

2강

C프로그래밍 첫걸음

동양미래대학교 강환수 교수

본 강의 사용 및 참조 자료

▶ Perfect C, 3판, 강환수 외 2인 공저, 인피니티북스, 2021



2장 C 프로그래밍 첫걸음



목차

- 1 프로그래밍 절차
- 2 첫 프로그램 작성과 실행
- 3 오류 메시지와 오류 수정



01

프로그래밍 절차

프로그램 구현 과정

➤ SW 개발 5단계

- 요구 분석
- 설계
- 구현
- 검증
- 유지보수

➤ SW 개발 과정 연구 분야

- 소프트웨어 공학



C 프로그램 구현 과정

1 프로그램 구상

2 소스 편집

3 컴파일

4 링크

5 실행

▶ 3, 4, 5는 IDE에서 한번에 가능



➤ 소스파일(source file) 또는 소스코드(source code)

- 프로그래밍 언어로 일련의 명령어가 저장된 파일
- 텍스트 파일

➤ 소스 파일 확장자

- C 언어
 - *.C
- 자바
 - *.java
- C++
 - *.cpp



▶ 컴파일러(compiler)

- 통역사가 한국어를 영어로 번역하듯이
- 고급언어에서 기계어를 생성하는 프로그램
 - 소스파일에서 기계어로 작성된 목적파일(object file)을 생성
- 소스파일
 - main.c, submoudule1.c, submoudule2.c
- 3개의 목적파일이 생성
 - main.obj, submoudule1.obj, submoudule2.obj



➤ 링크(link) 또는 링킹(linking)

■ 링커(linker)가 수행하는 과정

- 여러 개의 목적 파일을 연결하여 하나의 실행 파일(execute file)을 생성해 주는 과정
- 참조하는 여러 라이브러리를 포함시킴
- 링크의 결과인 실행 파일의 확장자는 .exe 또는 .dll, .com 등

➤ 비주얼 스튜디오, 빌드(build)를 제공

■ 컴파일과 링크 과정을 하나로 합친 메뉴

- 성공하면 파일 확장자가 .exe인 하나의 실행 파일이 생성



➤ 다음 파일을 연결

- 프로젝트 소스파일의 목적파일
- 사용한 라이브러리 목적파일

➤ 실행 파일을 만들어 내는 과정

- 실행 파일 이름
 - 프로젝트 이름



소프트웨어에서 라이브러리란?

▶ 리포트 작성 시

- 인터넷이나 자신의 지식이 부족하면 도서관의 자료를 찾듯

▶ 다른 전문가가 자주 사용하는 기능의 단위 프로그램을 이미 만들어 놓은 모듈(module)

- 개발자마다 새로 작성할 필요 없이 이미 만들어 놓은 라이브러리를 사용
- 단위 프로그램
 - C: 함수 중심
 - Java: 클래스 중심



오류: 발생 시점에 따른 구분

오류 또는 에러(error)

- 프로그램 개발 과정에서 나타나는 모든 문제

발생 시점에 따른 구분

오류 종류	내 용
컴파일(시간) 오류	<ul style="list-style-type: none"> • 개발환경에서 오류 내용과 오류 발생 위치를 어느 정도 알려줌 • 오류를 찾아 수정하기가 비교적 용이
링크(시간) 오류	<ul style="list-style-type: none"> • 컴파일 오류보다 상대적으로 희소 • main() 함수 이름이나 라이브러리 함수 이름을 잘못 기술하여 발생
실행(시간) 오류	<ul style="list-style-type: none"> • 실행하면서 오류가 발생해 실행이 중지되는 경우



오류: 원인과 성격에 따른 구분

오류의 원인과 성격에 따른 구분

■ 구문 오류(syntax error) 또는 문법 오류

- 프로그래밍 언어 문법(syntax)을 잘못 기술
- 문법이 잘못된 소스로 발생하는 오류

■ 논리 오류(logic error)

- 내부 알고리즘이 잘못되거나 원하는 결과가 나오지 않는 등의 오류
- 원의 면적을 구하는데 ‘ $2 \times 3.14(\text{원주율}) \times \text{반지름}$ ’으로 잘못 계산
- 해당 년도의 평균평점을 계산했는데 잘못된 결과가 나오는 등



▶ 디버깅(debugging)

- 다양한 오류를 찾아 소스를 수정하여 다시 컴파일, 링크, 실행하는 과정
- 디버거(debugger)
 - 디버깅을 도와주는 프로그램
 - 오류: 버그



➤ 1947년 9월 9일, 미국 해군 호퍼 중위

■ 컴퓨터가 갑자기 작동 멈춤

- 컴퓨터 패널의 릴레이를 살핌
- 바로 여기에 나방이 끼어 있었던 것

9/9

0800 Antan started
1000 " stopped - antan ✓ { 1.2700 9.037 847 025
13°C (032) MP-MC 2.130476415 (03) 4.615925059(-2)
(033) PRO 2 2.130476415
convd 2.130676415
Relays 6-2 in 033 failed special speed test
in relay " 10,000 test -

1100 Started Cosine Tape (Sine check)
1525 Started Multi-Adder Test.

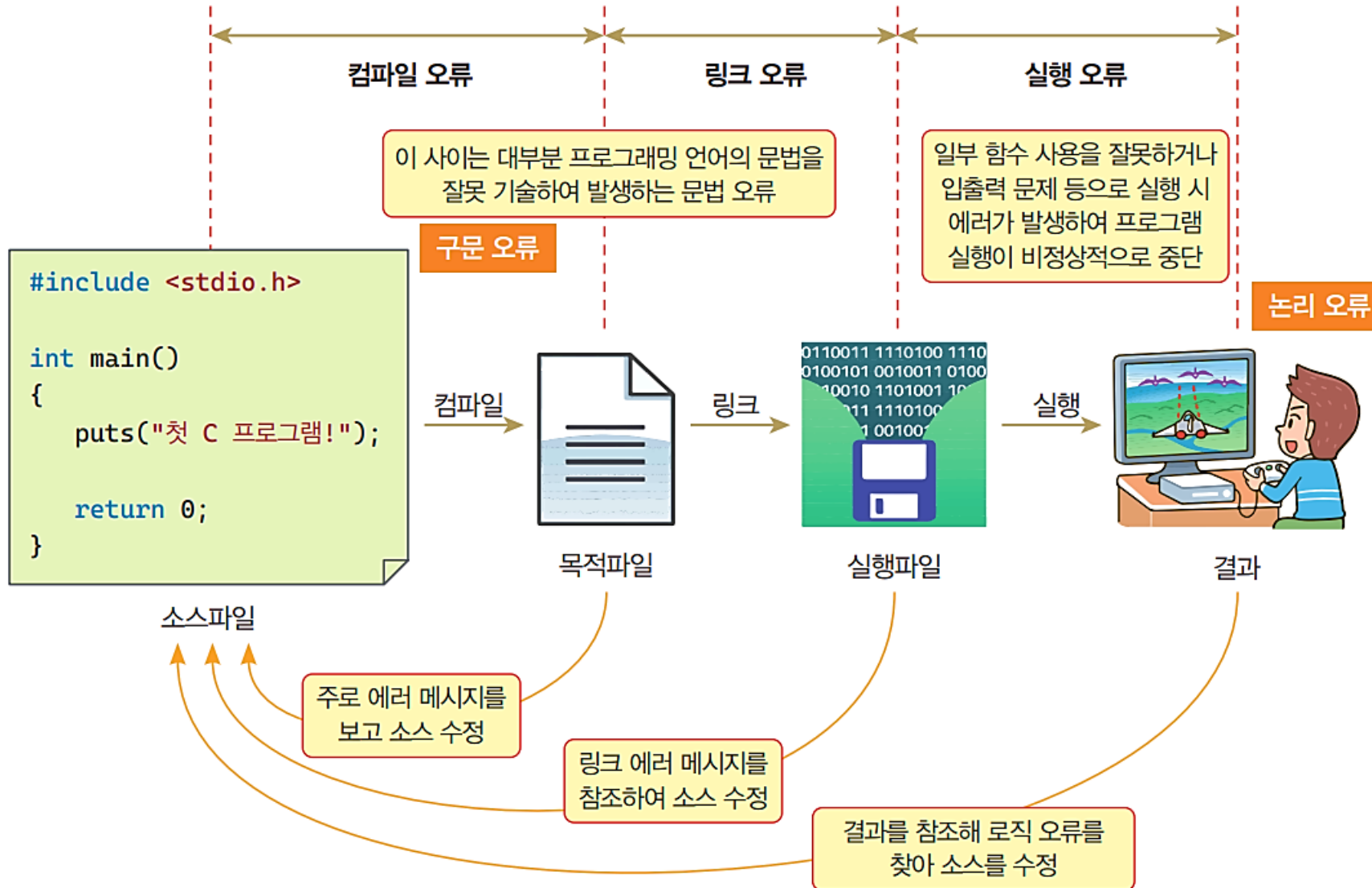
1545 Relay #70 Panel F
(moth) in relay.

First actual case of bug being found.

1630 Antan started.
1700 closed down.

Relay 3345
Relay 3370





프로그램 구현 과정

- ▶ 컴파일과 링크, 실행 시 오류가 발생
 - 대부분 소스 코드를 수정 후
 - 다시 컴파일, 링크, 실행
 - 즉 소스 코드를 수정한다면 반드시 다음을 다시 수행
 - 컴파일 → 링크 → 실행



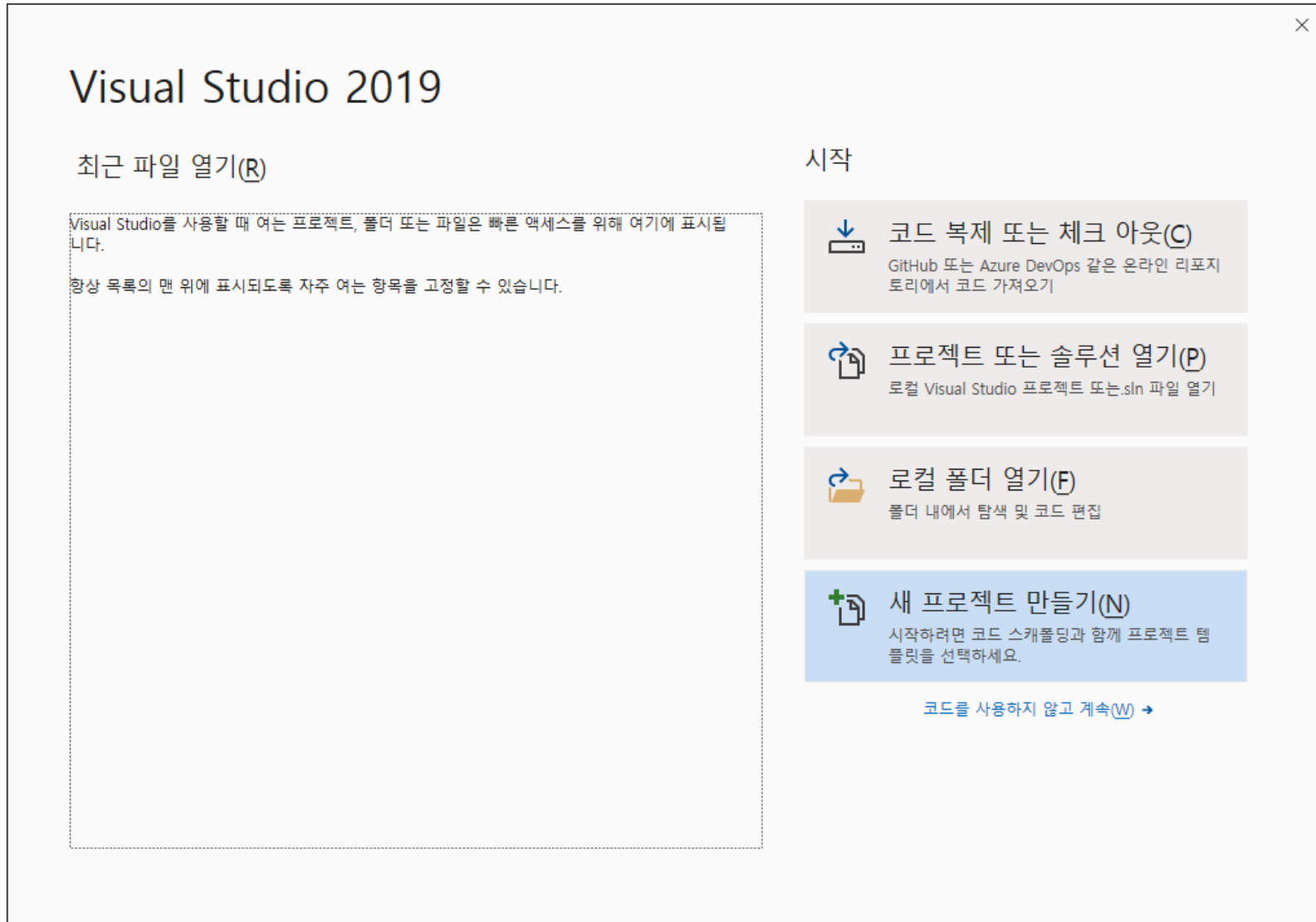
02

첫 프로그램 작성과 실행

- 로고 화면 이후 표시된 화면
 - [새 프로젝트 만들기(N)]
 - [빈 프로젝트]를 선택



[새 프로젝트 만들기]



[빈 프로젝트]

첫 프로그램 작성과 실행



➤ 솔루션

- 여러 개의 프로젝트를 모아 놓은 폴더
- 단원 이름
 - ch02



- 임의로 지정된 프로젝트 이름과 위치, 그리고 솔루션 이름 지정
- 다음 순서로 입력
 - 프로젝트 이름
 - P01-HelloWorld
 - 위치:
 - 모든 솔루션(단원)을 저장할 최상위 폴더 지정
 - 솔루션 이름: 단원 이름으로
 - ch02



▶ 주 화면 오른쪽 '솔루션 탐색기'

■ 생성된 솔루션과 프로젝트

- 솔루션의 프로젝트와 파일을 쉽게 관리



솔루션 탐색기 1/2

솔루션 탐색기에서는 프로젝트와 파일을 체계적으로 보여주며, 관련된 메뉴를 활용할 수 있다. 프로젝트는 소스코드를 작성하기 위한 기본 환경으로 하부에 리소스파일, 소스파일, 헤더 파일 등이 있다.

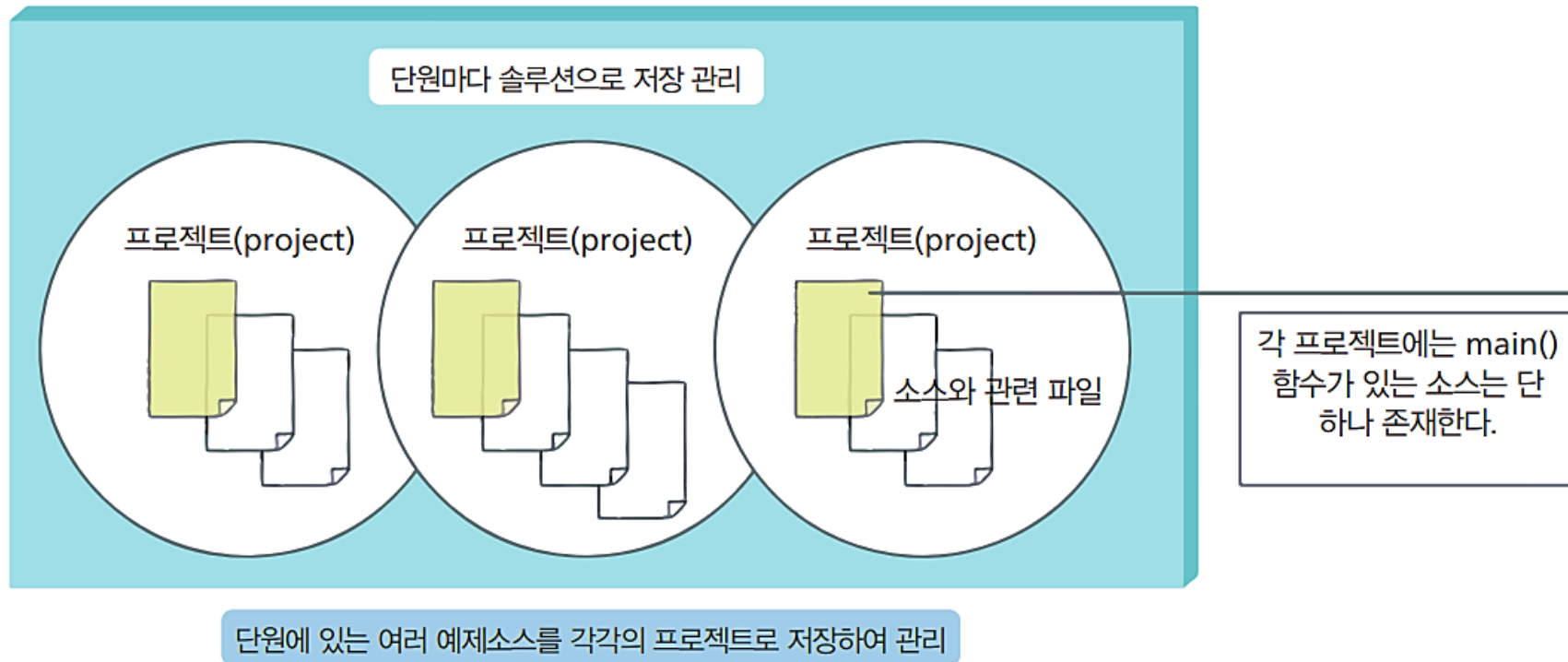
속성에서는 솔루션 탐색기에서 선택한 프로젝트 [P01- HelloWorld]에 대한 속성 정보가 표시된다.

P01-HelloWorld 프로젝트 속성	
기타	
(이름)	P01-HelloWorld
루트 네임스페이스	P01HelloWorld
프로젝트 종속성	
프로젝트 파일	D:\WC code\ch02\WP01-HelloWc
(이름)	
프로젝트 이름을 지정합니다.	



➤ 솔루션: 단위

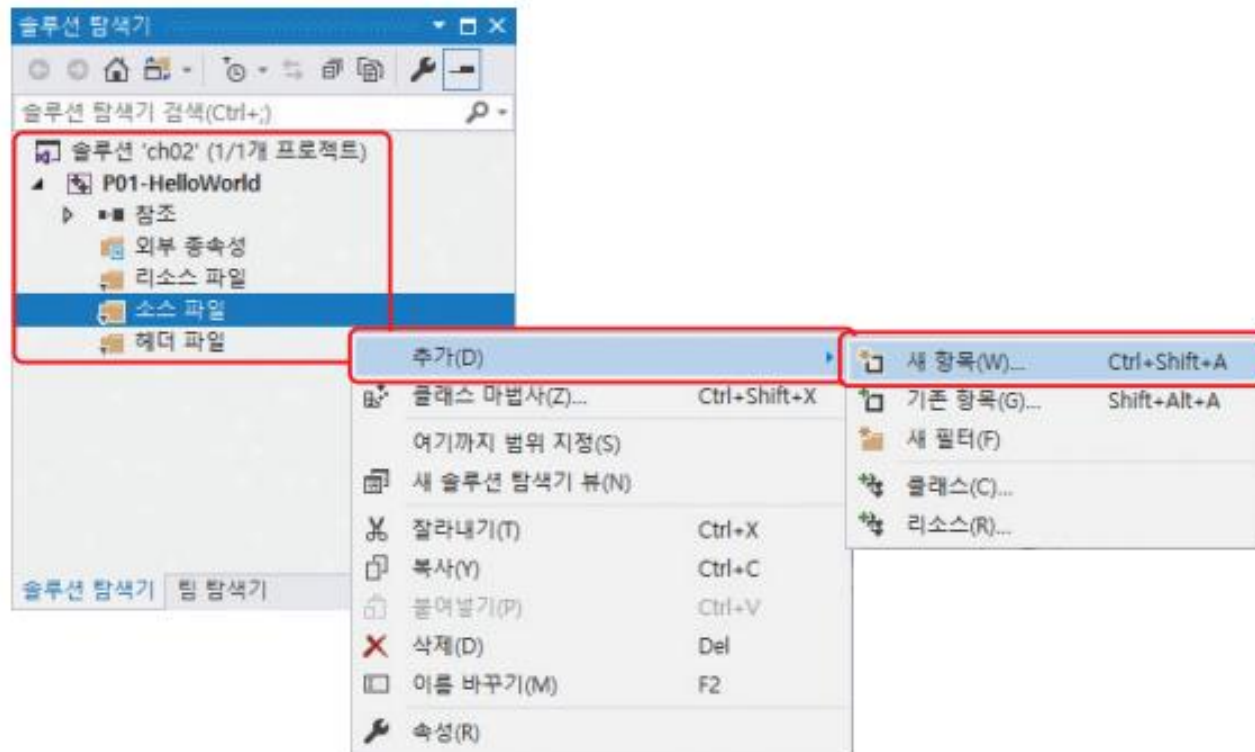
■ 프로젝트: 각각의 예제



프로젝트에서 소스 작성 1/2

▶ 메뉴 [프로젝트/새 항목 추가]

- 또는 '솔루션 탐색기'의 '소스파일' 폴더에서
메뉴 [추가] / [새 항목]을 선택



▶ 대화상자 [새 항목 추가 P01-HelloWrold]

- 각각 'Visual C++'와 'C++ 파일(.cpp)'을 선택
 - '이름': 소스파일 이름 01hello.c 입력
 - '위치': '솔루션 폴더/프로젝트 폴더'인 'ch02/P01-HelloWrold'를 확인



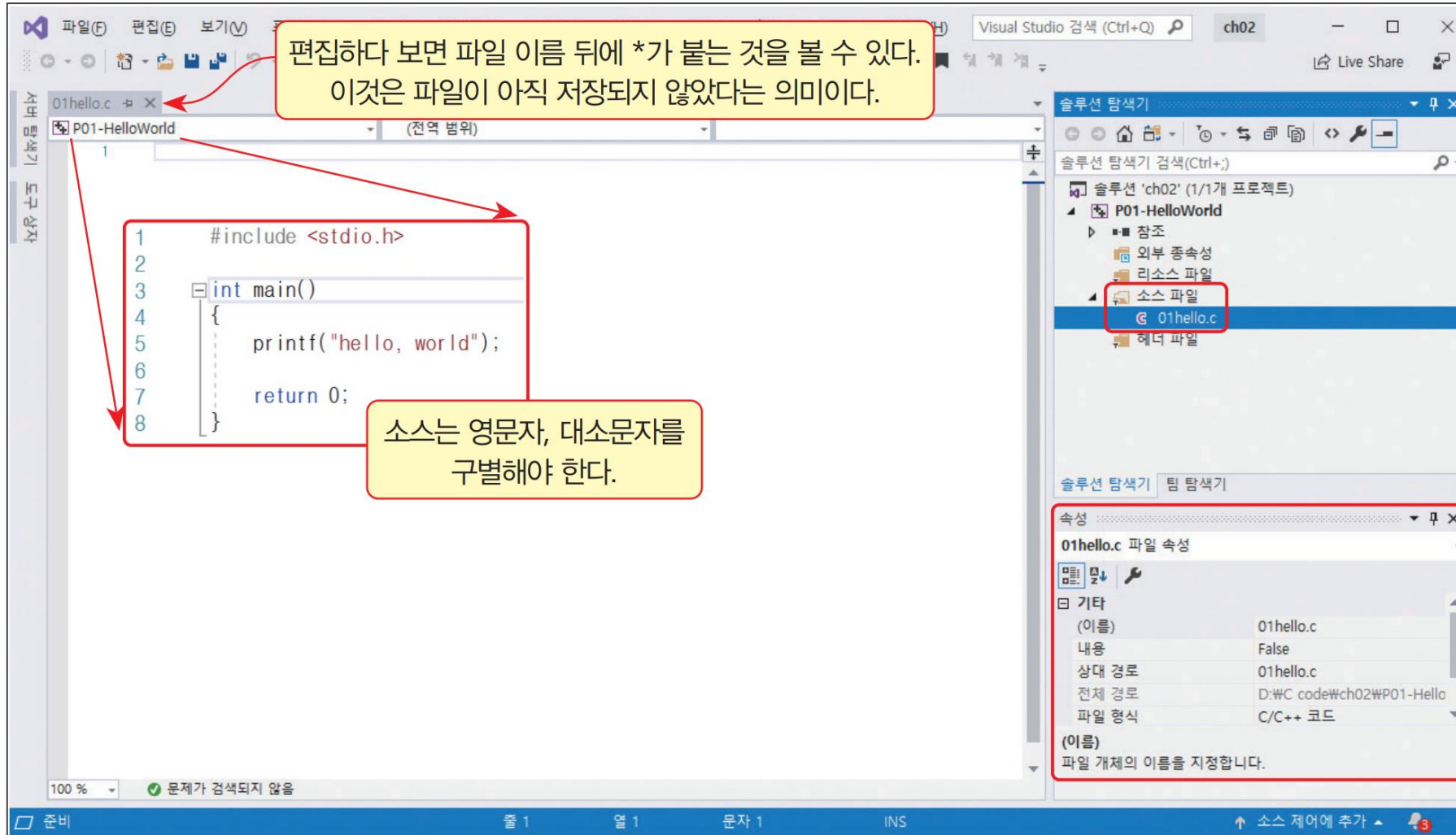
소스파일 01hello.c 생성 결과

- 솔루션 탐색기의 '소스파일' 폴더 하단
 - 파일 01hello.c가 표시
- 왼쪽에는 소스를 편집할 수 있는 창
- 오른쪽은 솔루션 탐색기
 - 각 소스파일을 선택
 - 하단 속성 탭에서 선택된 파일의 자세한 정보 표시



소스파일이 생성된 화면 모습

첫 프로그램 작성과 실행



▶ 하나의 솔루션

- 하나 이상의 프로젝트를 저장
- 하나의 프로젝트는 여러 개의 프로그램 소스
 - 하나의 main() 함수 만을 구성



소스 작성의 주의점과 들여쓰기의 이해

▶ 세미콜론 ;

- 문장의 종료를 표시
- 콜론 :으로 잘못 입력하면 컴파일 오류 발생

▶ 들여쓰기(indentation)

- IDE의 편집기에서 자동으로 맞추어 줌
- 소스의 가독성(readability)을 위해 반드시 필요



코딩 시 주의점

함수 main() 위의 #include는 일반 C 문장이 아니며, 전처리 부분이라고 부르고 컴파일 전에 처리된다.

비주얼 스튜디오에서는 int와 main()처럼 글자의 성격에 따라 색상으로 구분해 준다. 프로그래머가 실수를 줄여주기 위해 제공하는 기능으로 잘 활용하면 유용하다.

함수 main()의 구현은 탭(공간 3~4개)을 누른 후 입력하는데 이를 들여쓰기라 하며, 편집기에서 자동으로 지원한다. C 소스에서 들여쓰기는 소스를 읽기 쉽게 하기 위한 방법으로 잘못하더라도 문법적인 문제는 발생하지 않는다.

문자의 모임인 문자열은 반드시 큰따옴표로 묶어야 한다.

이 사이에는 반드시 공간(space)이 필요하며 일반적으로 하나의 빈 문자를 입력한다.

일반 문장의 마지막은 세미콜론 ; 으로 입력한다. 그러나 첫 줄의 #include 의 끝에는 ;이 없어야 한다. 일반 C 언어 문장이 아니기 때문이다.

```
01 # i n c l u d e < s t d i o . h >
02
03 i n t   m a i n ( )
04 {
05     p r i n t f ( " h e l l o ,   w o r l d " ) ;
06     ↔
07     r e t u r n   0 ;
08 }
```



프로젝트 실행 ctrl + F5

첫 프로그램 작성과 실행

❶ 작성된 소스에는 문제가 없어야 실행에 성공

❷ [디버그]

❸ [디버깅하지 않고 시작]을 선택

❹ ‘출력’에 빌드 과정이 표시

- 마지막 줄에 성공 1, 실패 0와 같이 표시

❺ 검정색 바탕의 콘솔 화면이 표시



문자열 "hello, world"가 콘솔 창에 출력되는 프로그램

- printf()의 괄호 사이에 기술된 "hello, world"가 출력
 - 대소문자를 구별
 - include, stdio.h, int, main, printf, return
 - 특별한 의미의 여러 문자들로 구성
 - #, <, >, (,), :, {, }



실습예제

첫 프로그램 작성과 실행

P01-HelloWorld

01hello.c

콘솔에 hello, world 출력

난이도: ★

01 `#include <stdio.h>`

02

03 `int main()`

04 `{`

05 `printf("hello, world");`

06

07 `return 0;`

08 `}`

printf() 함수가 콘솔에 출력할 문자열을
큰 따옴표로 묶어야 한다.

문장을 종료할 때는 ; (세미콜론)을
입력해야 한다.

5 줄에서 시작해서 6 줄을 지나, 7 줄 return 0로
0을 반환하고 프로그램이 종료된다.

hello, world



03

오류 메시지와 오류 수정

구문 오류 메시지의 이해와 소스 수정

오류 내용

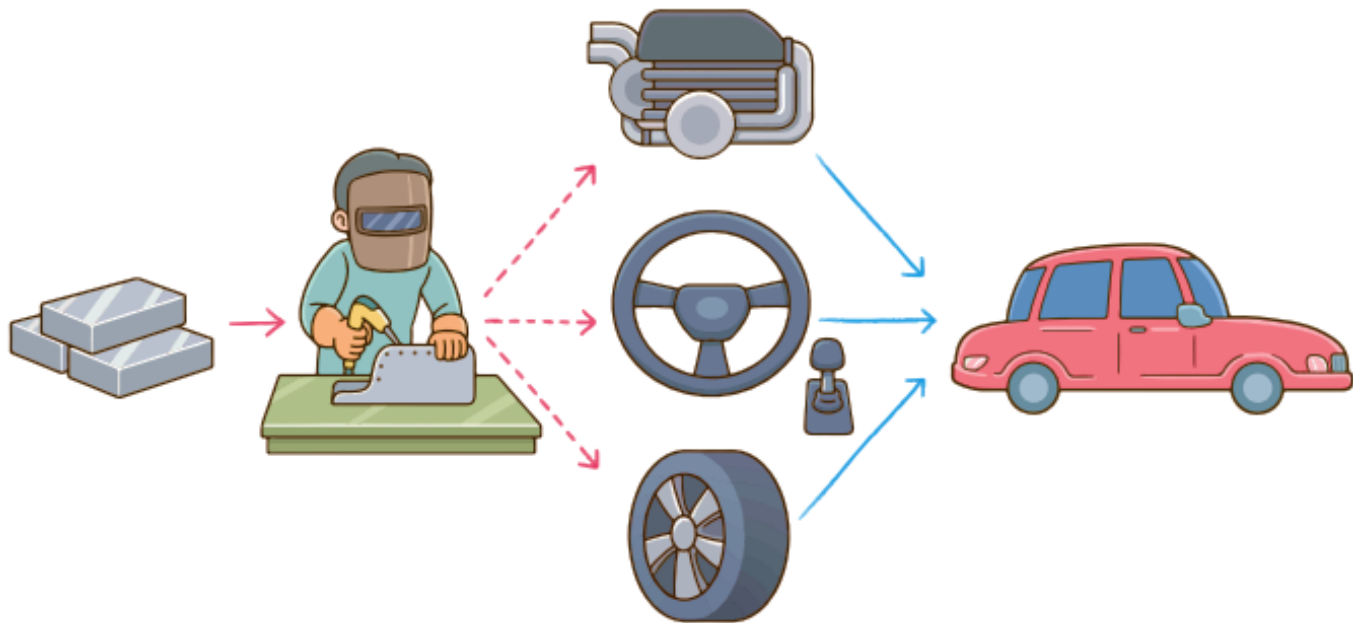
- 오류발생 파일의 경로를 포함한 전체이름
- 추정되는 오류발생 줄과 열 번호
- 오류 코드 번호
- 오류 원인 메시지 등

```
05 printf("컴퓨터공학과 학생이며\n")  
06 printf("C 프로그래밍 언어를 수강합니다.\n");
```

문장 마지막에 ;가 빠져 발생하는
오류는 다음과 같은 메시지가 표시됨.

... \debugging.c(6,2): error C2146: 구문 오류: ';'이(가) 'printf' 식별자 앞에 없습니다.

- ▶ 부품을 조립하는 링크 과정에서 발생하는 오류가 '링크 오류'
- ▶ 대표적인 링크 오류
 - 라이브러리 함수인 `printf()`의 철자를 잘못 기술하는 경우
 - 마지막 f를 빼고 `print()`로 기술



▶ 논리 오류 (logical error) 예

- 문장을 출력
 - 원하는 문장이 아닌 경우
- 교과목의 성적을 산출하는 프로그램
 - 잘못된 성적 결과



정 리 하 기



정리하기

- 프로그램 구현 과정을 이해하고 소스 작성과 컴파일, 링크, 실행 절차를 수행해 본다.
- 오류의 종류를 이해하고 수정하는 절차를 수행해 본다.
- 비주얼 스튜디오에서 C 프로그램을 작성하기 위해 프로젝트를 생성해 본다.
- 비주얼 스튜디오에서 솔루션 하부에 여러 프로젝트가 있으며, 프로젝트는 하나의 `main()` 함수를 갖는 여러 소스파일로 구성된다.
- 비주얼 스튜디오에서 작성된 C 프로그램을 실행해 본다.
- 오류 발생시 오류의 의미를 이해하고 잘못된 부분을 수정해 본다.

3강

다음시간안내

프로그래밍 기초

