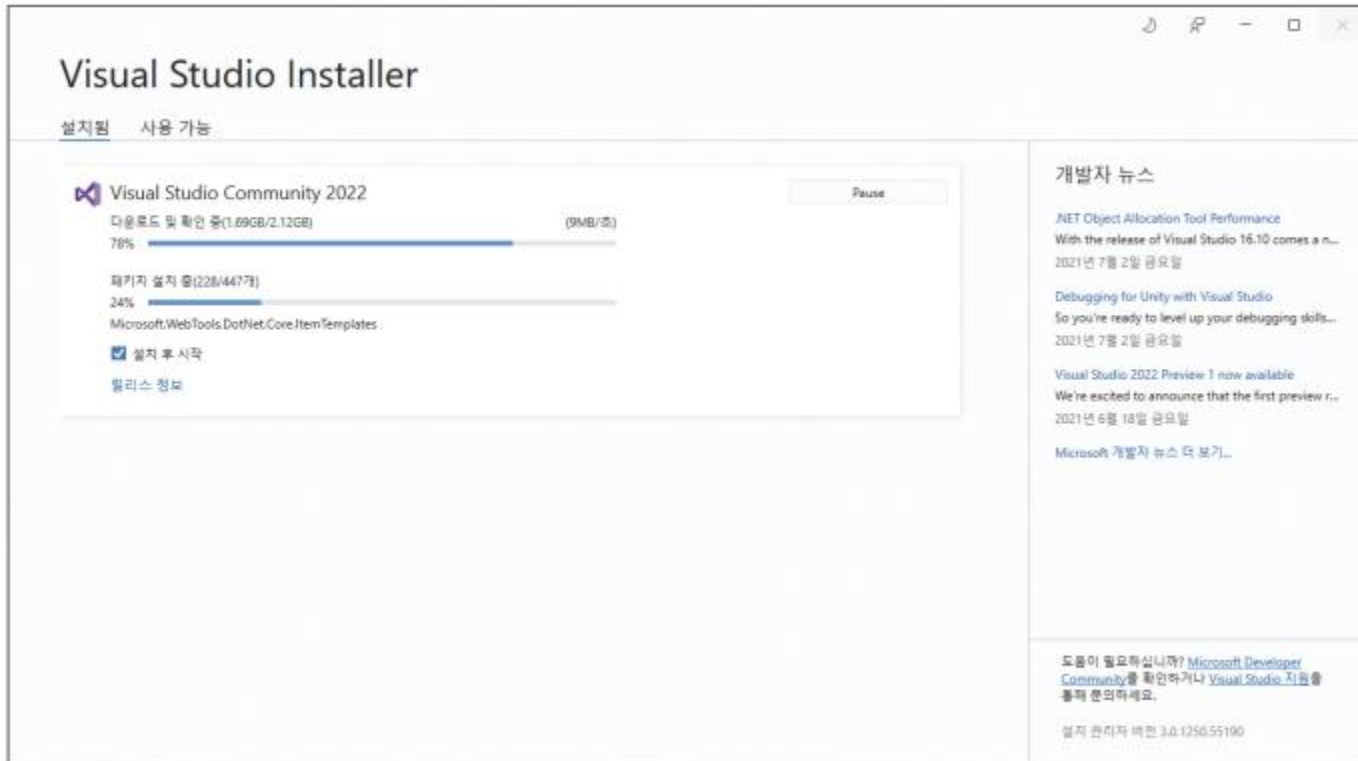


그림 1-13 설치 진행

### 3. C 컴파일러 설치와 간단한 실습

**07** 설치 완료 후, 다시 부팅하라는 메시지 창이 나오면 컴퓨터를 재부팅

**08** 윈도우의 [시작] 클릭 → <모든 앱> 클릭 → [V] 입력 → [Visual Studio 2022 Currents] 또는 [Visual Studio 2022]를 클릭해서 Visual Studio 실행



그림 1-14 Visual Studio 시작

### 3. C 컴파일러 설치와 간단한 실습

**09** [Visual Studio 로그인] 화면이 나오면 우선은 제일 아래 <나중에 로그인>을 클릭  
[친숙한 환경에서 시작] 창이 나오는데 원하는 화면 색상을 선택하고 < Visual Studio>을 클릭

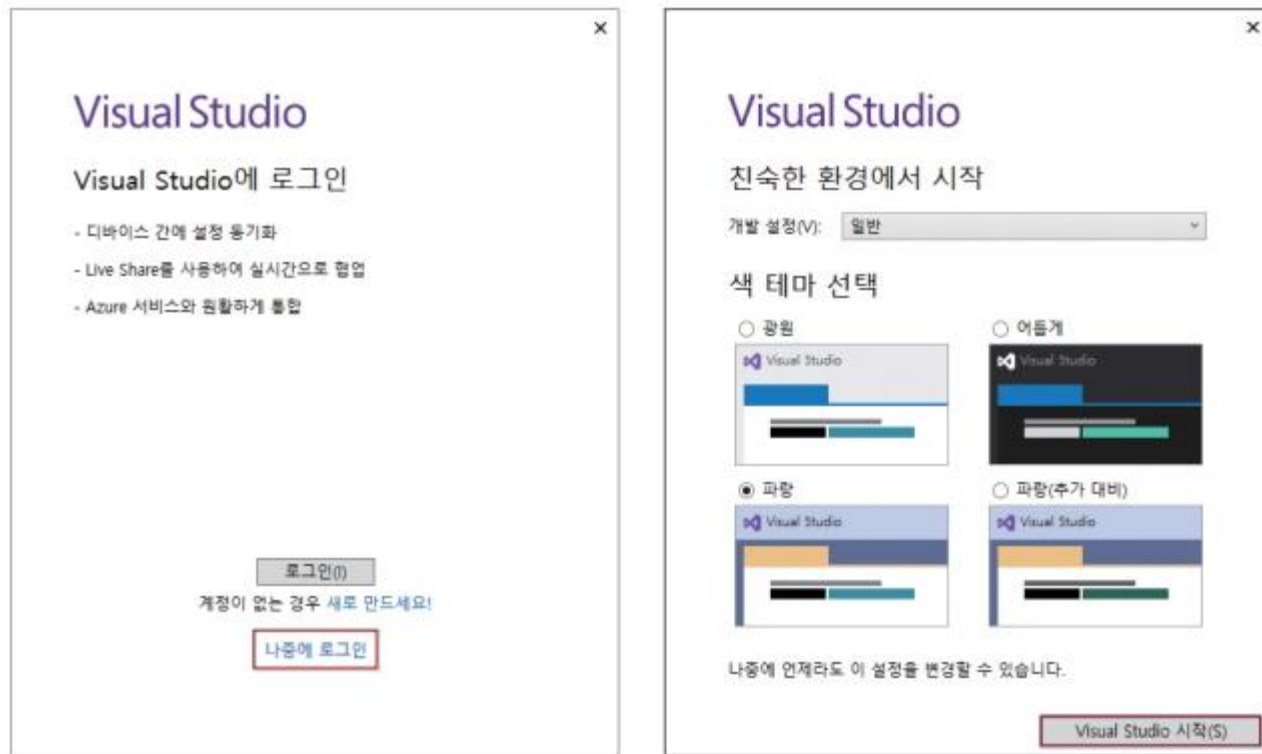


그림 1-15 로그인 및 친숙한 환경에서 시작



### 3. C 컴파일러 설치와 간단한 실습

#### 10 Visual Studio 초기 화면이 나옴



그림 1-16 Visual Studio 초기 화면

### 3. C 컴파일러 설치와 간단한 실습

**11** Visual Studio Community는 마이크로소프트 계정으로 로그인해야 기간에 제한 없이 무료로 사용할 수 있음(별도의 로그인 절차 없이 30일간 무료로 사용할 수 있으나, 30일이 지나면 다시 로그인해야 됨)

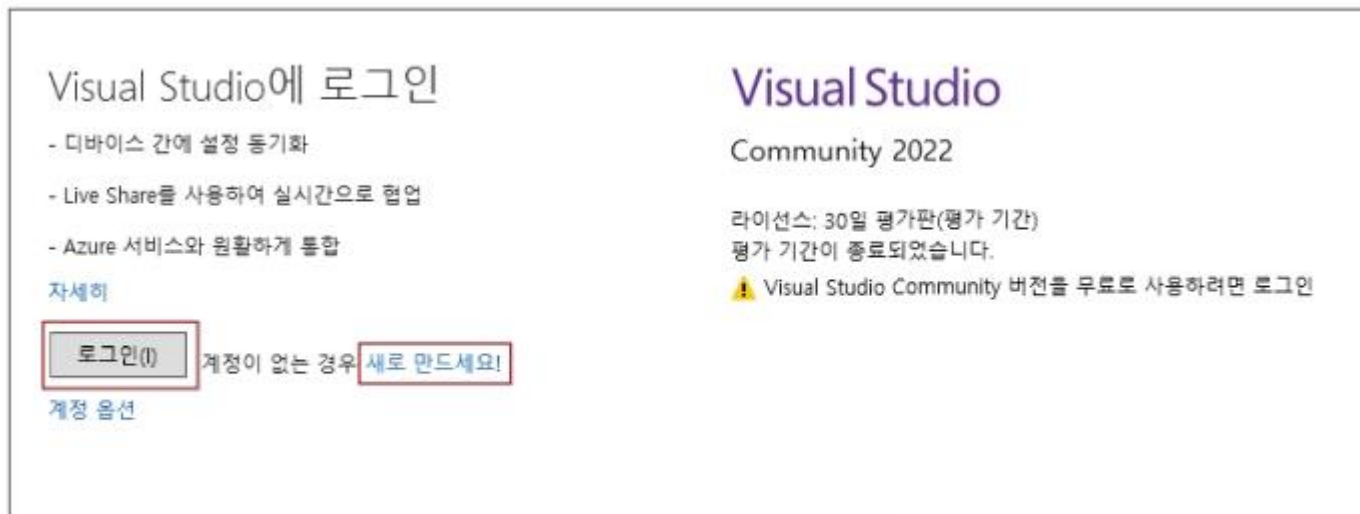


그림 1-17 [Visual Studio에 로그인] 화면

## 3. C 컴파일러 설치와 간단한 실습

### 3. 처음으로 작성해보는 간단한 프로그램

- 메모장은 코드를 입력하는 단순한 기능만 제공하지만 Visual Studio는 코드를 입력할 때 실수로 잘못 입력한 글자를 찾아주고 클릭 한 번으로 컴파일과 링크를 해결해주는 강력하고 편리한 통합 개발 환경을 제공

#### 여기서 잠깐 프로젝트와 소스 파일의 상관관계

- 단순한 C 컴파일러(예를 들어 Turbo C 등)를 사용한다면 C 소스 파일을 하나만 작성해서 컴파일과 링크를 수행하면 됨
- 하지만 Visual C++로 프로그래밍을 하려면 단순히 소스 파일만 필요한 것이 아니라 프로젝트를 먼저 생성해야 함
- 프로젝트는 지금 공부하는 C뿐만 아니라 고급 프로그래밍에서 중요한 개념
- 프로젝트의 개념은 오른쪽 그림과 같이 여러 개의 C 프로그램을 담아놓는 그릇이라고 보면 됨

## 3. C 컴파일러 설치와 간단한 실습

### 3. 처음으로 작성해보는 간단한 프로그램

- 프로젝트 생성

**01** 이 책에서 작성할 프로젝트를 모아놓기 위해 'C:\CookC' 폴더를 미리 생성

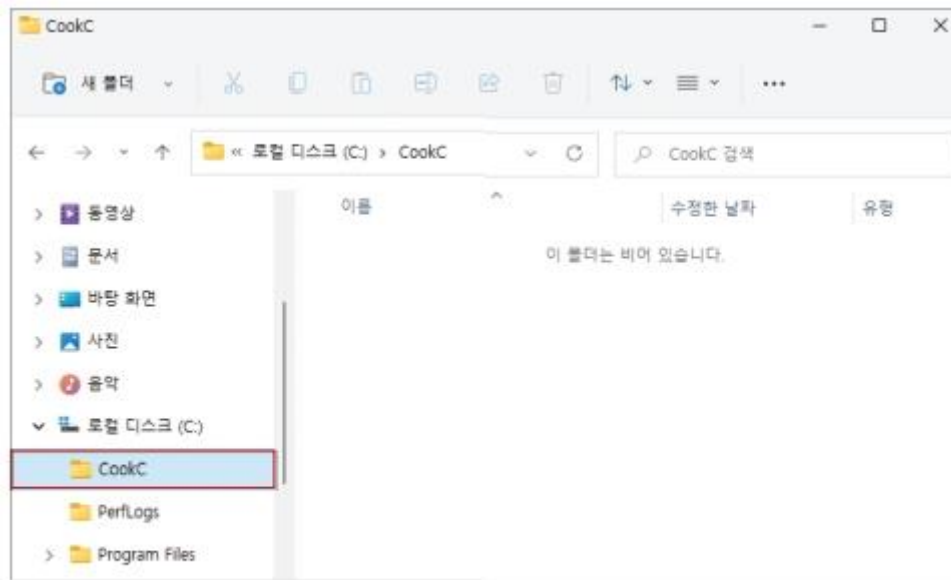


그림 1-18 폴더 생성

**02** 앞서 실행한 Visual Studio를 닫았다면 윈도우의 [시작] - [Visual Studio 2022]를 클릭해서 다시 실행

### 3. C 컴파일러 설치와 간단한 실습

**03** C 프로그램을 작성하려면 먼저 프로젝트를 생성, [시작 화면] 오른쪽 아래 [새 프로젝트 만들기]를 클릭



그림 1-19 새 프로젝트 생성 1

### 3. C 컴파일러 설치와 간단한 실습

**04** [새 프로젝트 만들기] 창 오른쪽의 [모든 언어]를 [C++]로 선택하고, [Windows 데스크톱 마법사]를 선택한 후 <다음>을 클릭



그림 1-20 새 프로젝트 생성 2

### 3. C 컴파일러 설치와 간단한 실습

**05** 프로젝트 이름 'First'로 입력 → 위치는 앞에서 생성한 폴더인 'C:\WCookC'를 입력하거나 선택 → '솔루션 및 프로젝트를 같은 디렉터리에 배치' 체크 → <만들기> 클릭

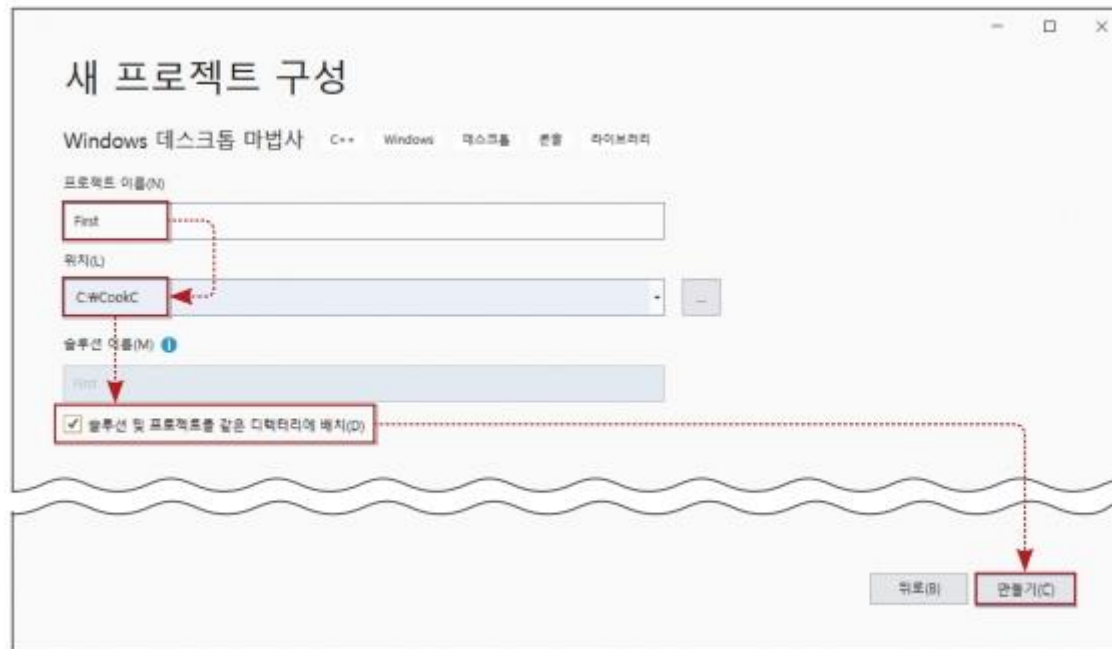


그림 1-21 새 프로젝트 생성 3

### 3. C 컴파일러 설치와 간단한 실습

**06** [Windows 데스크톱 프로젝트] 창에서 애플리케이션 종류는 '콘솔 애플리케이션 (.exe)'으로 되어 있을 것, 추가 옵션에서 '빈 프로젝트'에 체크하고 <확인>을 클릭

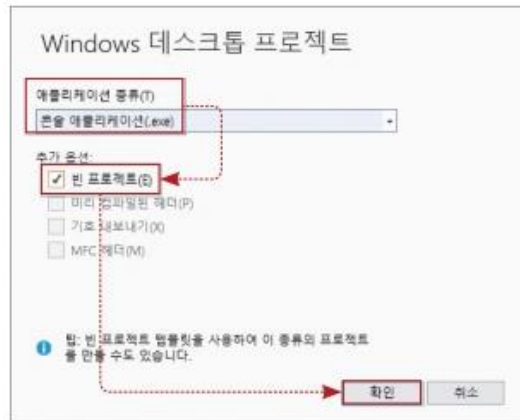


그림 1-22 설정

**07** 빈 프로젝트가 완성

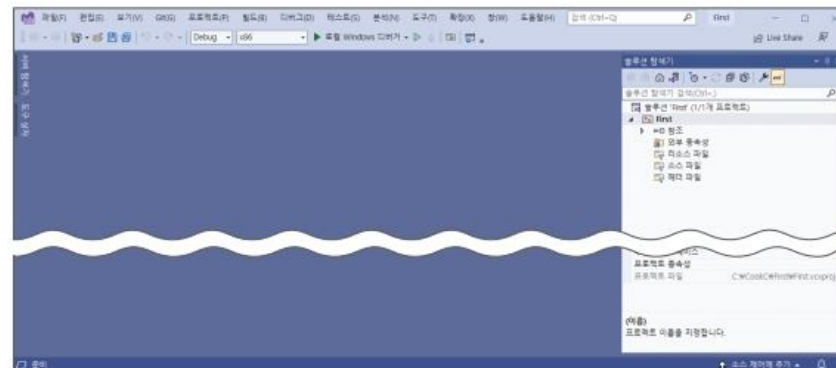


그림 1-23 프로젝트 생성 화면



### 3. C 컴파일러 설치와 간단한 실습

#### ■ C 프로그램 코딩

**01** 오른쪽 [솔루션 탐색기]의 프로젝트 이름(현재 First) 아래 '소스 파일' 폴더에서 마우스 오른쪽 버튼을 클릭하여 [추가] - [새 항목]을 선택

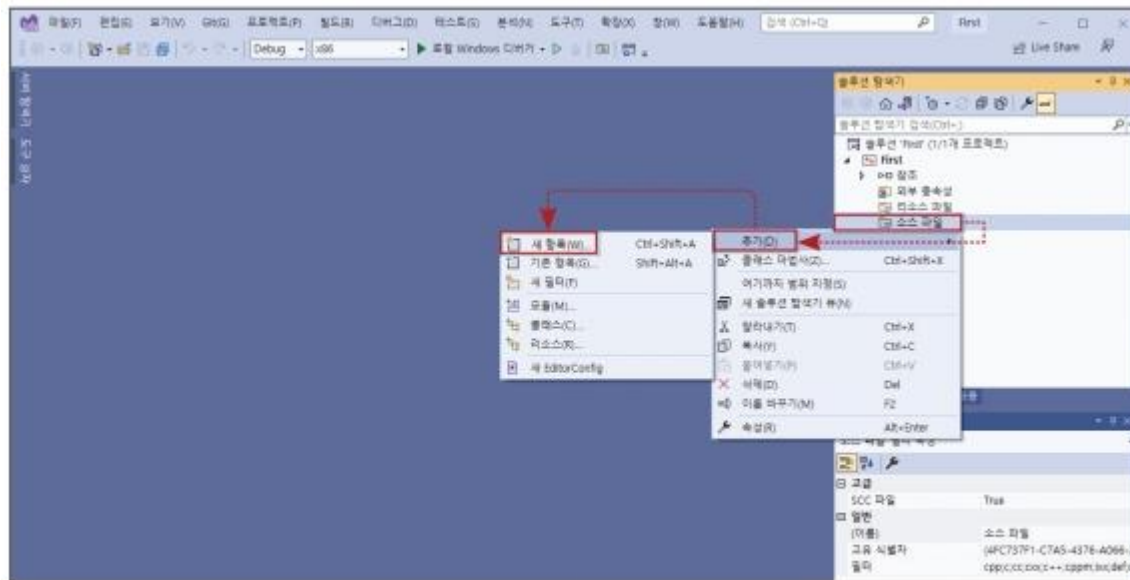


그림 1-24 프로젝트에 소스 파일 추가

### 3. C 컴파일러 설치와 간단한 실습

**02** [새 항목 추가] 창에서 왼쪽의 [Visual C++]를 선택하고, 'C++ 파일(.cpp)'을 선택한 뒤 이름에 'First.c'를 입력하고 <추가>를 클릭

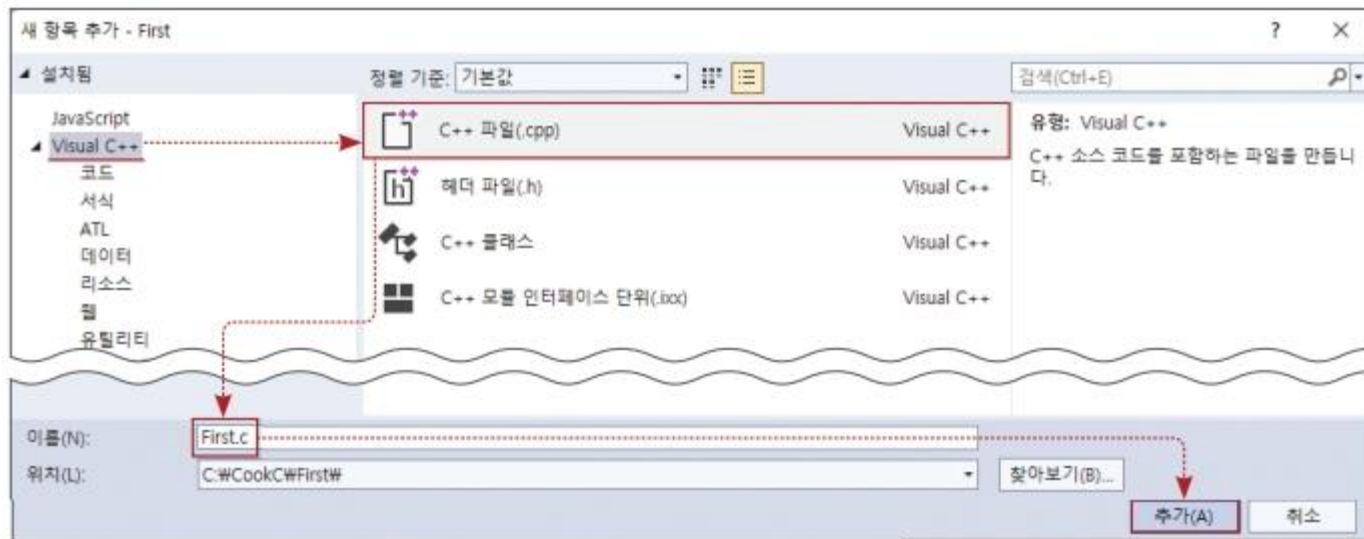


그림 1-25 First.c 파일 생성

### 3. C 컴파일러 설치와 간단한 실습

**03** 왼쪽에 빈 창이 나왔을 것, [그림 1-26]과 똑같이 입력하면 색깔로 표시된 글자가 보이는데 파란색 글자는 예약어(문법으로 약속된 글자) 입력을 완료했으면 [파일] - [First.c 저장] 메뉴를 선택하거나 파일 저장 아이콘을 클릭

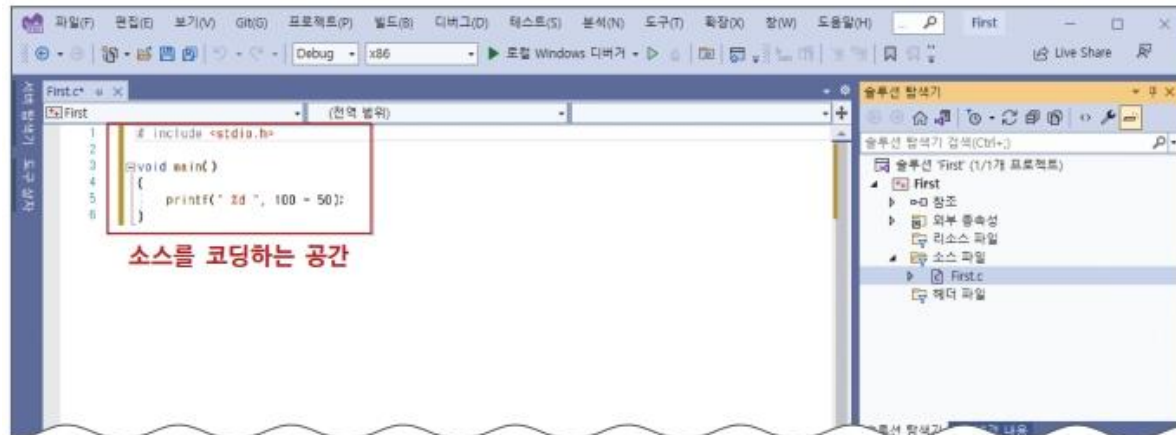


그림 1-26 C 소스 입력

#### 기본 1-2 처음 만드는 C 프로그램

1-2.c

01 #include <stdio.h>	—— 무조건 들어가는 부분이다(의미는 뒤에서 설명하겠다).
02	
03 void main( )	—— 프로그램의 시작 부분으로 항상 고정되어 있다.
04 {	—— 실제 작동하는 프로그램을 { }로 묶어준다.
05     printf(" %d ", 100 - 50);	—— 100-50의 값을 모니터에 출력하라는 의미이다.
06 }	

### 3. C 컴파일러 설치와 간단한 실습

- 빌드(컴파일 및 링크)

**01** 메뉴바의 [빌드] - [솔루션 빌드]를 선택, 단축키인 [Ctrl]+[Shift]+[B]를 눌러도 됨

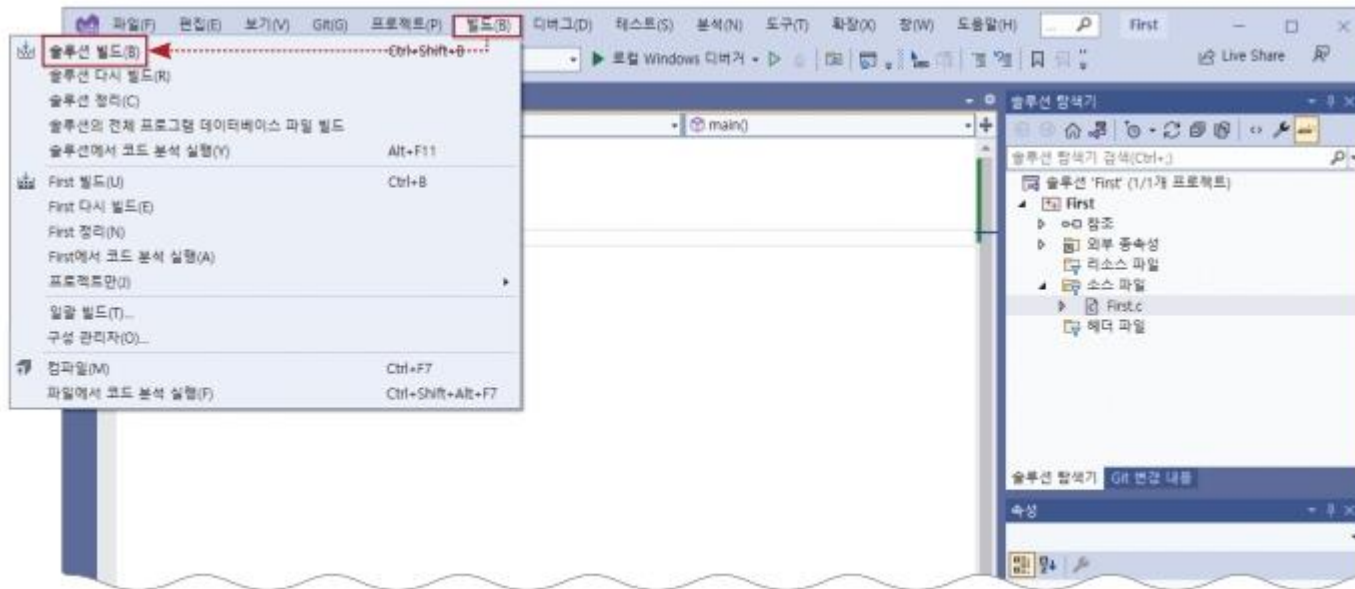
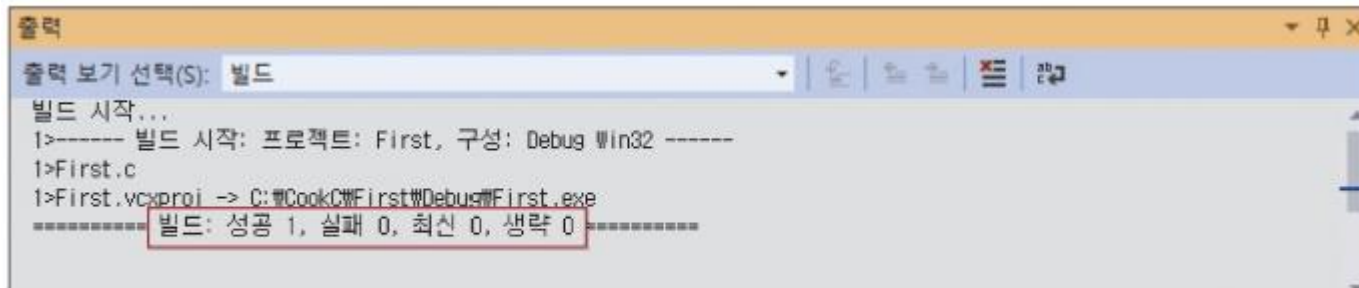


그림 1-27 빌드 실행

### 3. C 컴파일러 설치와 간단한 실습

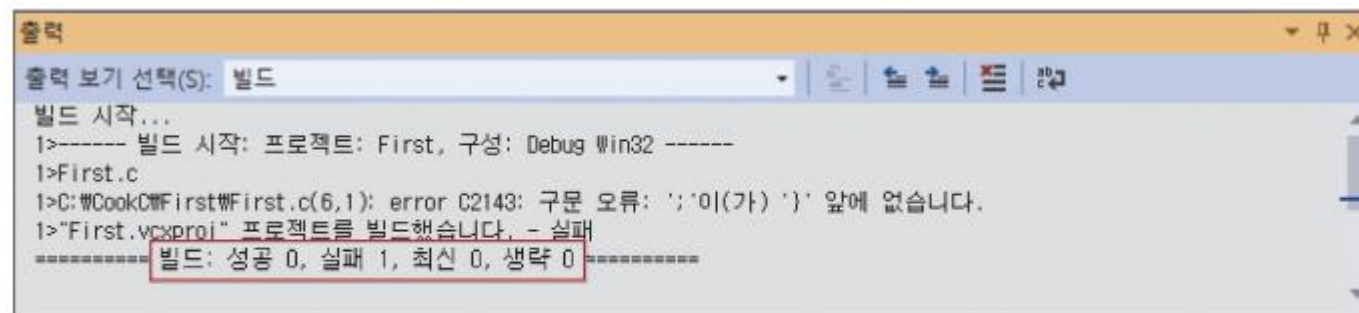
**02** 소스에 아무런 오류가 없다면 다음 화면과 같이 '빌드: 성공 1, 실패 0, 최신 0, 생략 0' 이라는 메시지가 나타남



```
출력
출력 보기 선택(S): 빌드
빌드 시작...
1>----- 빌드 시작: 프로젝트: First, 구성: Debug Win32 -----
1>First.c
1>First.vcxproj -> C:\CookC\First\Debug\First.exe
===== 빌드: 성공 1, 실패 0, 최신 0, 생략 0 =====
```

그림 1-28 빌드 결과: 이상 없음

**03** [기본 1-2]의 5행에서 세미콜론 (;)을 지우고, [F7]을 눌러서 다시 빌드, 결과가 '빌드: 성공 0, 실패 1, 최신 0, 생략 0'으로 표시, 이런 오류가 나온다면 대부분 소스가 틀렸기 때문



```
출력
출력 보기 선택(S): 빌드
빌드 시작...
1>----- 빌드 시작: 프로젝트: First, 구성: Debug Win32 -----
1>First.c
1>C:\CookC\First\First.c(6,1): error C2143: 구문 오류: ';'이(가) ')' 앞에 없습니다.
1>"First.vcxproj" 프로젝트를 빌드했습니다. - 실패
===== 빌드: 성공 0, 실패 1, 최신 0, 생략 0 =====
```

그림 1-29 빌드 결과: 오류 발생

### 3. C 컴파일러 설치와 간단한 실습

**04** [오류 목록] 탭을 클릭한 다음, 오류가 있는 부분을 마우스로 더블클릭하면 해당 위치로 바로 이동 → 이동한 곳에 닫는 중괄호(})가 옳게 들어가 있는데 오류가 되는 이유는 윗줄에 세미콜론이 빠져 있기 때문

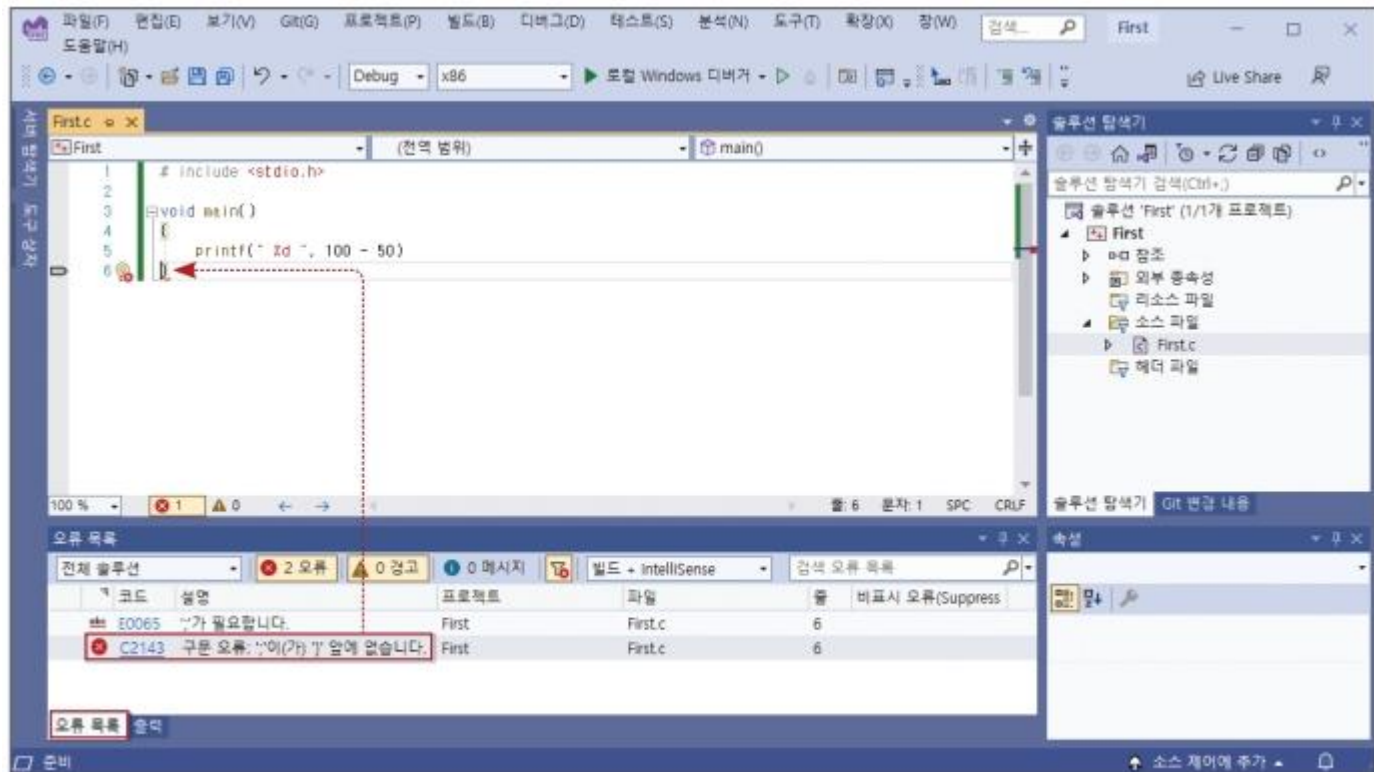


그림 1-30 소스에서 틀린 부분 찾기

### 3. C 컴파일러 설치와 간단한 실습

**05** 오류가 발생한 부분을 찾았으면 오류를 수정 → 5행의 맨 뒤에 세미콜론을 넣은 후 파일을 저장하고 [Ctrl]+[Shift]+[B]를 눌러서 다시 빌드

Q / U / I / Z | 세미콜론이 빠지면 세미콜론이 빠진 줄에 오류가 표시된다. (o x)

#### 여기서 잠깐 C 소스의 줄 바꿈

- C는 줄을 바꾼다고 문장이 끝나는 것이 아니라 세미콜론(;)이나 닫는 중괄호( } ) 등을 만나야 하나의 문장이 종료
- 다음 네 문장은 표현은 다르지만 모두 같은 의미

① `printf("%d", 100-50);`

② `printf("%d",  
100-50);`

③ `printf("%d",  
100-50)  
;`

④ `printf  
("%d", 100-50);`

### 3. C 컴파일러 설치와 간단한 실습

#### ■ 파일 실행

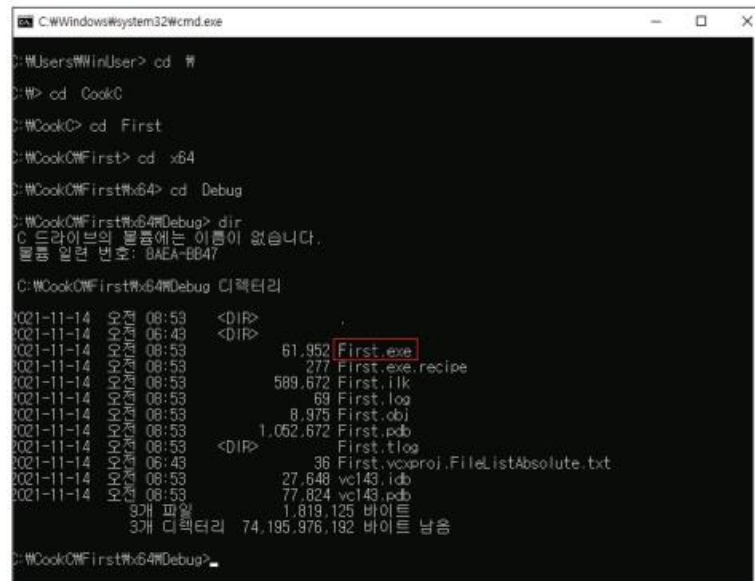
- 명령 프롬프트 창에서 실행하는 방법

#### 01 먼저 명령 프롬프트 창에서 직접 실행

Windows키+[R] → cmd 입력 → [Enter] → 명령 프롬프트 창 생성 →

다음 명령을 차례대로 입력하여 실행 파일이 들어 있는 폴더로 이동('cd'는 폴더 이동 명령이며 대문자와 소문자의 구분이 없음)

```
cd \  
cd CookC  
cd First  
cd Debug  
cd x64  
dir
```



```
C:\Windows\system32\cmd.exe  
C:\Users\WinUser> cd \  
C:\> cd CookC  
C:\CookC> cd First  
C:\CookC\First> cd x64  
C:\CookC\First\64> cd Debug  
C:\CookC\First\64\Debug> dir  
C 드라이브의 볼륨에는 이름이 없습니다.  
볼륨 일련 번호: 8AEA-BB47  
C:\CookC\First\64\Debug 디렉터리  
2021-11-14 08:53 <DIR> .  
2021-11-14 08:43 <DIR> ..  
2021-11-14 08:53 61,952 First.exe  
2021-11-14 08:53 277 First.exe.recipe  
2021-11-14 08:53 589,672 First.ilc  
2021-11-14 08:53 69 First.log  
2021-11-14 08:53 8,975 First.obj  
2021-11-14 08:53 1,052,672 First.pdb  
2021-11-14 08:53 <DIR> First.tlog  
2021-11-14 08:43 36 First.vcxproj.FileListAbsolute.txt  
2021-11-14 08:53 27,648 vc143.idb  
2021-11-14 08:53 77,824 vc143.pdb  
2021-11-14 08:53 9개 파일 1,819,125 바이트  
3개 디렉터리 74,195,976,192 바이트 남음  
C:\CookC\First\64\Debug>
```

그림 1-31 명령 프롬프트 창에서 결과 확인



### 3. C 컴파일러 설치와 간단한 실습

- 01** 먼저 명령 프롬프트 창에서 직접 실행 → 실행 파일인 'First.exe' 을 볼 수 있음  
실행 파일 이름인 'First'를 입력해서 실행하면 우리가 코딩 한 결과인 '50'이 나옴

실행 결과▼



```
C:\Windows\system32\cmd.exe
C:\CookC\First\64\Debug> First
50
C:\CookC\First\64\Debug>
```

- 02** 이번에는 '100-50'이 아니라 '100-99'의 결과가 나오도록 코딩  
[기본 1-2]를 그대로 사용하되 5행만 약간 수정, [Ctrl]+[S]를 눌러 저장

기본 1-3 소스 수정하기(계산값 수정)

1-3.c

```
01 #include <stdio.h>
02
03 void main( )
04 {
05     printf(" %d ", 100 - 99); —— 100-99의 값을 모니터에 출력한다.
06 }
```

### 3. C 컴파일러 설치와 간단한 실습

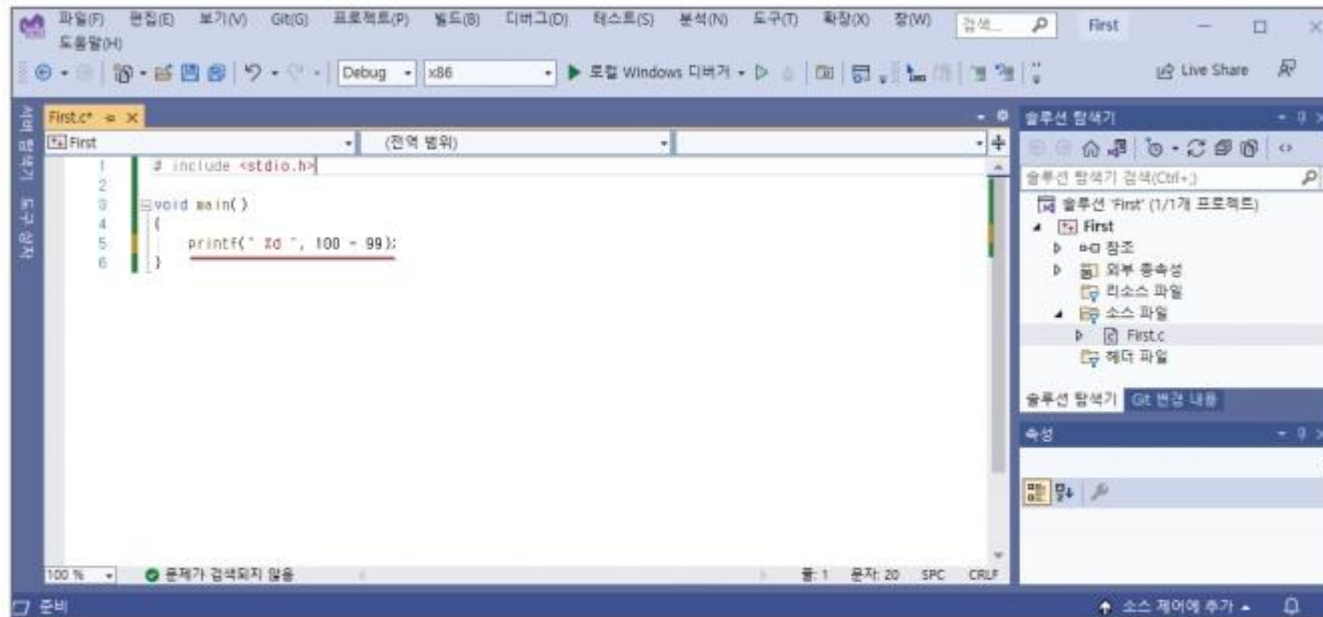
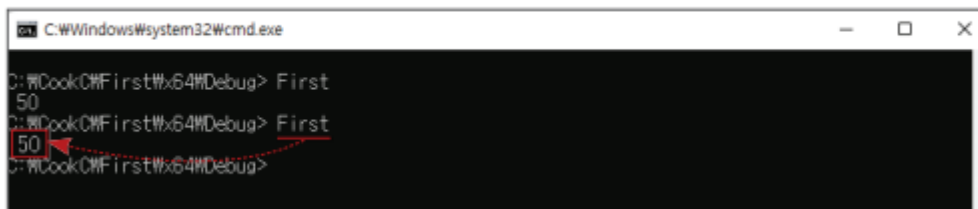


그림 1-32 소스 수정

**03** 새로운 결과를 보기 위해서 다시 명령 프롬프트 창에서 실행

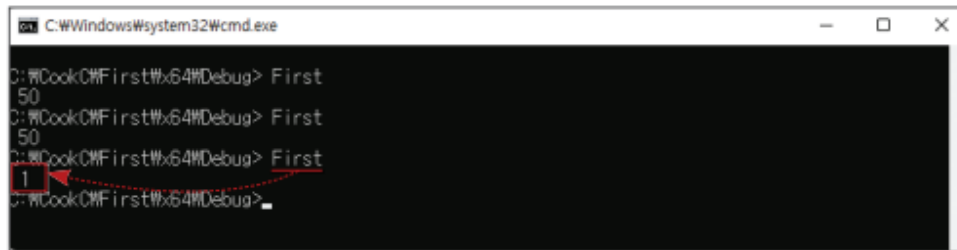
실행 결과▼



### 3. C 컴파일러 설치와 간단한 실습

**04** Visual Studio에서 [빌드] → [솔루션 빌드] 메뉴를 선택하거나 [Ctrl]+[Shift]+[B]를 눌러서 다시 빌드 → 출력 창에 '빌드: 성공 1, 실패 0, 최신 0, 생략 0'이라는 메시지가 나왔다면 다시 명령 프롬프트 창에서 실행

실행 결과▼



```
C:\Windows\system32\cmd.exe
C:\CookC\First\Debug> First
50
C:\CookC\First\Debug> First
50
C:\CookC\First\Debug> First
1
C:\CookC\First\Debug>
```

#### 여기서 잠깐 기본 도스 명령

명령(대·소문자 구분 없음)	설명
DIR	현재 폴더의 파일 목록을 보여준다.
CD [폴더 이름]	폴더로 이동한다.
COPY [원본] [사본]	원본 파일을 사본 파일로 복사한다.
TYPE [파일 이름]	텍스트 파일의 내용을 화면에 출력한다.
DEL [파일 이름]	파일을 삭제한다.
MD [폴더 이름]	새로운 폴더를 생성한다.
RD [폴더 이름]	폴더를 삭제한다(단, 폴더가 비어 있어야 한다).
CLS	화면을 깨끗이 지운다.

### 3. C 컴파일러 설치와 간단한 실습

- Visual Studio 환경에서 바로 실행하는 방법

**01** 빌드한 후 명령 프롬프트 창에서 바로 실행 결과를 보기 위해 메뉴의 [디버그]-[디버그하지 않고 시작]을 선택하거나, 단축키인 [Ctrl]+[F5]를 선택

만약 프로젝트가 완료되었다는 창이 나오면 <예(Y)>를 클릭

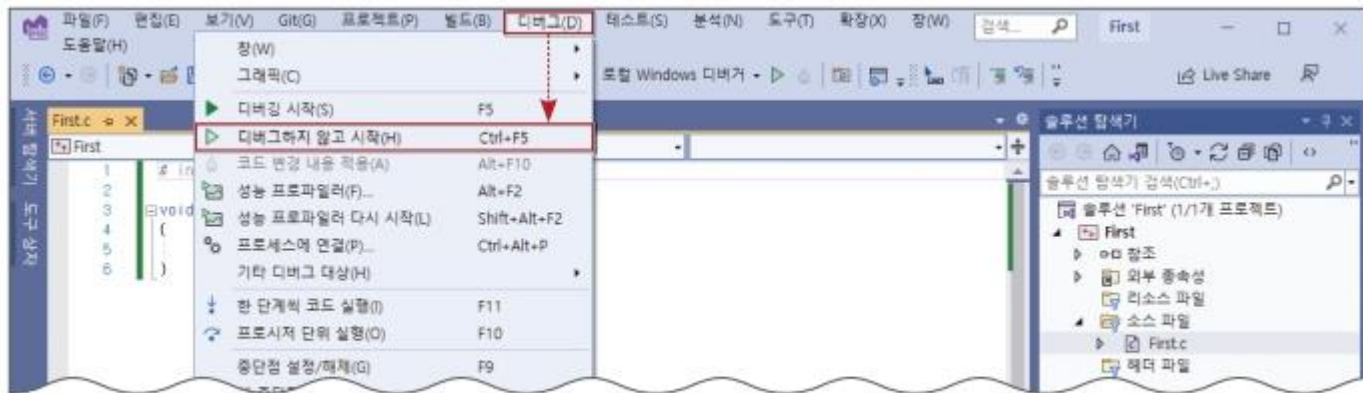
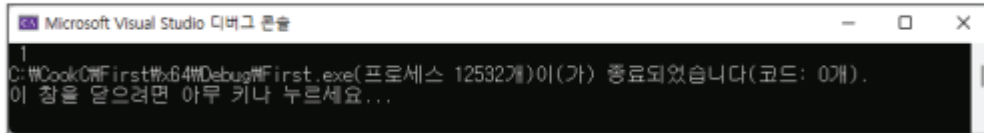


그림 1-33 Visual Studio에서 실행

### 3. C 컴파일러 설치와 간단한 실습

#### 02 명령 프롬프트 창이 나타나면서 바로 실행된 결과가 보일 것

실행 결과▼



#### 03 '%d'를 '%d \n'으로 수정

'\n'은 '다음 줄 넘기기'라는 의미로, 3장에서 상세하게 다루게 될 것

#### 기본 1-4 소스 수정하기(한 줄 띄우기)

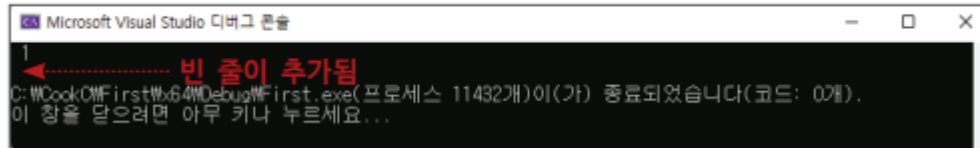
1-4.c

```
01 #include <stdio.h>
02
03 void main( )
04 {
05     printf(" %d \n", 100 - 99); —— 100-99의 값을 출력한 후 한 줄을 넘긴다.
06 }
```

### 3. C 컴파일러 설치와 간단한 실습

**04** 수정한 소스를 저장하고 다시 [Ctrl]+[F5] 를 눌러 실행한다.

실행 결과▼



### 3. C 컴파일러 설치와 간단한 실습

**05** 프로그램 작성 및 실행이 모두 끝났으면 메뉴의 [파일] → [솔루션 닫기]를 선택해서 현재 작업 중인 프로젝트를 종료

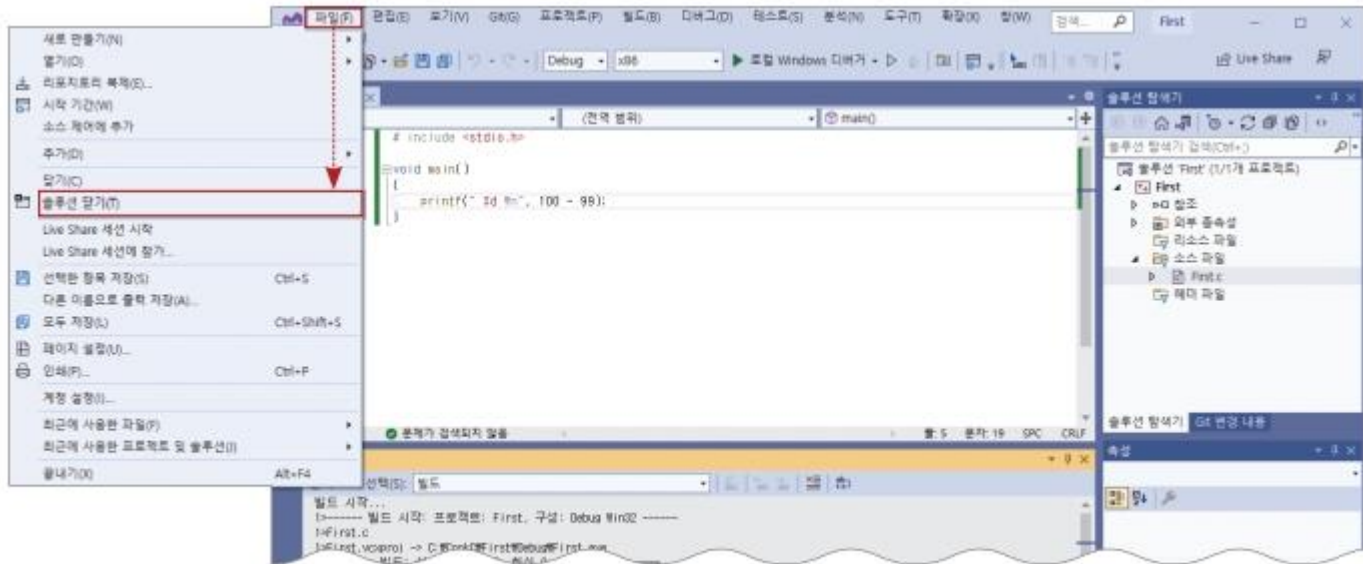


그림 1-34 프로젝트(솔루션) 닫기

**06** [원하는 작업을 선택하세요] 화면이 나오면 오른쪽 위 <x>를 클릭해서 화면을 닫음 → Visual Studio가 나오면 메뉴의 [파일] - [끝내기]를 선택해서 종료

### 3. C 컴파일러 설치와 간단한 실습

- 기존의 C 소스 다시 열기

**01** Visual Studio를 실행하고 오른쪽 [시작] 부분의 [프로젝트 또는 솔루션 열기]를 선택한 후 앞서 작업했던 'C:\WCookC\First' 폴더의 'First.sln' 파일을 선택

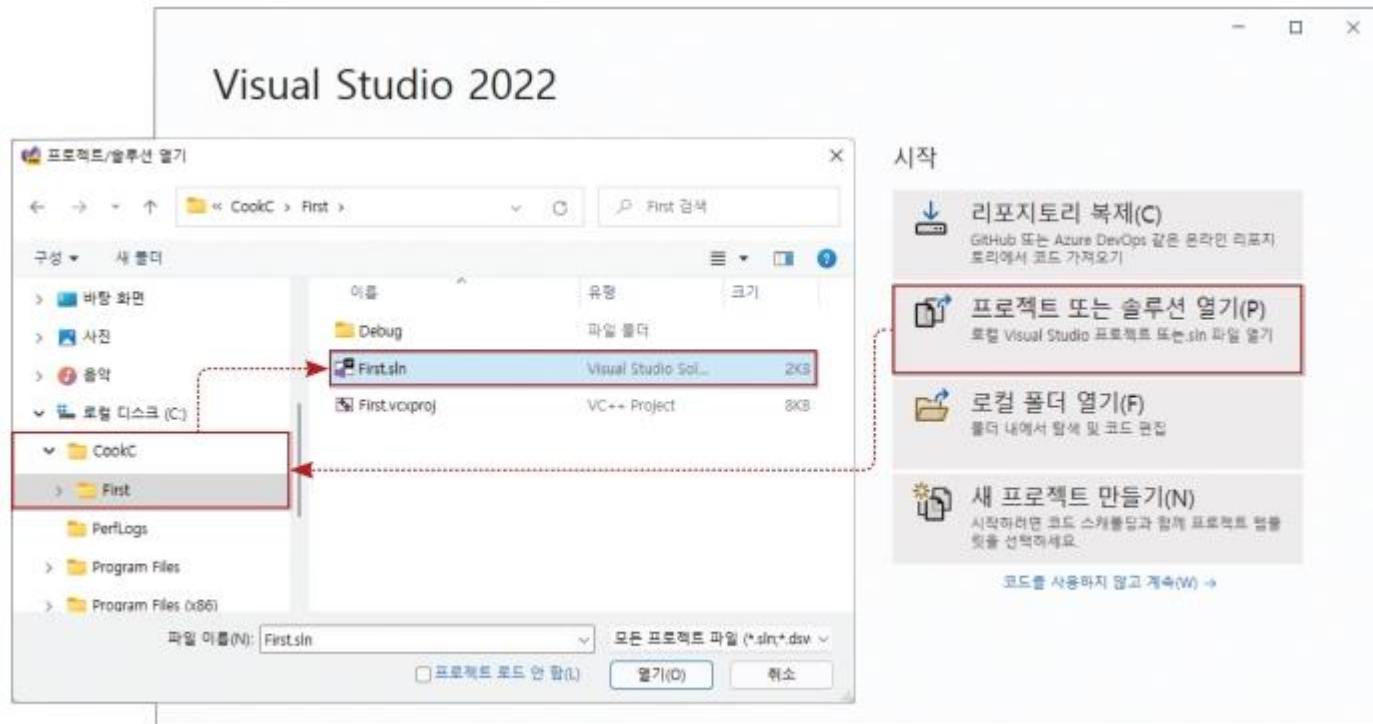


그림 1-35 기존 프로젝트 열기



### 3. C 컴파일러 설치와 간단한 실습

**02** First.c 파일이 보일 것, 만약 C 소스가 보이지 않는다면 왼쪽 아래의 [솔루션 탐색기] 탭을 클릭한 후 '소스 파일'를 클릭해서 확장하고, First.c를 더블클릭

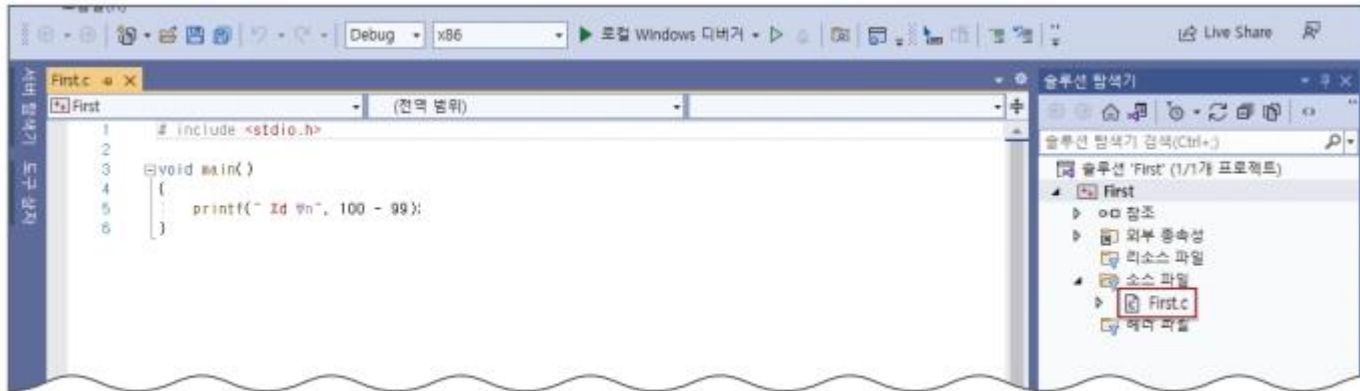


그림 1-36 기존 프로젝트 열기 완료

**여기서 잠깐 윈도우 11에서 파일의 확장명이 보이지 않는 경우**

- [파일 탐색기]를 실행하고 [레이아웃 및 보기 옵션] 아이콘을 클릭한 후, [표시] >> [파일 확장명]을 선택
- 참고로 윈도우 10에서는 [파일 탐색기]의 [보기] - [파일 확장명]을 체크

감사합니다!

