




누구나 즐기는 C언어 콘서트

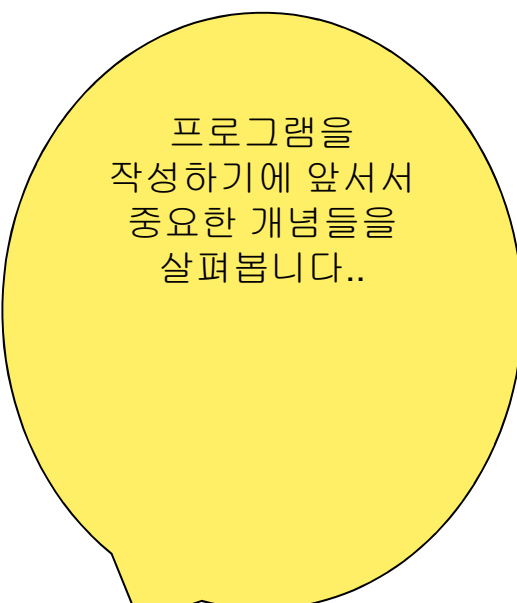
제1장 프로그래밍 소개





이번 장에서 학습할 내용

- 
- 프로그램의 이해
 - 프로그래밍 언어
 - 알고리즘
 - 프로그램 개발 과정



프로그램을
작성하기에 앞서서
중요한 개념들을
살펴봅니다..





컴퓨터란?

Q) 컴퓨터(computer)는 무엇인가?

A) 컴퓨터는 기본적으로 계산(compute)하는 기계

Q) 컴퓨터를 이용하여 데이터를 처리하려면 반드시 데이터가 숫자 형태이어야 한다. 왜?

A) 컴퓨터는 숫자 계산을 하기 때문에 데이터는 숫자로 표시되어야 한다.



각 화소의 밝
기와 색상을
숫자로 표현
한다.

디지털 이미지의 경우



각 파형의 높
이를 숫자로
표현한다.

디지털 음악의 경우



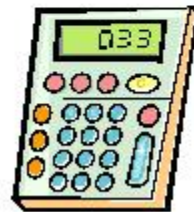
컴퓨터의 정의

Q) 그렇다면 계산만 빠르게 할 수 있으면 컴퓨터인가?

A) 현대적인 의미에서의 컴퓨터는 명령어들의 리스트에 따라 데이터를 처리하는 기계라고 할 수 있다



계산기는 정해진
기능만을 수행한
다. 기능을 변경
할 수 없다.



프로그램이라는
개념을 도입하여
수행하는 기능을
쉽게 변경할 수
있다.





컴퓨터의 장점

Q) 컴퓨터의 가장 큰 장점은 무엇일까?

A) 컴퓨터는 범용적인 기계이다. 프로그램만 바꿔주면 다양한 작업이 가능하다.

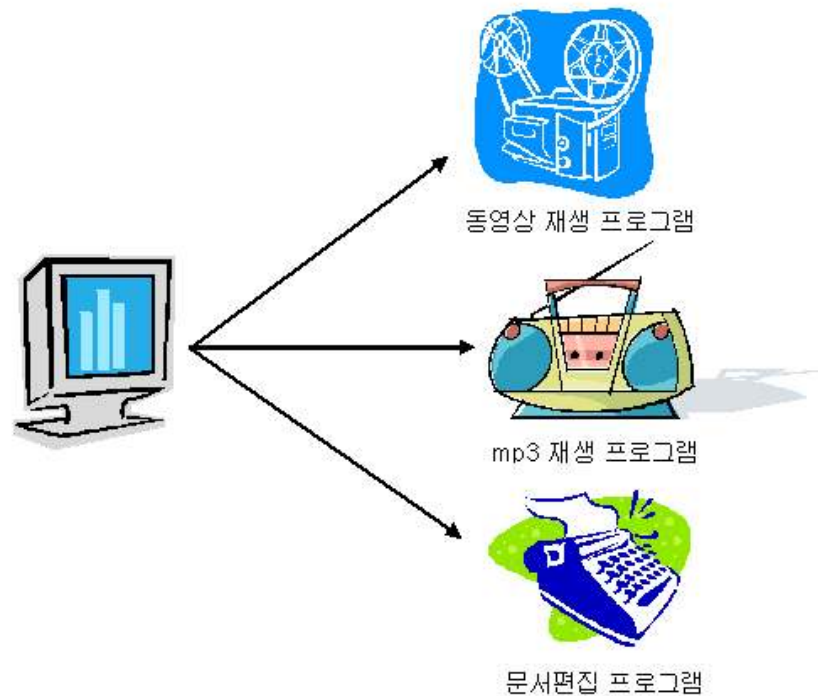


그림 1.1 프로그램만 바꾸어주면 컴퓨터는 다양한 작업을 할 수 있다.



스마트폰도 컴퓨터의 일종

- 피쳐폰과 다르게 애플리케이션만 변경하면 다양한 용도로 사용 가능





컴퓨터의 구성 요소

Q) 컴퓨터의 구성 요소를 크게 2가지로 분류하면?

A) 컴퓨터는 기본적으로 하드웨어와 소프트웨어로 구분



하드웨어



소프트웨어

그림 1.2 하드웨어와 소프트웨어



하드웨어와 소프트웨어의 분리

- 최초의 컴퓨터에서는 분리되지 않았음



- 1950년대에 폰노이만이 제시한 아이디어
 - 프로그램을 메모리에 저장
 - 메모리에서 프로그램의 문장을 꺼내와서 하나씩 실행



프로그램의 역할

Q) 컴퓨터에서 프로그램이 하는 일은 무엇인가?

A) 프로그램이란 우리가 하고자 하는 작업을 컴퓨터에게 전달하여 주는 역할을 한다.

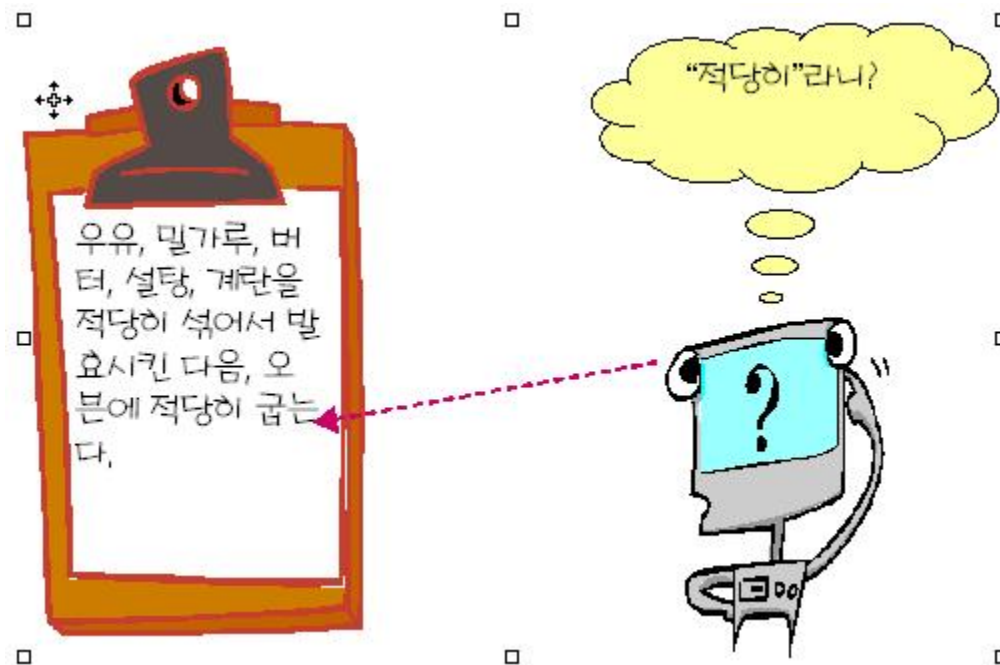




작업을 지시하는 방법

Q) 컴퓨터에게 적당히 작업을 시킬 수 있을까?

A) 상식이나 지능이 없기 때문에 아주 자세하고 구체적으로 일을 지시하여야 한다.





프로그램 안에 들어있는 것

- 컴퓨터에게 무엇을 어떻게 시킬 지를 기록해놓은 문서가 프로그램(program)이다.
- 프로그램의 각 문장은 컴퓨터에게 작업을 지시하는 명령(instruction)으로 되어 있다.

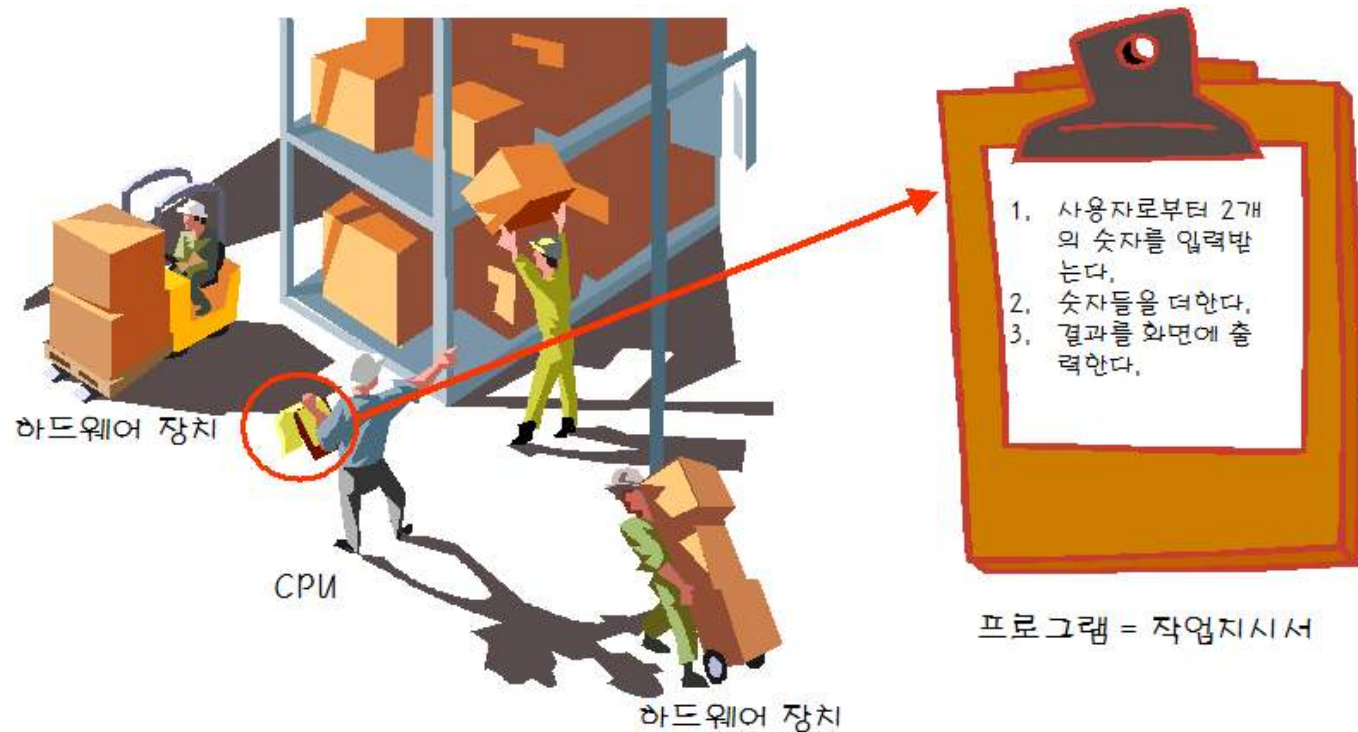

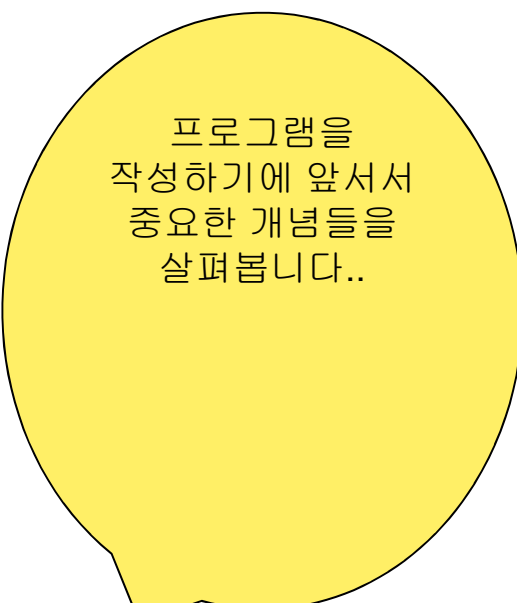


그림 1.4 프로그램은 작업 지시서와 같다.



이번 장에서 학습할 내용

- 
- 프로그램의 이해
 - 프로그래밍 언어
 - 알고리즘
 - 프로그램 개발 과정



프로그램을
작성하기에 앞서서
중요한 개념들을
살펴봅니다..





컴퓨터가 이해하는 언어

- 컴퓨터는 인간의 언어를 이해할 수 없다.
- 컴퓨터는 이진수로 된 기계어만을 이해한다.

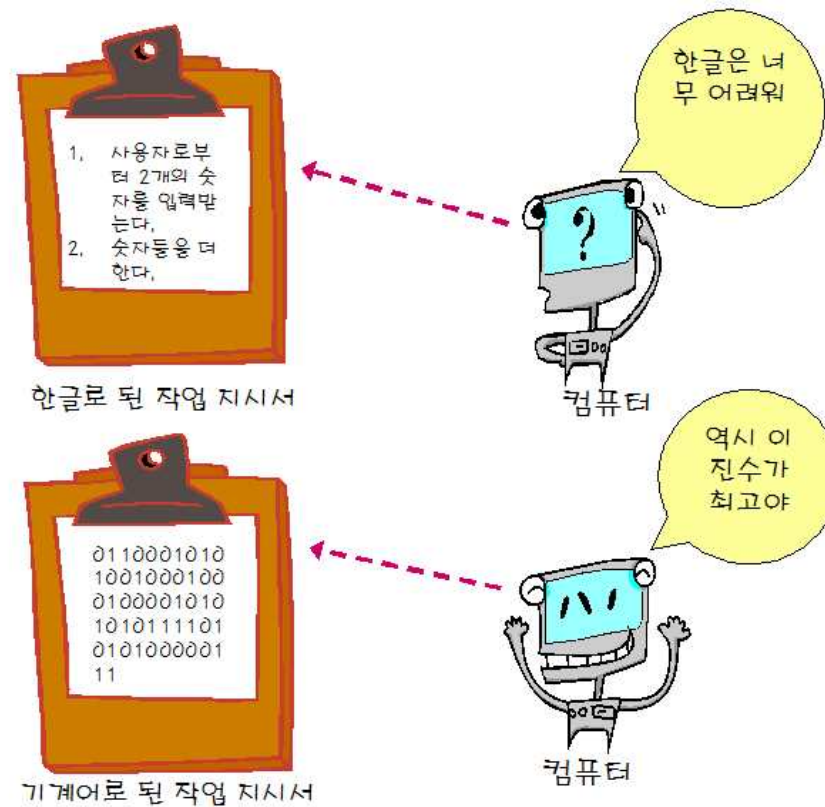


그림 1.5 컴퓨터는 한글로 된 작업 지시서는 이해하지 못하는 반면 기계어로 된 작업 지시서는 이해할 수 있다.



프로그래밍 언어의 역할

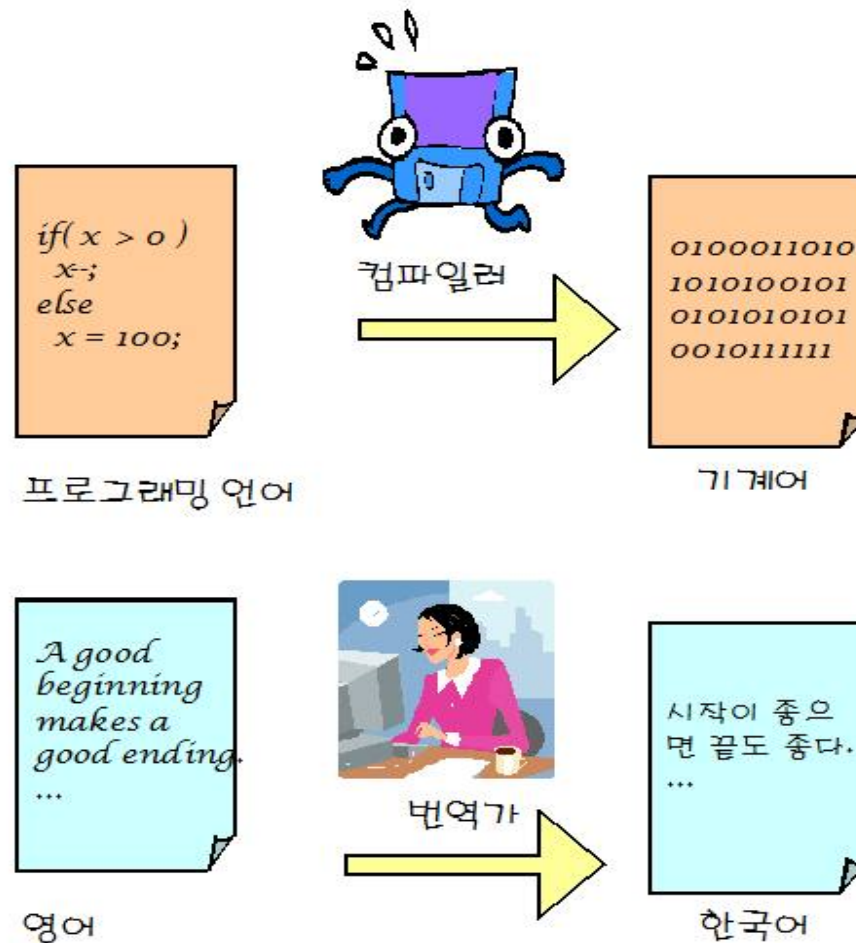
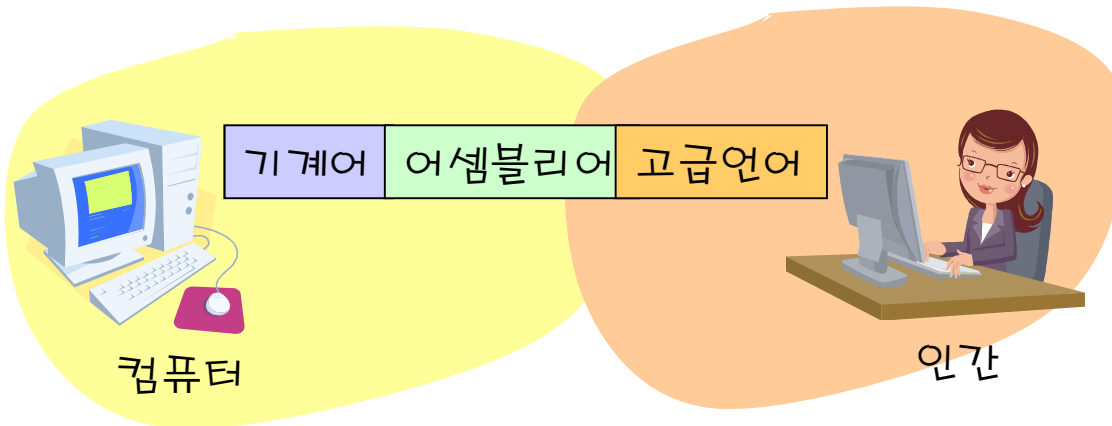


그림 1.6 컴파일러는 프로그램을 기계어로 변환한다.



프로그래밍 언어의 분류

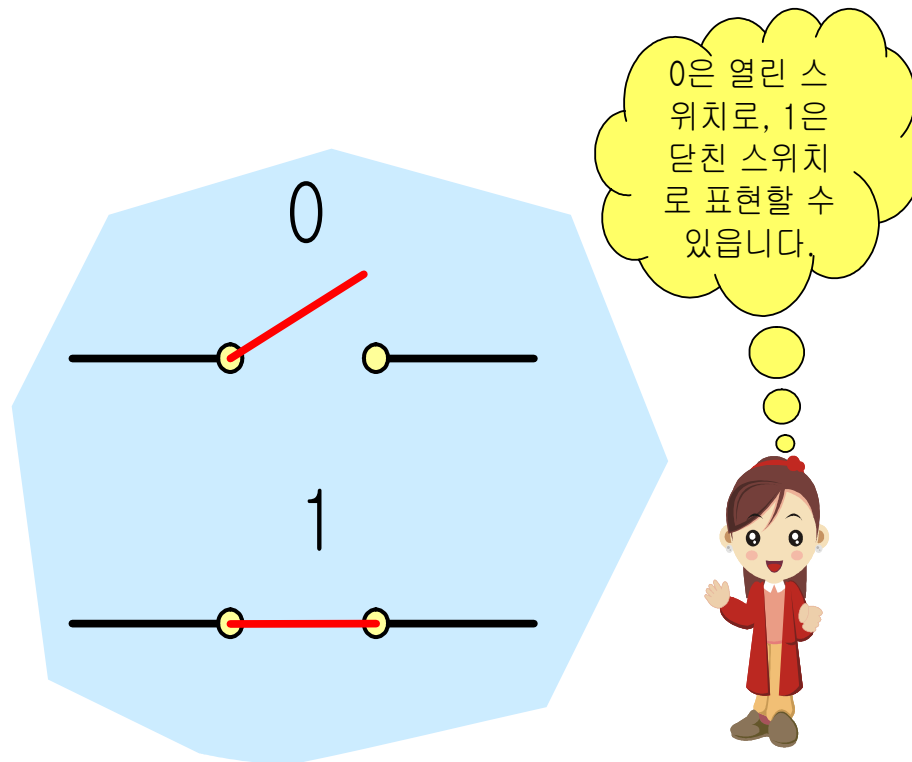
- 기계어(machine language)
- 어셈블리어(assembly language)
- 고급 언어(high-level language)





컴퓨터에 이진수가 사용되는 이유

- 이진수는 전자회로로 구현하기가 쉽다.





C

- 1970년대 초 AT&T의 Dennis Ritchie 에 의하여 개발
- B언어->C언어
- UNIX 운영 체제 개발에 필요해서 만들어짐
- 처음부터 전문가용 언어로 출발



Ken Thomson과 Dennis Ritchie가 클린턴 대통령으로부터 National Medal of Technology 상을 받는 장면



C언어의 버전

- K & R C
 - 1978년 “C Programming Language” 책 출간
 - 비공식적인 명세서 역할
- ANSI C
 - 1983년 ANSI(American National Standards Institute)는 X3J11이라는 위원회에 의한 표준
- C99
 - 1999년에 ISO에 의한 표준
 - C++에서 사용되는 특징 추가
 - 아직 마이크로소프트는 지원하지 않음(이유: C++에 집중)



C언어의 특징

- 간결하다.
- 효율적이다.
- C 언어는 하드웨어를 직접 제어하는 하는 저수준의 프로그래밍도 가능하고 고수준의 프로그래밍도 가능하다.
- C언어는 이식성이 뛰어나다.
- 초보자가 배우기가 어렵다.



C언어의 특징



그림 1.10 C 언어의 특징



C언어의 미래

Q) 앞으로도 C언어는 사용될 것인가?

- C언어는 C++와 JAVA, C#, Objective-C의 공통적인 부분이다.
- 임베디드 시스템에서는 C언어가 많이 사용된다

mp3 플레이어도 CPU와 플래시 메모리 등이 들어가 있는 임베디드 시스템이다.



임베디드 시스템: 임베디드 시스템이란 특수 목적의 시스템으로 컴퓨터가 장치 안에 MP3 플레이어, 스마트폰등이 여기에 속한다.



이번 장에서 학습할 내용



- 프로그램의 이해
- 프로그래밍 언어
- 알고리즘
- 프로그램 개발 과정

프로그램을
작성하기에 앞서서
중요한 개념들을
살펴봅니다..





알고리즘

Q) 오븐의 사용법만 배우고 음식 재료만 있으면 누구나 요리가 가능한가?

A) 요리법을 알아야 한다.

- 프로그램이 요리와 같다면 알고리즘은 요리법에 해당한다.
- **알고리즘(algorithm)**: 문제를 해결하는 절차(방법)





빵을 만드는 알고리즘

- ① 빈 그릇을 준비한다.
- ② 이스트를 밀가루, 우유에 넣고 저어준다.
- ③ 버터, 설탕, 계란을 추가로 넣고 섞는다.
- ④ 따뜻한 곳에 놓아두어 발효시킨다
- ⑤ 170~180도의 오븐에서 굽는다



->



->



->



->





1부터 10까지의 합을 구하는 알고리즘

① 1부터 10까지의 숫자를 직접 하나씩 더한다.

$$I \quad 1 + 2 + 3 + \dots + 10 = 55$$

·② 두수의 합이 10이 되도록 숫자들을 그룹핑하여 그룹의 개수에 10을 곱하고 남은 숫자 5를 더한다.

$$(0 + 10) = 10$$

$$(1 + 9) = 10$$

$$(2 + 8) = 10$$

$$(3 + 7) = 10$$

$$(4 + 6) = 10$$

5

$$10 * 5 = 50 \quad + \quad 5 \quad = \quad 55$$

③ 공식을 이용하여 계산할 수도 있다.

$$10 * (1 + 10) / 2 = 55$$



알고리즘의 기술

- 순서도(flow chart): 프로그램에서의 논리 순서 또는 작업 순서를 그림으로 표현하는 방법

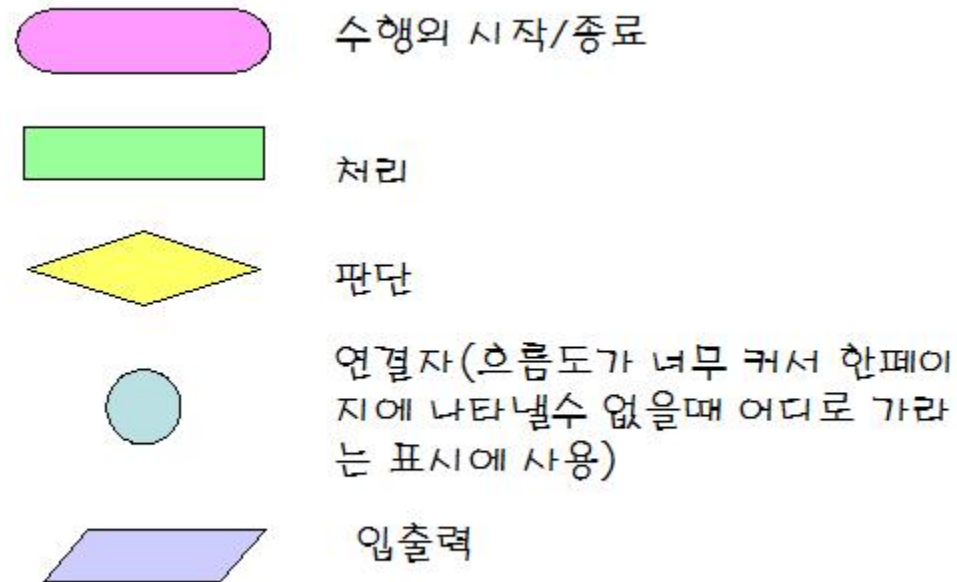


그림 1.13 순서도 작성에 사용되는 기호



알고리즘의 예

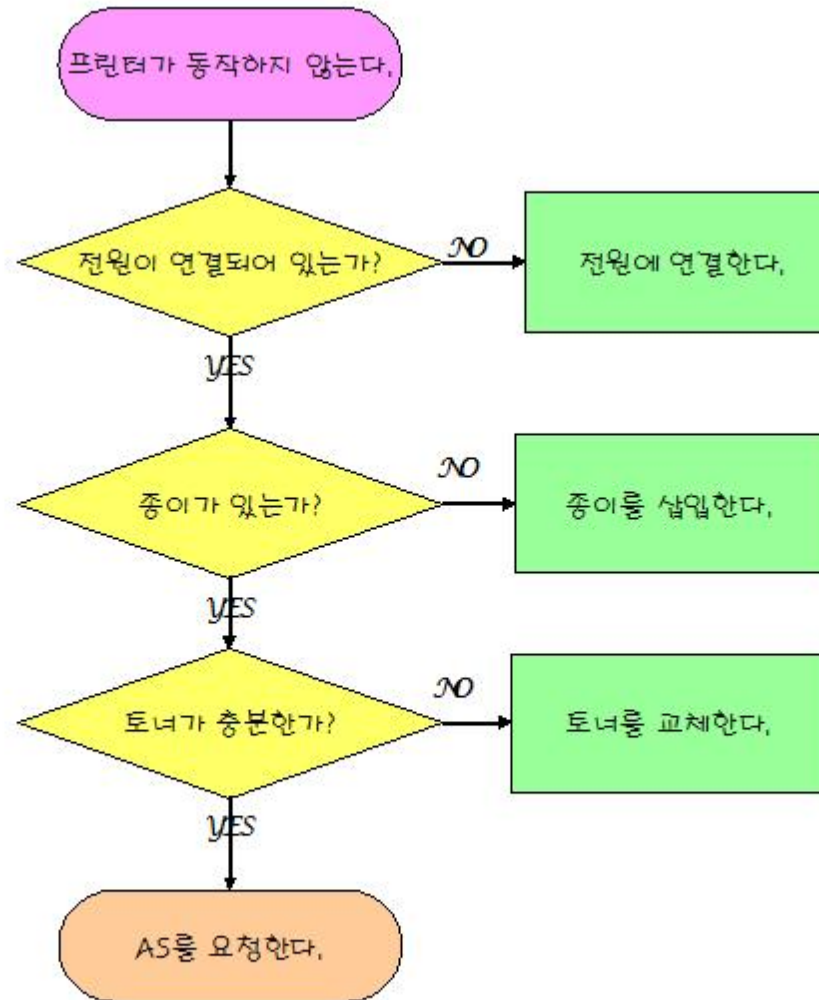


그림 1.14 순서도의 예: 프린터 고장을 처리하는 알고리즘




중간 점검

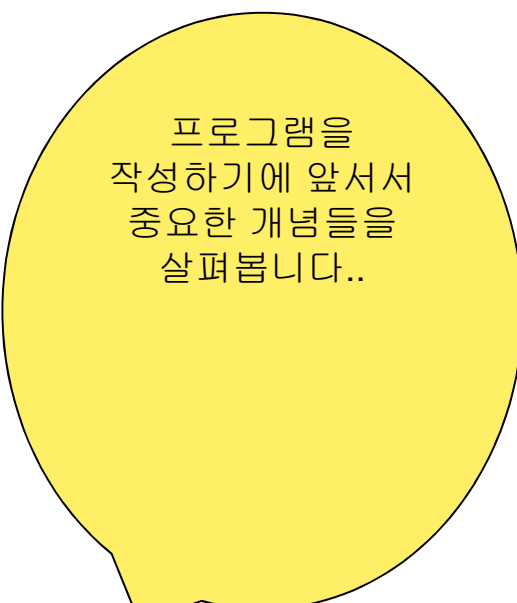
1. 친구에게 전화를 거는 알고리즘을 만들어보라.
2. 세탁기를 이용하여서 세탁을 하는 알고리즘을 만들어보라.





이번 장에서 학습할 내용

- 
- 프로그램의 이해
 - 프로그래밍 언어
 - 알고리즘
 - 프로그램 개발 과정



프로그램을
작성하기에 앞서서
중요한 개념들을
살펴봅니다..





프로그램 작성 과정

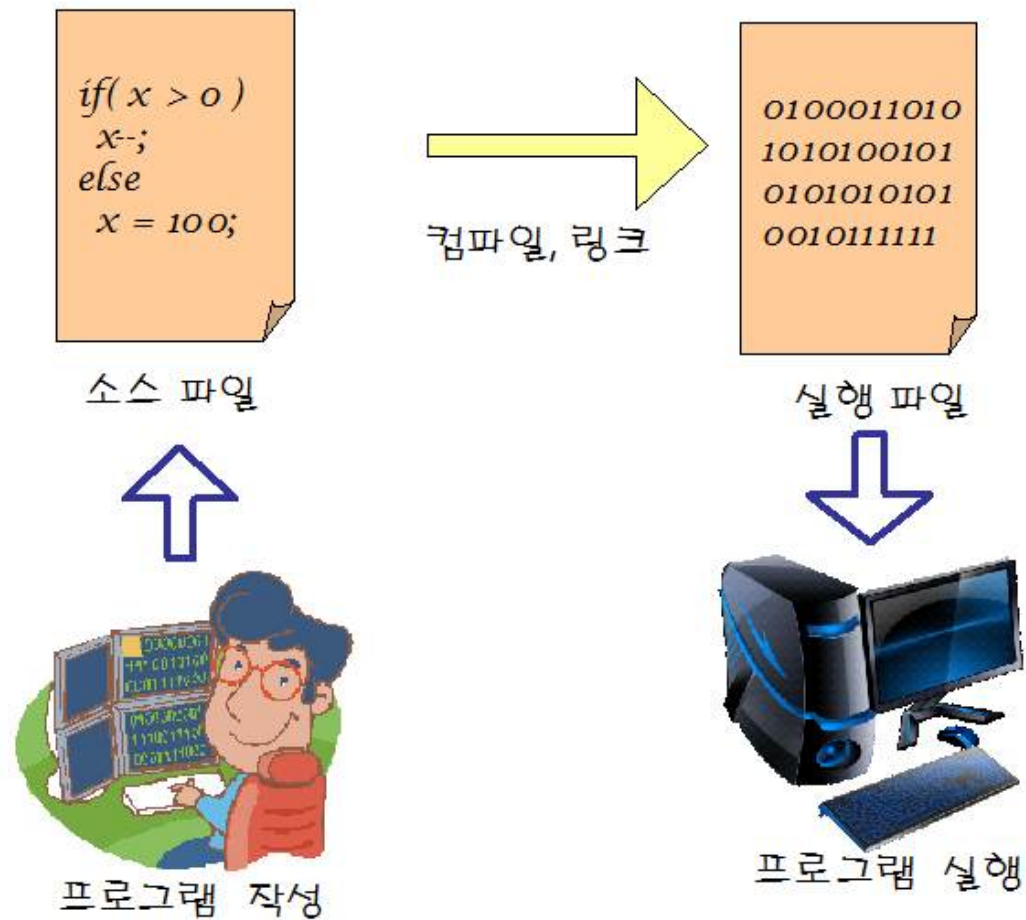
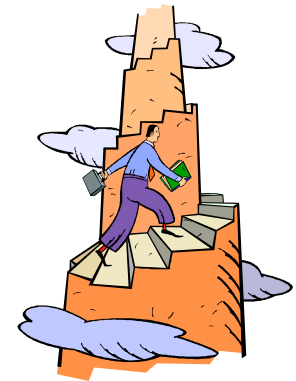


그림 1.15 프로그래밍의 순서



프로그램 작성 단계

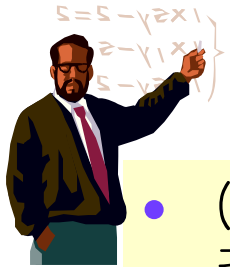
- 편집 (*edit*)
 - 에디터를 이용하여 원하는 작업의 내용을 기술하여 소스 코드 작성
 - 소스 파일(**source file**): 소스 코드가 들어 있는 텍스트 파일
 - (예) `test.c`
- 컴파일 (*compile*)
 - 소스 파일 → 기계어로 변환
 - 오브젝트 파일(**object file**): 기계어로 변환된 파일
 - (예) `test.obj`
- 링크(*link*)
 - 오브젝트 파일들을 라이브러리 파일들과 연결하여 하나의 실행 파일 생성
 - 실행 파일 (**executable file**): 실행이 가능한 파일
 - (예) `test.exe`





Q & A

- (Q)소스 파일과 오브젝트 파일, 실행 파일 중에서 반드시 보관하여야 하는 파일은 무엇일까?



- (A) 정답은 **소스 파일이다**. 소스 파일만 있으면 컴파일러를 수행시켜서 오브젝트 파일, 실행 파일은 만들 수 있다. 하지만 소스 파일을 삭제하면 컴파일이 불가능하다.



통합 개발 환경

- 통합 개발 환경(IDE: integrated development environment):
 - 에디터 + 컴파일러 + 디버거





통합 개발 환경의 종류

- 비주얼 C++(Visual C++)
 - 마이크로소프트사의 제품
 - 윈도우 기반의 거의 모든 형태의 응용 프로그램 제작 가능
 - 우리가 사용할 버전: 비주얼 스튜디오 2008
 - <http://www.microsoft.com/express/vc/>



Visual Studio

MSDN Subscription 천인국 로그아웃

제품 둘러보기 다운로드 시작하기 뉴스 지원

무료 평가판 Express 추가 소프트웨어 2010 Express

Microsoft Visual Studio Express 2013 for Windows Desktop

Visual Studio Express 2013 for Windows Desktop을 사용하면 C#, Visual Basic 및 C++로 데스크톱 앱을 만들 수 있습니다. WPF(Windows Presentation Foundation), Windows Forms 및 Win32도 지원됩니다.

30일 안에 Visual Studio에 Microsoft 계정으로 로그인하면 여러 컴퓨터에서 설정을 동기화하고 제품을 등록할 수 있습니다.

시스템 요구 사항

다운로드 언어

한국어

설치 옵션

Microsoft Visual Studio Express 2013 for Windows Desktop 언어 팩 - český

지금 다운로드

한국어를 선택하도록 하자.

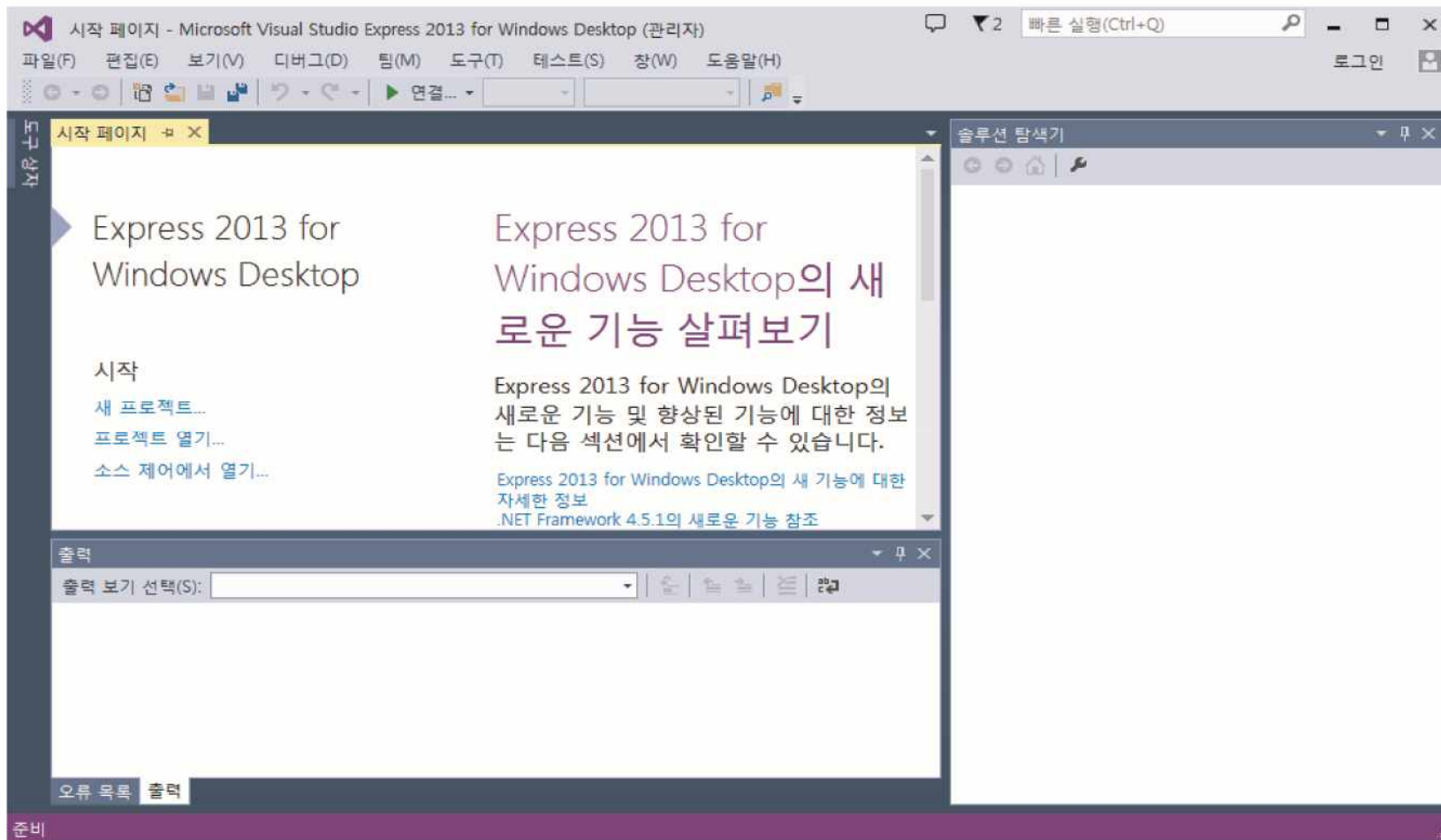
한국어

지금 설치

클릭하여 설치를 시작한다.



비주얼 C++의 실행





솔루션과 프로젝트

- 솔루션(solution): 하나의 애플리케이션을 만들기 위한 프로젝트의 그룹.
- 프로젝트(project): 하나의 실행 파일을 만들기 위하여 필요한 파일들의 그룹

솔루션



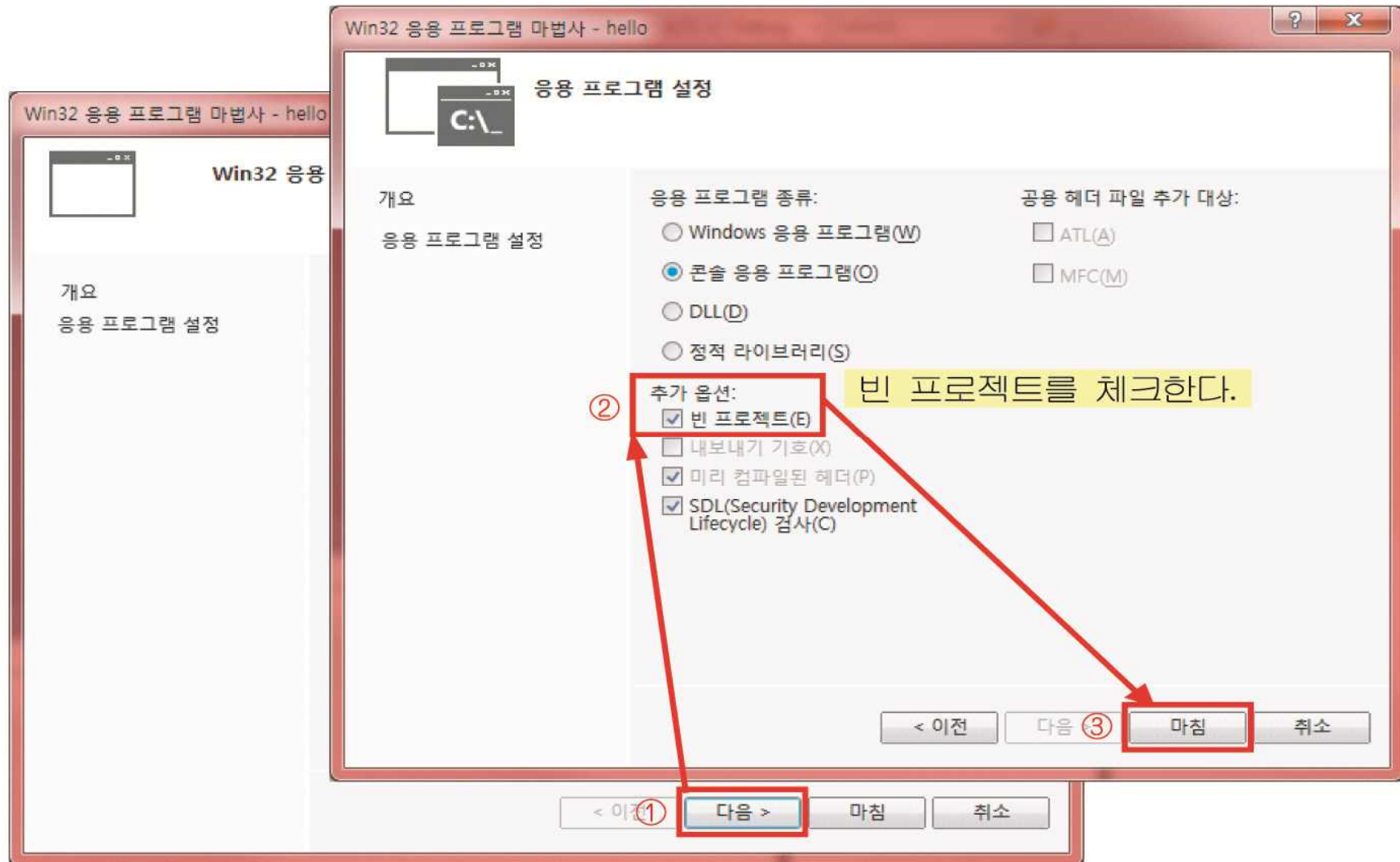
프로젝트

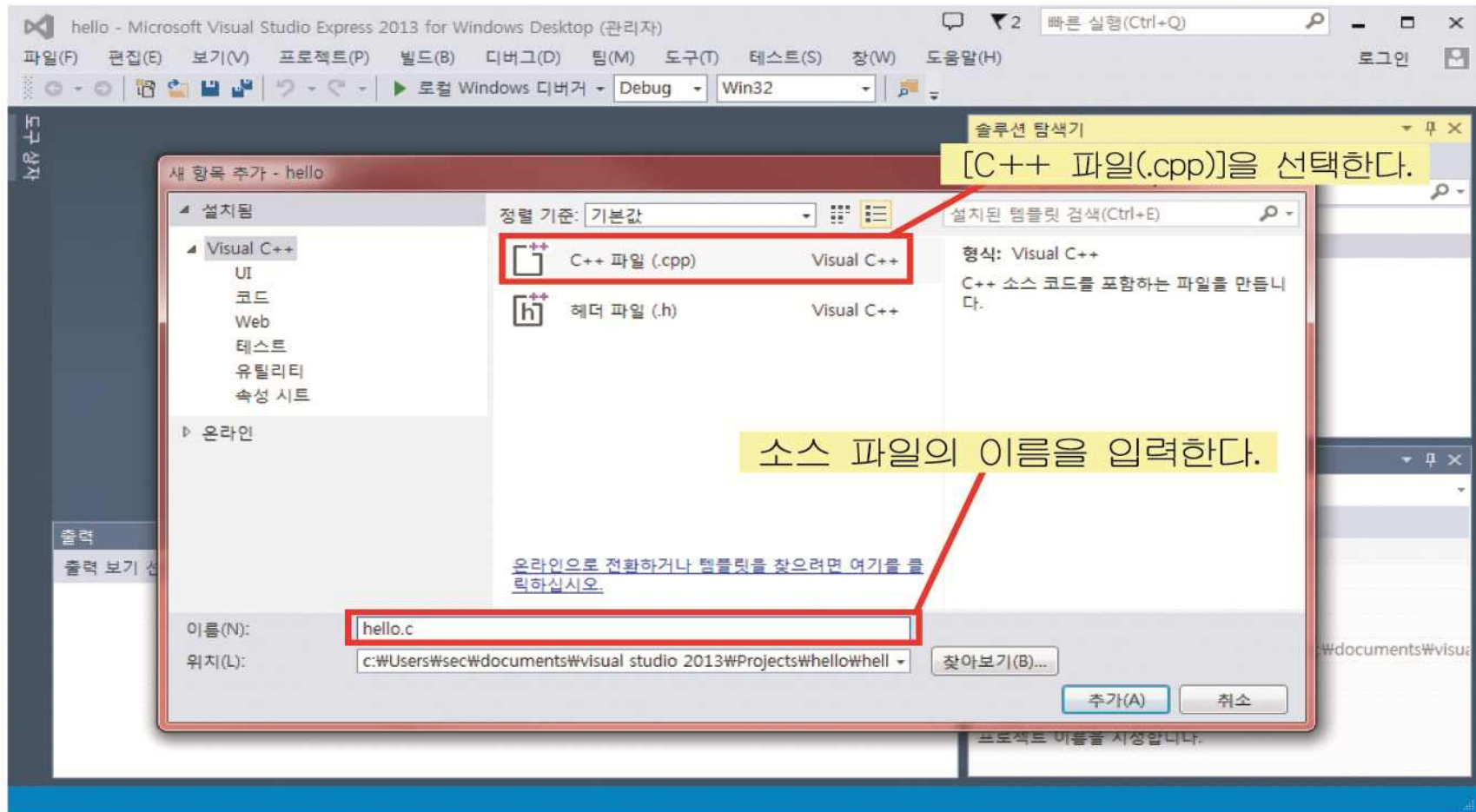


프로젝트



프로젝트







hello - Microsoft Visual Studio Express 2013 for Windows Desktop (관리자)

파일(F) 편집(E) 보기(V) 프로젝트(P) 빌드(B) 디버거(D) 팀(M) 도구(T) 테스트(S) 창(W) 도움말(H) 로그인

로컬 Windows 디버거 Debug Win32

hello.c

(전역 범위) main(void)

```
/* 첫 번째 프로그램 */  
  
#include <stdio.h>  
  
int main(void)  
{  
    printf("Hello World!");  
    return 0;  
}
```

소스 코드를 입력하는 윈도우

솔루션 탐색기

솔루션 탐색기 검색(Ctrl+;)

솔루션 'hello' (1 프로젝트)

- hello
 - 리소스 파일
 - 소스 파일
 - hello.c
 - 외부 종속성
 - 헤더 파일

프로젝트에 속하는 파일들을 보여준다.

속성

main VCodeFunction

C++

(Name)	main
File	c:\users\sec\documents\h
FullName	main
IsDefault	False
IsDelete	False

C++

출력

출력 보기 선택(S): 빌드

컴파일과 링크 과정에서의 메시지 출력

빌드했습니다. 줄: 1 열: 1 문자: 1 INS



소스 코드 입력

```
/* 첫번째 프로그램 */  
#include <stdio.h>  
  
int main(void)  
{  
    printf("Hello World!");  
    return 0;  
}
```

한글자
도
틀리면
안됨!





hello - Microsoft Visual Studio Express 2013 for Windows Desktop (관리자)

파일(F) 편집(E) 보기(V) 프로젝트(P) 빌드(B) 디버그(D) 팀(M) 도구(T) 테스트(S) 창(W) 도움말(H) 로그인

솔루션 빌드(B) F7

솔루션 다시 빌드(S) Ctrl
솔루션 정리(E)
솔루션에서 코드 분석 실행(Y) Alt
프로젝트만(J)
구성 관리자(O)...

컴파일, 링크하여서 실행 파일을 생성한다.

```
hello.c  
(전역 범위)  
/* 첫 번째 프로그램 */  
  
#include <stdio.h>  
  
int main(void)  
{  
    printf("Hello World!");  
    return 0;  
}
```

출력

출력 보기 선택(S): 빌드

1>----- 모두 다시 빌드 시작: 프로젝트: hello, 구성: Debug Win32 -----
1> hello.c
1> hello.vcxproj -> c:\users\sec\documents\visual studio 2013\Projects\hello\Debug\hello.exe
===== 모두 다시 빌드: 성공 1, 실패 0, 생략 0 =====

오류가 없다면 이러한 메시지가 나온다.

모두 다시 빌드했습니다.

속성

소스 파일 필터 속성

고급

SCC 파일	True
구문 분석 파일	True

일반

(이름)	소스 파일
고유 식별자	{4FC737F1-C7A5-4376-A066-...}

(이름)

필터 이름을 지정합니다.



hello - Microsoft Visual Studio Express 2013 for Windows Desktop (관리자)

파일(F) 편집(E) 보기(V) 프로젝트(P) 빌드(B) 디버그(D) 팀(M) 도구(T) 테스트(S) 창(W) 도움말(H) 로그인

창(W)

- ▶ 디버깅 시작(S) F5
- ▶ 디버깅하지 않고 시작(H) Ctrl+F5
- ▶ 프로세스에 연결(P)...
- ▶ 예외(X)... Ctrl+D, E
- ▶ 한 단계씩 코드 실행(I) F11
- ▶ 프로시저 단위 실행(O) F10
- ▶ 중단점 설정/해제(G) F9
- ▶ 새 중단점(B)
- ▶ 모든 중단점 삭제(D) Ctrl+Shift+F9
- ▶ 옵션 및 설정(G)...

hello.c

```
/* 첫 번째 프로그램 */  
  
#include <stdio.h>  
  
int main(void)  
{  
    printf("Hello World!");  
    return 0;  
}
```

출력

출력 보기 선택(S): 빌드

프로그래밍의 출력 항상 나오는 메시지

C:\Windows\system32\cmd.exe

Hello World 계속하려면 아무 키나 누르십시오 . . .

main
c:\Users\WsecWdocuments\W
main
False
False
C++

빌드했습니다.



우리가 작성한 프로그램

```
/* 첫번째 프로그램 */  
#include <stdio.h>  
  
int main(void)  
{  
    printf("Hello World!");  
    return 0;  
}
```

다음 장에서
자세히
설명할
것입니다.





중간 점검

1. 새로운 프로젝트를 생성하고 프로젝트에 소스 파일을 추가하는 메뉴는 무엇인가?
2. 프로젝트에 속하는 소스 파일을 컴파일하여 실행하는 메뉴는 무엇인가?
3. C 언어에서는 대문자와 소문자를 구별하는가?





Q & A

