



PART 01. C 언어 입문

CHAPTER 01. 프로그램과 C 언어

Section 1.1

프로그램과 프로그래밍 언어

1. 프로그램과 프로그래밍 언어

■ 프로그램의 개념

- 컴퓨터가 특정 작업을 수행하도록 지시하는 명령어의 집합

■ 프로그래머

- **하드웨어:** 컴퓨터를 구성하는 기계
- 하드웨어를 동작시키는 프로그램을 만드는 사람

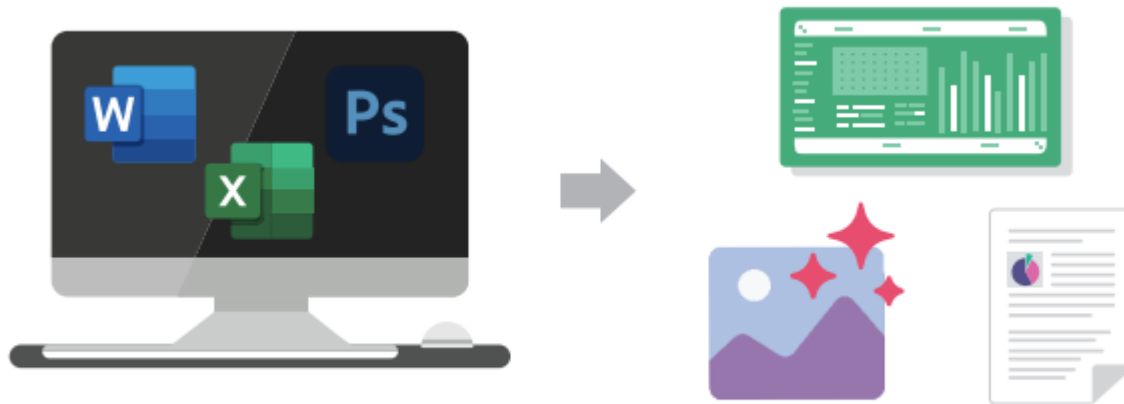


그림 1-1 프로그램을 사용한 작업 실행

1. 프로그램과 프로그래밍 언어

■ 프로그래밍의 개념

- 컴퓨터에 작업을 지시하는 명령 문서인 프로그램을 만드는 작업

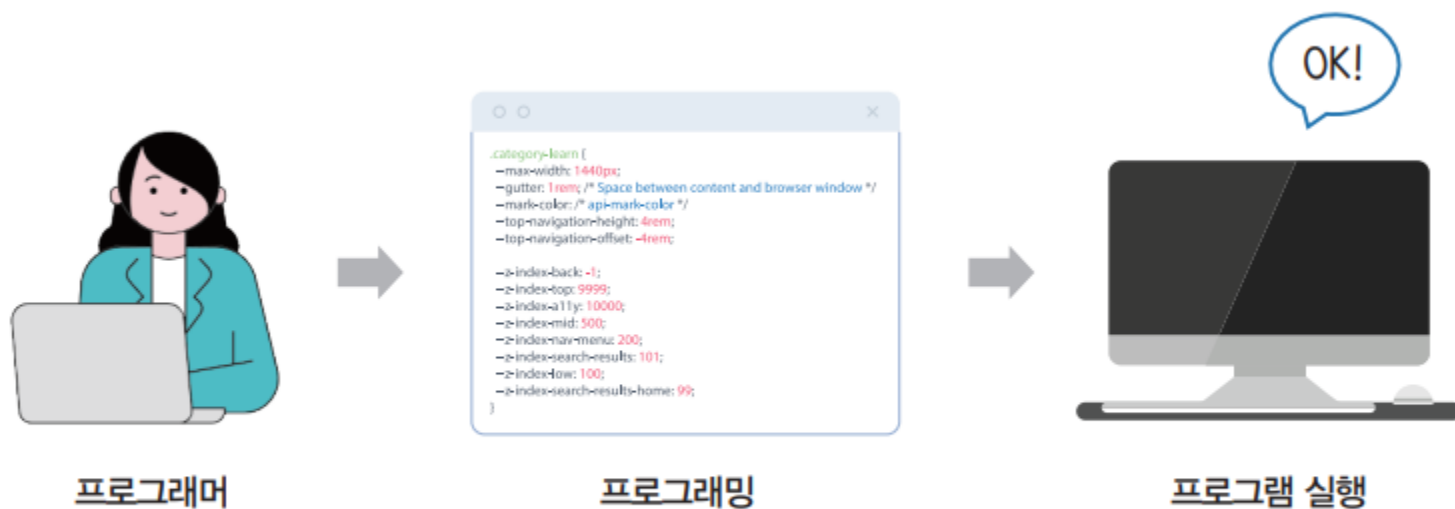


그림 1-2 프로그래밍 과정

1. 프로그램과 프로그래밍 언어

■ 프로그래밍 언어의 개념

- 프로그램을 만들기 위해서 컴퓨터에 명령할 때 사용하는 언어
- 다양한 수준과 목적에 맞게 설계

■ 기계어

- 컴퓨터가 이해하고 실행할 수 있는 2진수인 0과 1로 구성
- 컴퓨터와 직접 소통하는 일차적인 방법

■ 어셈블리어

- 기계어와 고급 언어의 중간 단계
- 컴퓨터가 이해하는 기계어를 사람이 이해하기 쉬운 기호나 단어로 표현

1. 프로그램과 프로그래밍 언어

■ 프로그래밍 언어의 개념

▪ 고급 언어

- 인간이 이해하기 쉽게 설계된 프로그래밍 언어
- 대표적인 예로는 C, C++, 파이썬, 자바 등

기계어	어셈블리어	고급 언어
<pre>01011110 10011111 01101010 11110000 00011001 10011100 01010101 10101111</pre>	<pre>LOAD A1 ADD A1, 100 MOV A1, M1 SUB B1, A1</pre>	<pre>#include <stdio.h> int main() { printf("Hello\n"); }</pre>

그림 1-3 기계어, 어셈블리어, 고급 언어의 예

1. 프로그램과 프로그래밍 언어

■ 프로그래밍 언어의 개념

■ 컴파일러

- 프로그래밍 언어로 작성된 코드 전체를 컴퓨터가 이해할 수 있는 기계어로 변환하는 프로그램

■ 인터프리터

- 프로그래밍 언어로 작성된 코드를 한 줄씩 읽어 해당 명령을 실행하는 프로그램

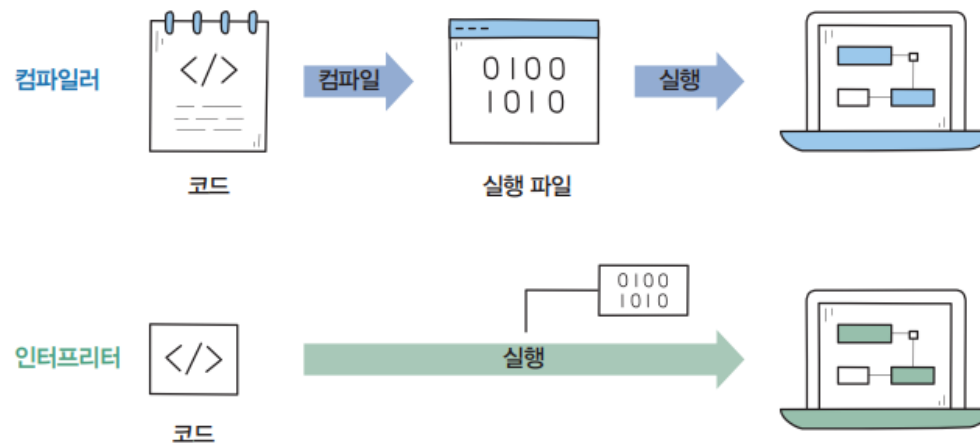


그림 1-4 컴파일러와 인터프리터

1. 프로그램과 프로그래밍 언어

■ 프로그래밍 언어의 종류

■ 절차적 프로그래밍 언어

- 프로그램을 연속적인 계산 단계로 나누고 각 단계에서 특정 작업을 수행

■ 객체지향 프로그래밍 언어

- 데이터와 해당 데이터를 처리하는 함수(메서드)를 '객체'라는 하나의 단위로 묶는 방식 사용

■ 함수형 프로그래밍 언어

- 자료 처리를 수학적 함수의 계산으로 취급하고 상태와 가변 데이터를 멀리하는 프로그래밍 언어

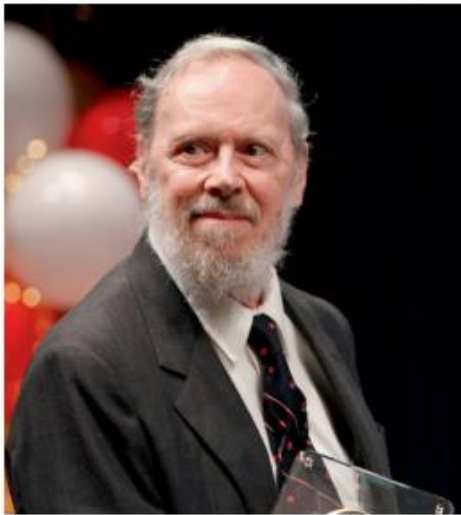
Section 1.2

C 언어의 개요

2. C 언어의 개요

■ C 언어의 탄생

- 1972년 미국 AT&T 벨 연구소에서 개발
- 데니스 리치 (Dennis Ritchie), 켄 톰프슨 (Ken Thompson)
- 유닉스 재작성을 위해 개발



(a) 데니스 리치



(b) C 언어의 로고

그림 1-5 C 언어의 창시자와 로고 © High Tech History

2. C 언어의 개요

■ C 언어의 특징

- 범용적인 프로그래밍 언어
 - 다양한 프로그래밍 분야에 사용 가능
- 저급 언어와 고급 언어의 장점이 반영된 언어
 - 저급 언어의 하드웨어 제어 능력
 - 고급 언어의 개발 편의성
- 간결한 표현이 가능한 언어
 - 복잡한 로직도 몇 줄의 코드로 표현 가능
- 이식성이 좋은 언어
 - 다양한 플랫폼에서 C 컴파일러 제공

Section 1.3

C 프로그램 개발 환경 구축

3. C 프로그램 개발 환경 구축

■ 프로그램 개발 과정

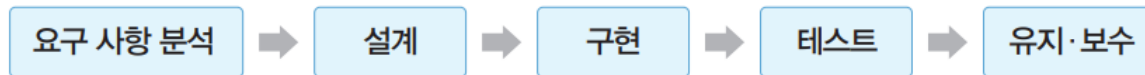


그림 1-6 프로그램 개발을 위한 5단계

3. C 프로그램 개발 환경 구축

■ 프로그램 개발 과정

- **요구 사항 분석**
 - 개발할 프로그램의 목적, 사용자와 시스템에 필요한 요구 사항 파악
- **설계**
 - 얻은 정보를 토대로 시스템의 아키텍처를 설계
- **구현**
 - 정의된 아키텍처에 따라 실제 코드 작성
- **테스트**
 - 설계대로 정확히 작동하는지 검증
- **유지 · 보수**
 - 소프트웨어 배포 이후에도 지속적인 관리 필요

3. C 프로그램 개발 환경 구축

■ C 프로그램 작성과 실행

- C 언어로 프로그램을 작성할 때는 일반적으로 [그림 1-7]과 같은 단계로 진행

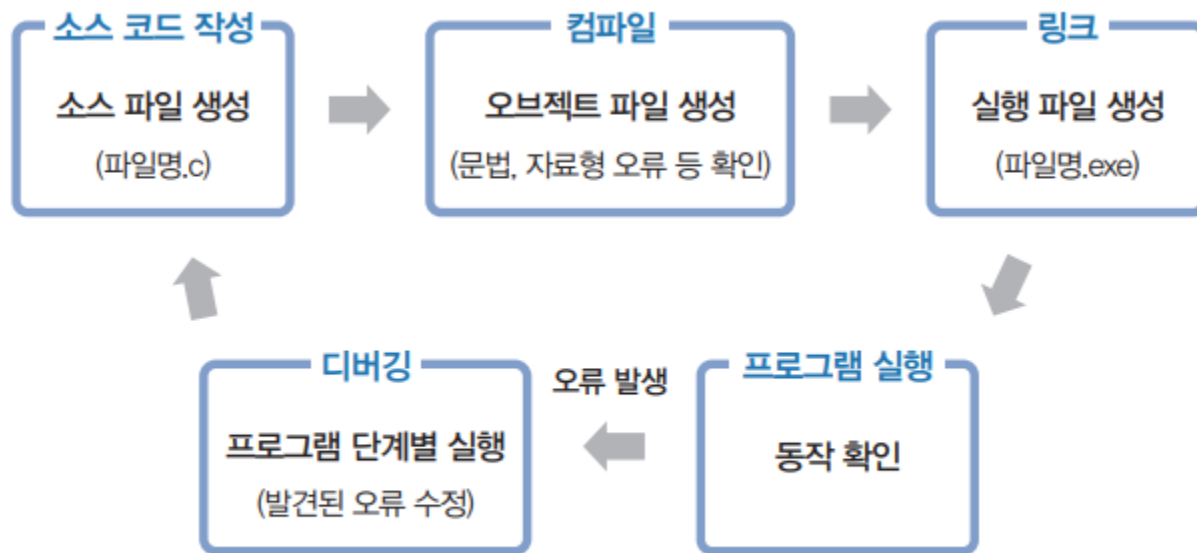


그림 1-7 C 프로그램 작성 순서

3. C 프로그램 개발 환경 구축

■ C 프로그램 작성과 실행

▪ C 프로그램 개발

- 소스 코드 작성

- 확장자가 '.c'인 파일로 저장



그림 1-8 소스 코드 작성

3. C 프로그램 개발 환경 구축

■ C 프로그램 작성과 실행

▪ C 프로그램 개발

- 컴파일과 링크

- 컴파일 단계: 소스 코드를 컴퓨터가 이해하게 변환
- 링크 단계: 컴파일로 생성된 오브젝트 파일을 모아 하나의 실행 파일로 결합

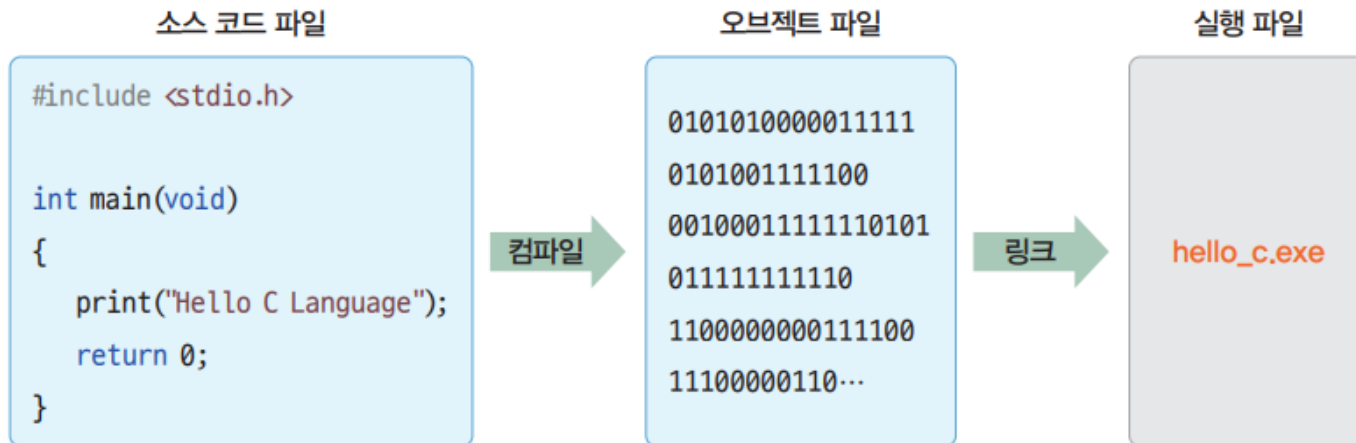


그림 1-9 오브젝트 파일 생성

3. C 프로그램 개발 환경 구축

■ C 프로그램 작성과 실행

▪ C 프로그램 개발

- 프로그램 실행과 디버깅

- 생성된 실행 파일로 프로그램을 실행하여 그 동작을 확인
- 디버깅: 프로그램을 단계별로 실행하여 문제를 수정하는 과정



그림 1-10 프로그램 출력

3. C 프로그램 개발 환경 구축

■ 컴파일러의 종류

▪ 통합 개발 환경(IDE)

- **Dev C++**

- C와 C++를 위한 무료 통합 개발 환경
- GCC를 내부 컴파일러로 사용

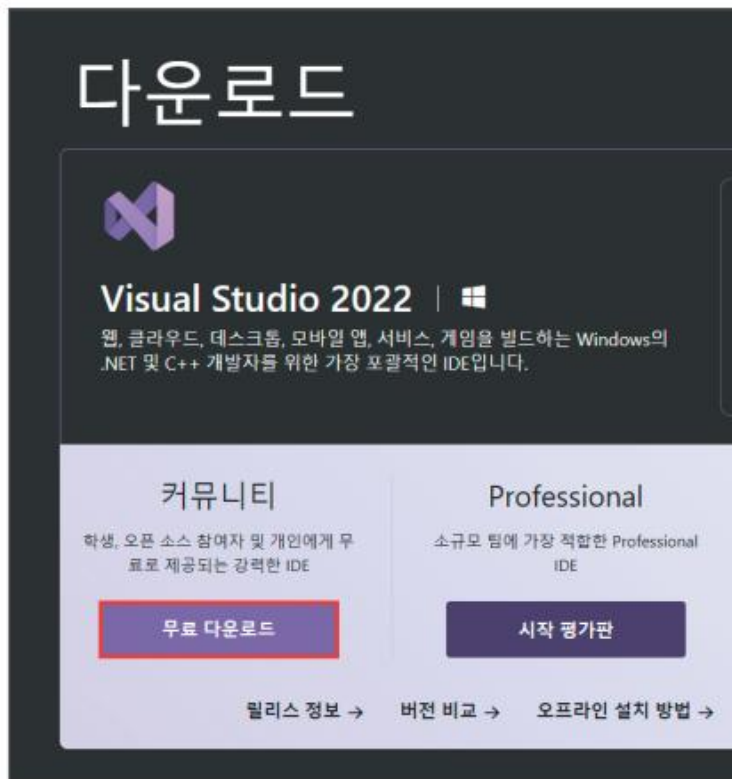
- **비주얼 스튜디오 2022**

- C, C++, C#, 비주얼 베이직 등 다양한 프로그래밍 언어 지원
- 사용자 친화적인 인터페이스, 코드 자동 완성, 디버깅 도구, 코드 분석 기능 등 제공
- 닷넷(.NET) 플랫폼 통합, 다양한 라이브러리, API 접근성 제공

3. C 프로그램 개발 환경 구축

■ 비주얼 스튜디오 2022 설치

- <https://visualstudio.microsoft.com/ko/downloads/>에 접속
- 커뮤니티 항목의 <무료 다운로드>를 클릭



TIP 한빛아카데미 유튜브에서 비주얼 스튜디오 2022 설치 영상을 제공한다. 아래 QR 코드로 접속할 수 있다.



3. C 프로그램 개발 환경 구축

■ 비주얼 스튜디오 2022 설치

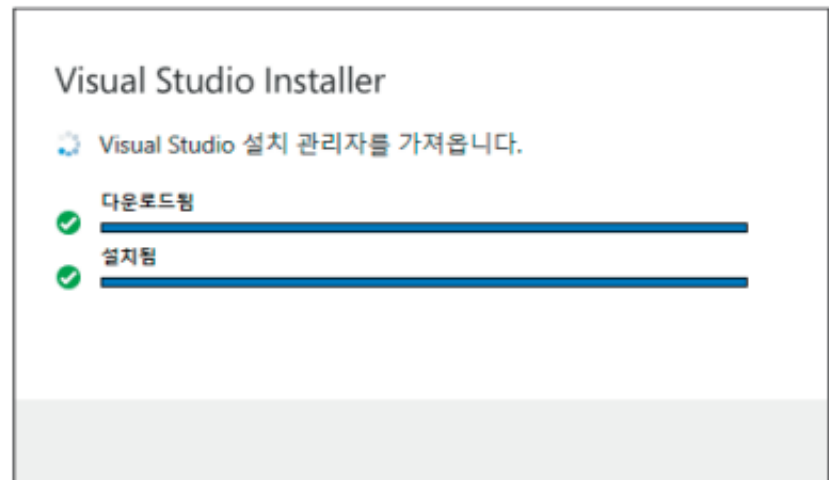
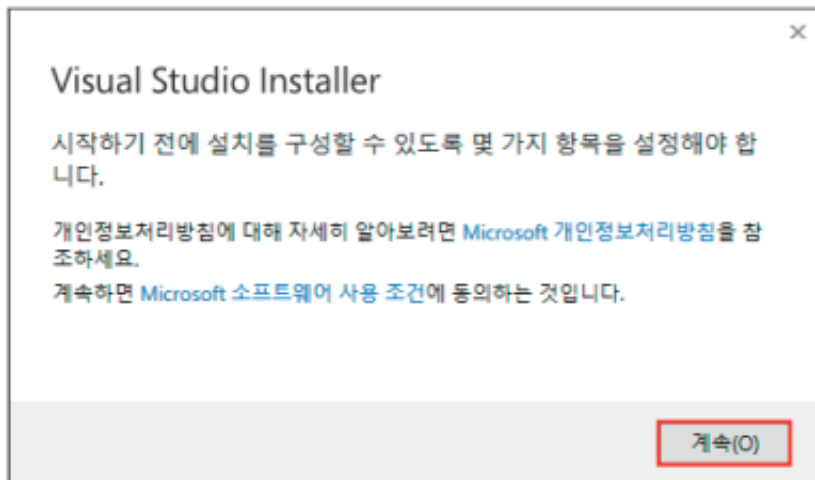
- 설치 파일인 'VisualStudioSetup.exe'가 자동 다운로드
- 자동 다운로드되지 않는다면 <여기를 클릭하여 다시 시도하세요>를 클릭



3. C 프로그램 개발 환경 구축

■ 비주얼 스튜디오 2022 설치

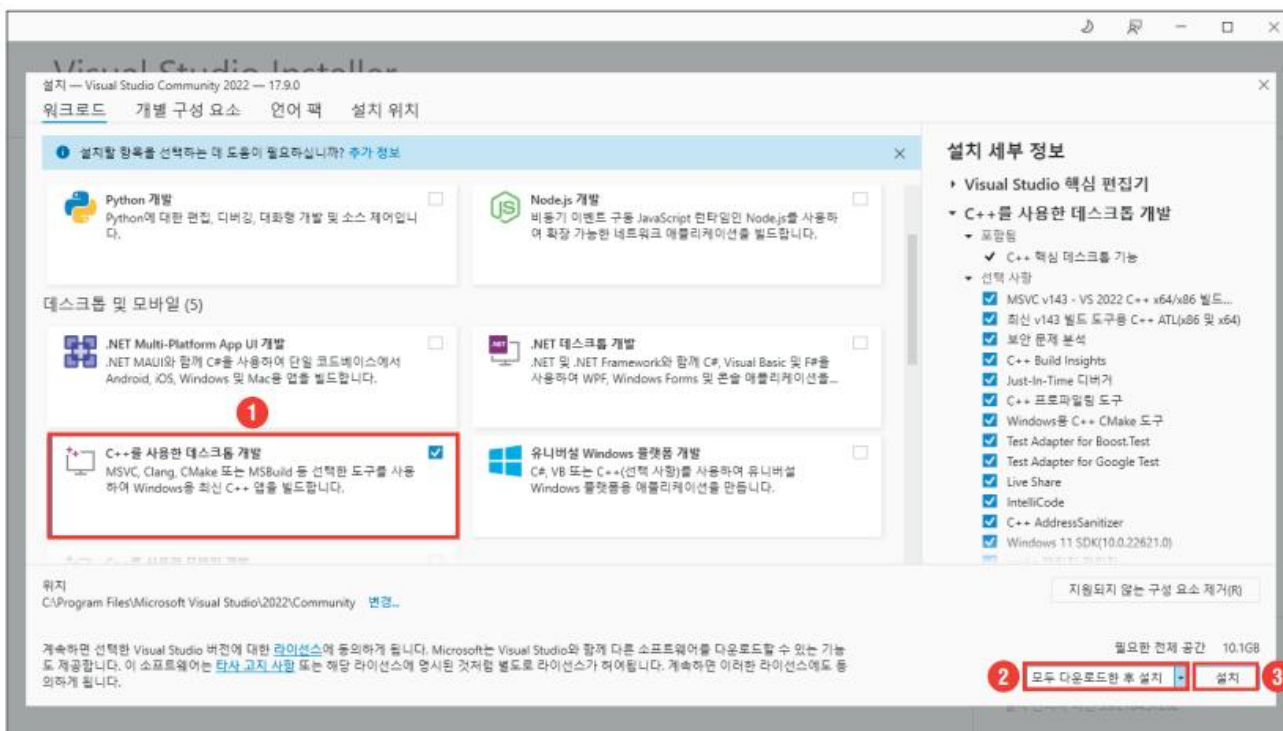
- 다운로드한 파일을 실행하고 <계속> 클릭



3. C 프로그램 개발 환경 구축

■ 비주얼 스튜디오 2022 설치

- 설치 화면이 나타나면 [C++를 사용한 데스크톱 개발]을 선택
- 오른쪽 하단의 [모두 다운로드한 후 설치]를 선택 후 <설치> 클릭



3. C 프로그램 개발 환경 구축

■ 비주얼 스튜디오 2022 설치

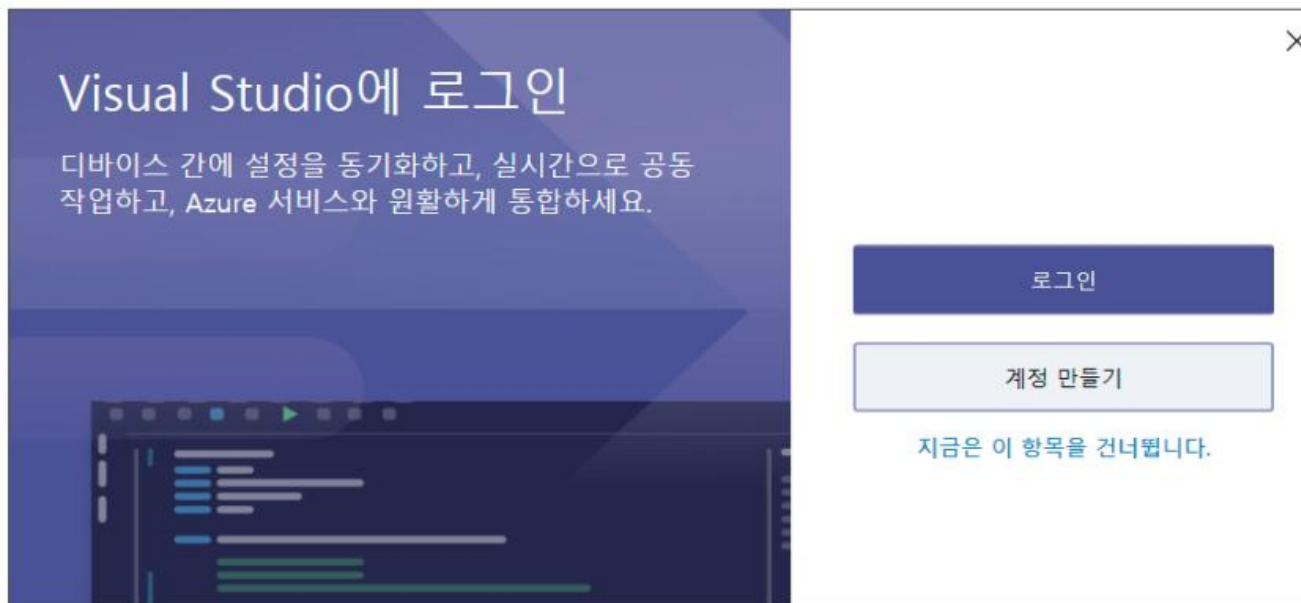
- 다음과 같은 화면이 나타나면 설치가 완료될 때까지 대기



3. C 프로그램 개발 환경 구축

■ 비주얼 스튜디오 2022 설치

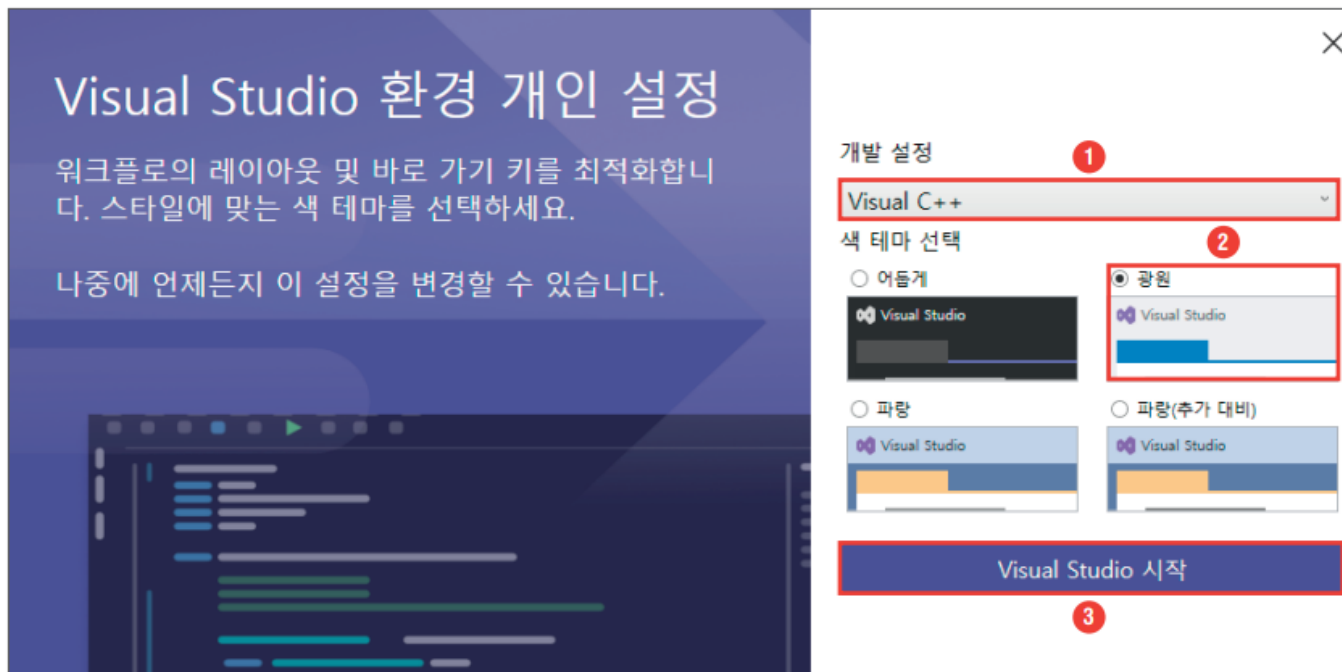
- 마이크로소프트 계정이 있다면 로그인
- 계정이 없다면 계정을 생성하여 로그인



3. C 프로그램 개발 환경 구축

■ 비주얼 스튜디오 2022 설치

- [Visual C++]를 선택하고 원하는 색 테마를 지정
- <Visual Studio 시작> 클릭



Section 1.4

C 프로그램 작성

4. C 프로그램 개발 환경 구축

■ 솔루션과 프로젝트

■ 솔루션

- 하나 이상의 프로젝트를 묶어 관리하는 컨테이너
- 프로젝트들의 관계, 빌드, 순서, 공통 설정 등을 정의

■ 프로젝트

- 관련 파일, 소스 코드, 설정 등이 포함된 개발 작업의 기본 단위

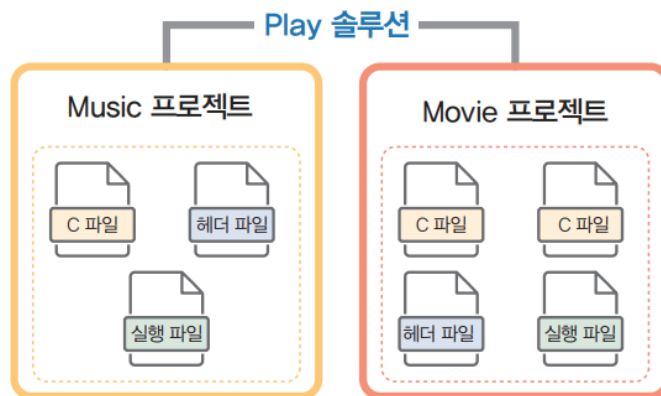


그림 1-11 솔루션과 프로젝트의 범위

4. C 프로그램 개발 환경 구축

■ 프로젝트 생성

- 비주얼 스튜디오 2022 실행
- 오른쪽의 시작 항목에서 [새 프로젝트 만들기] 선택



4. C 프로그램 개발 환경 구축

■ 프로젝트 생성

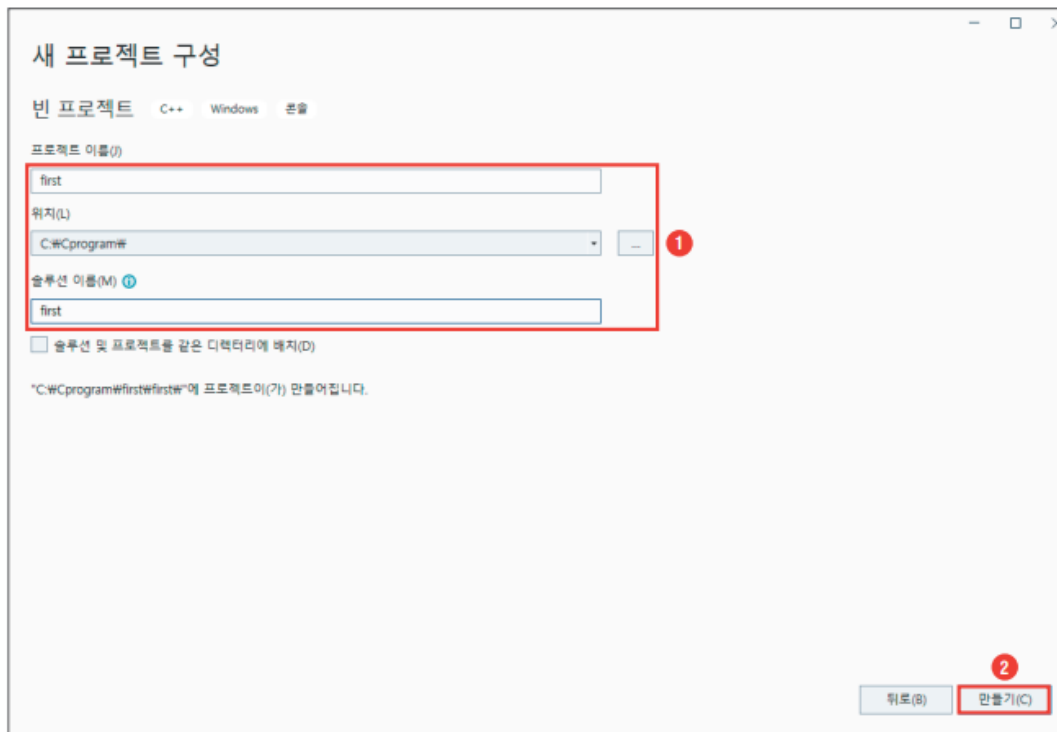
- [C++]로 설정한 뒤 [빈 프로젝트]를 선택하고 <다음> 클릭



4. C 프로그램 개발 환경 구축

■ 프로젝트 생성

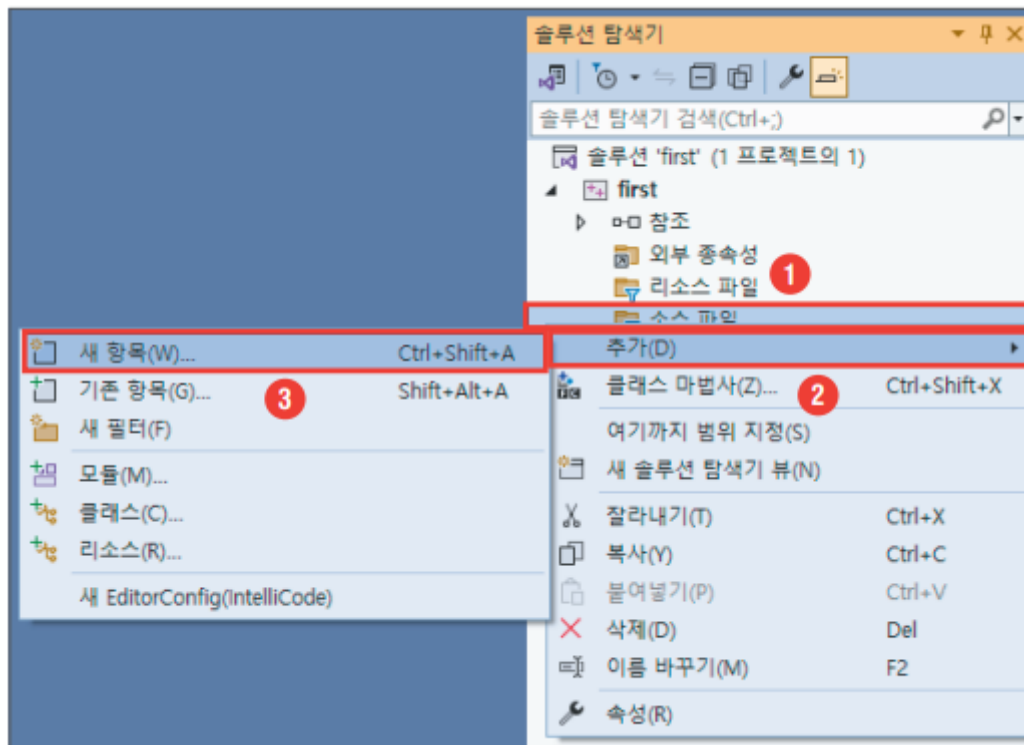
- '프로젝트 이름', '위치', '솔루션 이름' 을 설정하고 <만들기> 클릭
 - C 드라이브에 'Cprogram'폴더 생성 및 디렉터리 설정
 - 첫 프로젝트의 이름은 'first'로 작성



4. C 프로그램 개발 환경 구축

■ 소스 코드 작성

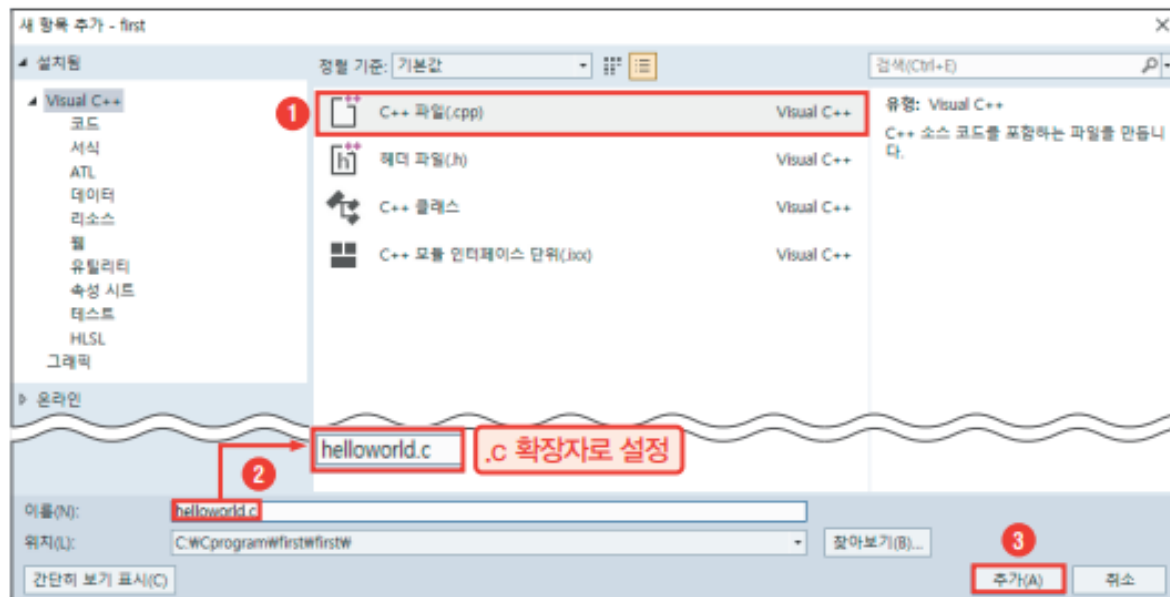
- 솔루션 탐색기의 [소스 파일]에서 마우스 오른쪽 버튼 클릭
- [추가]-[새 항목] 선택



4. C 프로그램 개발 환경 구축

■ 소스 코드 작성

- 'C++ 파일(.cpp)'을 선택한 뒤 이름에 'helloworld.c'를 입력하고 <추가> 클릭



4. C 프로그램 개발 환경 구축

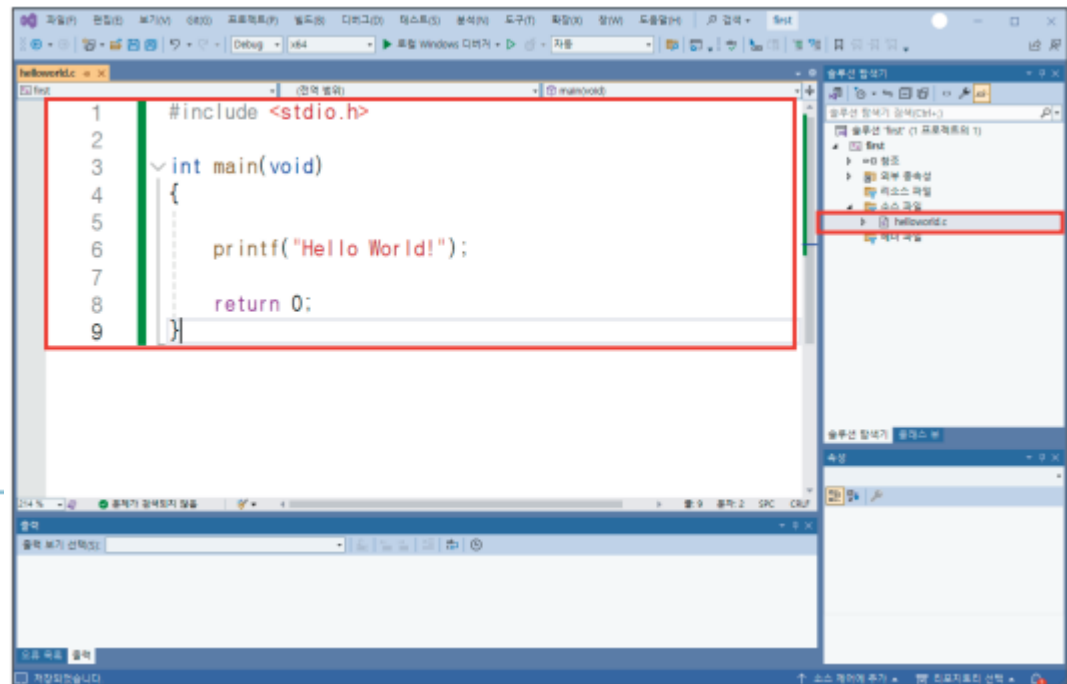
■ 소스 코드 작성

- 소스 파일이 생성되면 코드 편집기에서 다음과 같이 코드 작성

첫 프로그램 만들기

```
01 #include <stdio.h>
02
03 int main(void)
04 {
05
06     printf("Hello World!");
07
08     return 0;
09 }
```

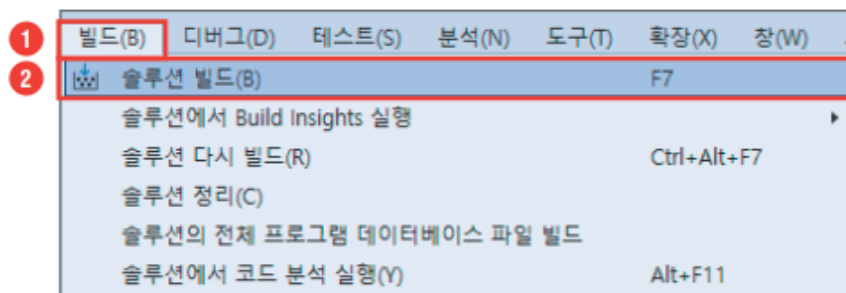
helloworld.c



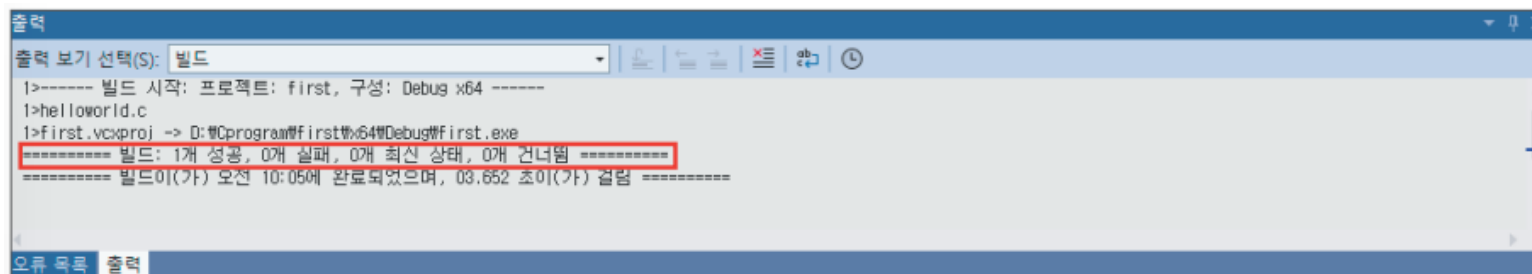
4. C 프로그램 개발 환경 구축

■ 빌드

- **빌드:** 소스 코드가 작성된 프로젝트를 실행 파일로 만드는 과정
- 상단에서 [빌드]-[솔루션 빌드] 선택



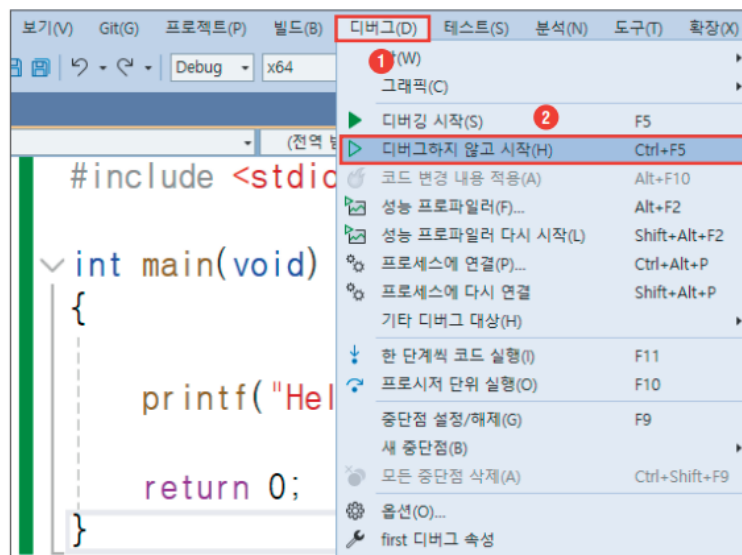
- 빌드에 성공하면 다음과 같이 '빌드: 1개 성공, 0개 실패, 0개 최신 상태, 0개 건너뛴'이라는 메시지 확인 가능



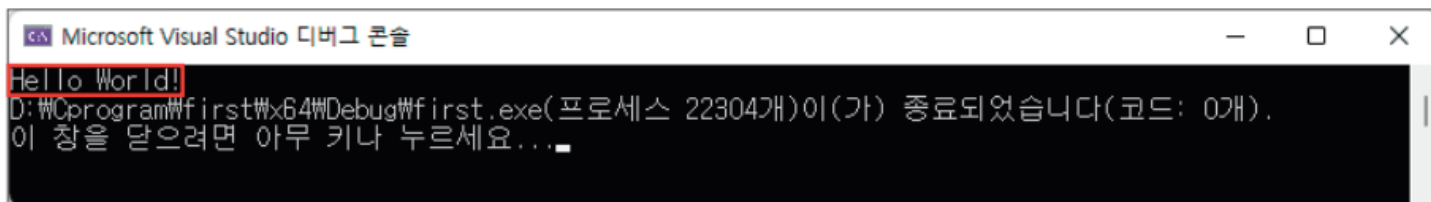
4. C 프로그램 개발 환경 구축

■ 디버깅과 실행

- 앞의 과정이 제대로 진행되었다면 [디버그]-[디버그하지 않고 시작] 선택



- 별다른 오류가 발견되지 않는다면 콘솔창이 나타나고 Hello World!가 출력

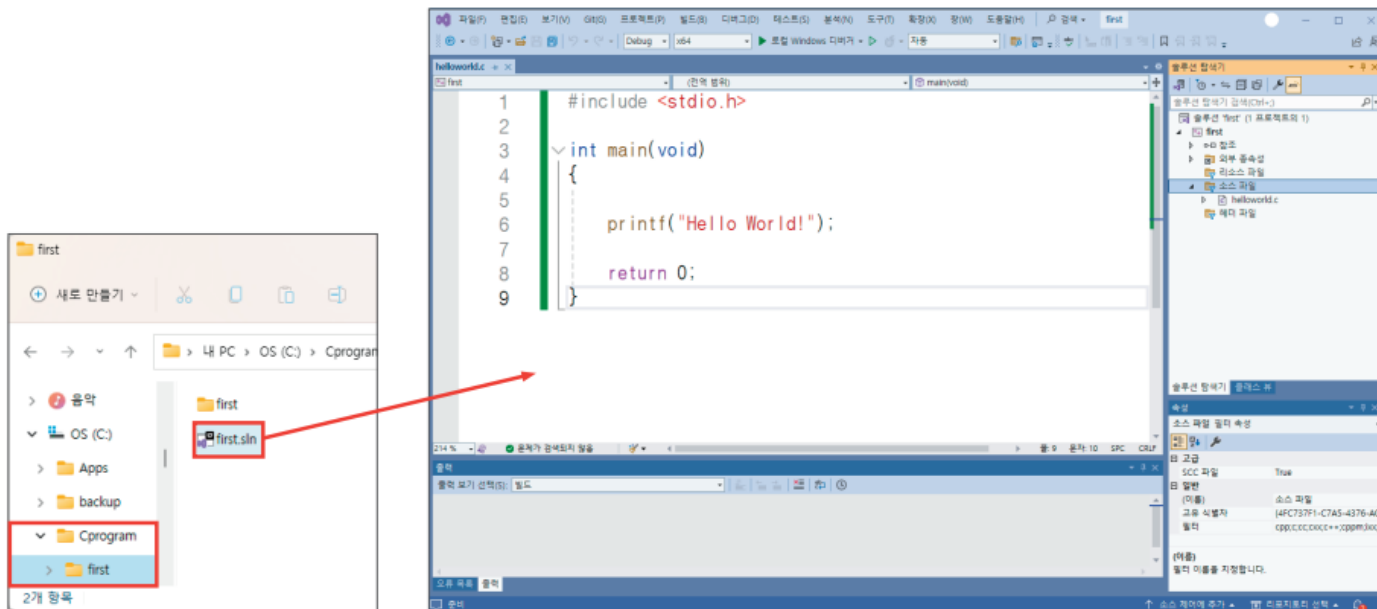


4. C 프로그램 개발 환경 구축

■ 프로젝트 불러오기

■ 솔루션 선택하기

- 프로젝트를 저장한 C:\Cprogram\first 폴더에 저장된 first.sln 파일 더블클릭
- 기존의 프로젝트와 코드 확인 가능



4. C 프로그램 개발 환경 구축

■ 프로젝트 불러오기

■ 비주얼 스튜디오 2022로 불러오기

- 왼쪽 목록의 최근에 작성한 프로젝트에서 원하는 솔루션 선택
- 없다면 [프로젝트 또는 솔루션 열기] 선택 후 C:\WCprogram\first 폴더에 저장된 first.sln 파일 선택

