

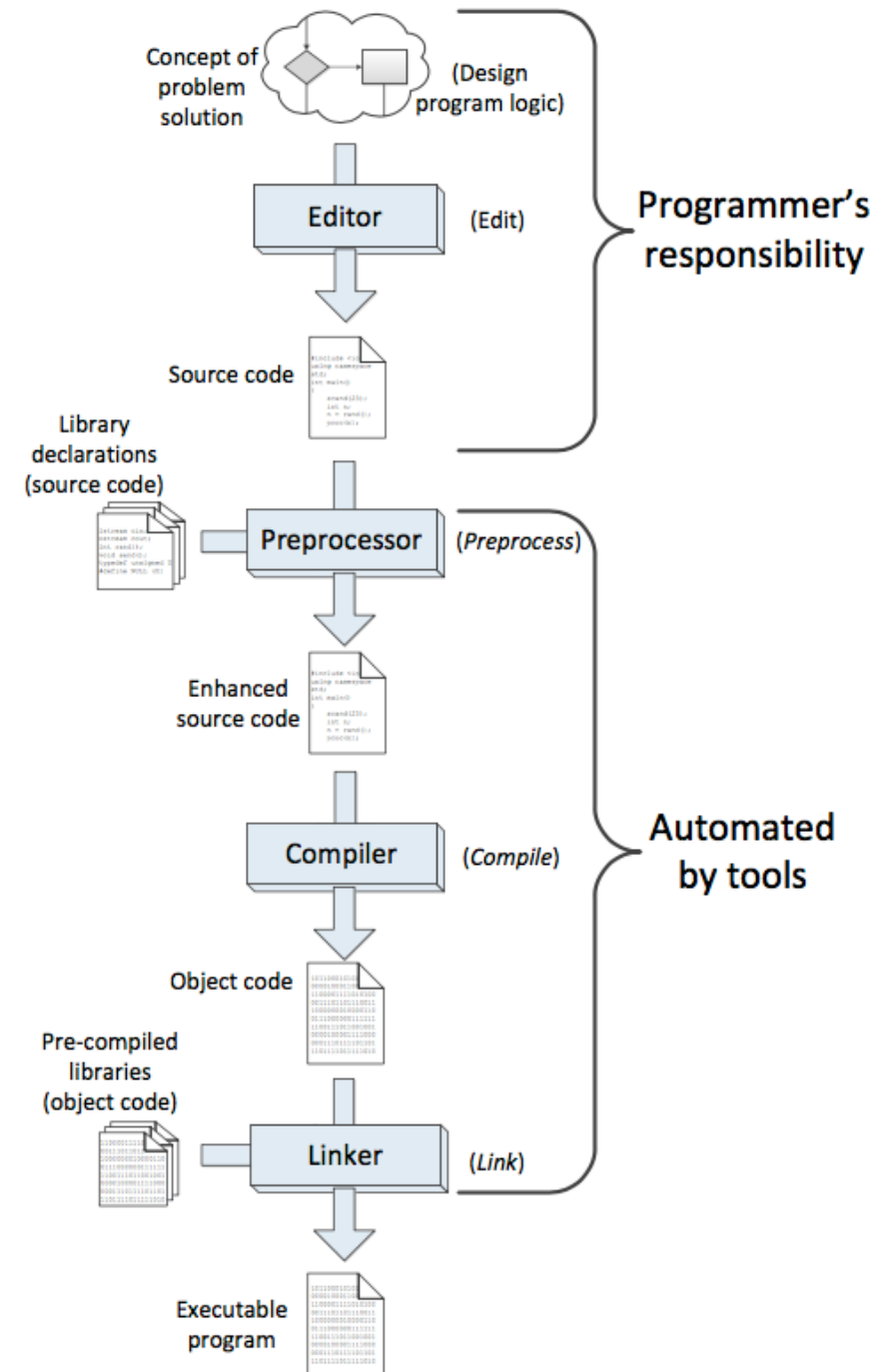


C++ 스터디

0. Visual Studio 설치 및 C++ 기초

배우기 전에..

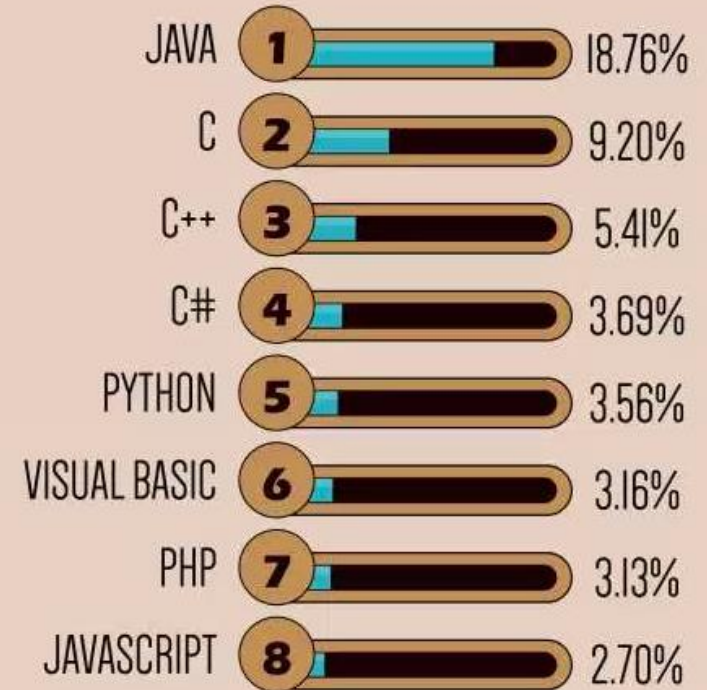
- Editor : 우리가 코드를 작성하는 IDE (파이참 , VS 등등..)
- Preprocessor : 전처리 과정으로 include, define 등의 컴파일 전의 사전 작업을 실시 한다 #include <iostream>, #define PI 3.14 등등
- Compiler : 우리가 쓴 소스 코드를 기계어로 번역해 주는 작업을 한다.
- Linker : 우리가 쓴 소스 코드와 미리 Compiled 된 라이브러리를 연결 시켜서 완전한 프로그램을 만들어 준다



왜 C++을 배우는 거죠?

- C++은 범용 프로그래밍 언어이다. 기계 제어를 위한 저 수준의 메모리 조작도 가능하며, 동시에 객체지향같은 추상화 개념도 지원한다.
(어려워 보이지만 옛날 언어 C도 지원하고 최신 기능도 된다는 뜻)
- C++ 은 어렵지만 메모리를 프로그래머가 원하는 만큼만 사용할 수 있다는 점에서 최적화가 강력하다는 장점이 있다
- 빅 데이터의 연산 , 게임 등등에서 많이 쓰이는 것을 볼 수가 있다
- 하드 코어한 컴퓨터 자원이 많이 드는) 프로그램을 만들 때 가장 좋다
- 유명한 프레임워크 : Unity, Unreal 같은 게임 엔진

TIOBE TOP LANGUAGES AS ON NOV 2016



C++ VS 파이썬

- 파이썬은 자동 메모리 관리 기능을 지원하지만 C++은 지원하지 않는다.
- C++의 변수는 타입이 고정되어 있으며 처음 선언할 때 명시해줘야 하며 서로간의 변환이 어렵다. 또한 지역변수의 이용에 대해 파이썬보다 엄격하다.
- C++의 함수들은 입력과 출력값에 대한 고정된 타입을 지녀 정해진 타입만을 주고 받을 수 있다.
- 파이썬은 C++보다 동작속도가 느리고 하위호환성(2.7버전 같은거)이 떨어지는 대신 다양한 라이브러리들이 지원된다.
- C++의 문법이 더 어렵고 비교적 코드 길이가 길다.
- 더 자세한 내용은 <https://www.educba.com/python-vs-c-plus-plus/> 참조

우리가 사용할 툴은!



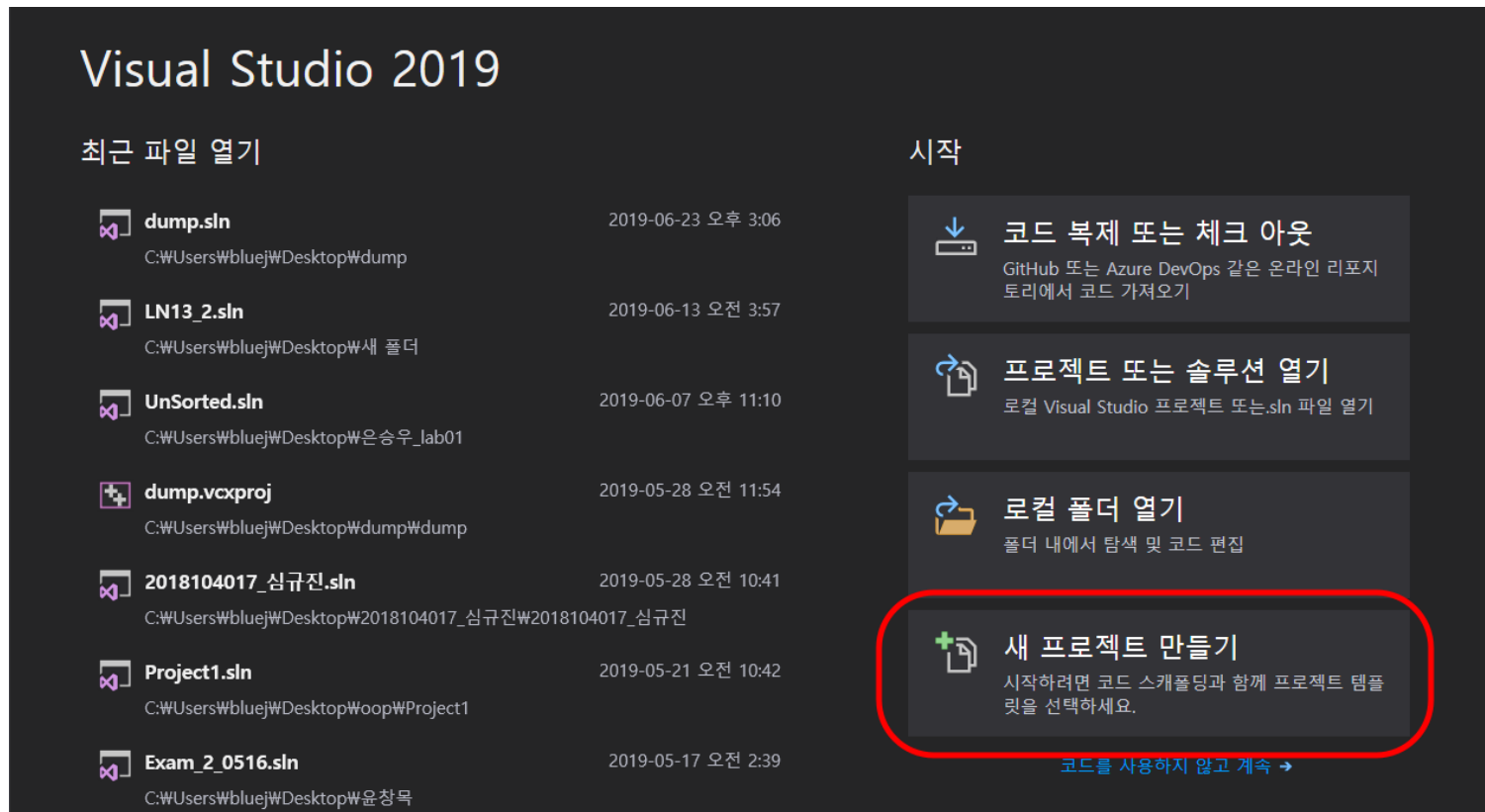
비주얼 스튜디오이다! 설치법은 아래 링크를
참조하자

<https://blog.naver.com/tipsware/221505528605>



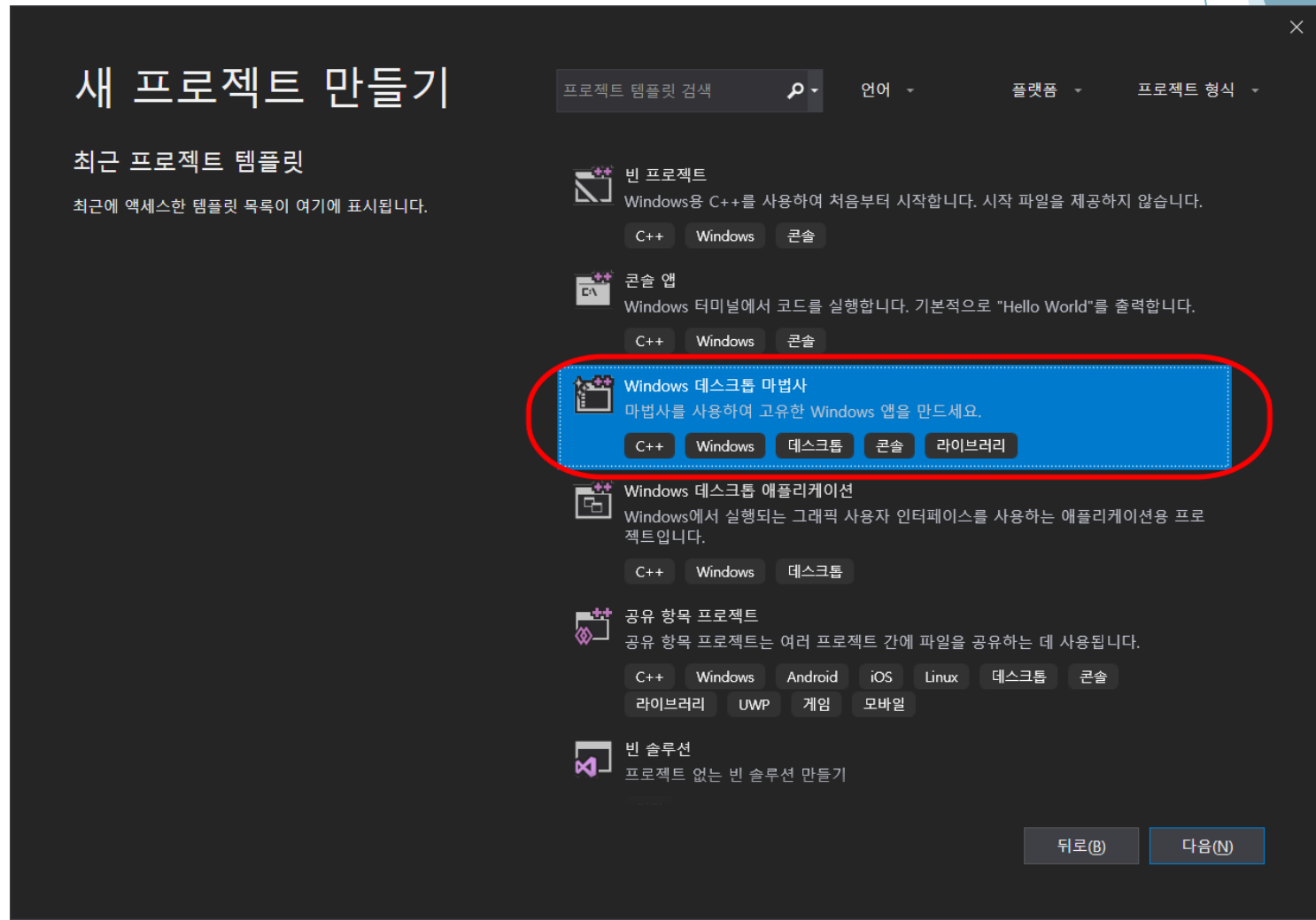
프로젝트 시작 하는 법

- 처음 켜보면 이런 화면이다. 새 프로젝트 만들기를 누른다.



프로젝트 시작 하는 법

- Windows 데스크톱 마법사를 클릭하고 다음으로 넘어간다.



프로젝트 시작 하는 법

- 프로젝트의 이름 및 경로를 설정하고 만들기를 누른다.

×

새 프로젝트 구성

Windows 데스크톱 마법사 C++ Windows 데스크톱 콘솔 라이브러리

프로젝트 이름

Project1

위치

C:\Users\bluej\Desktop\방향프로그래밍\c++스터디\tutorial\

...

솔루션 이름 ⓘ

Project1

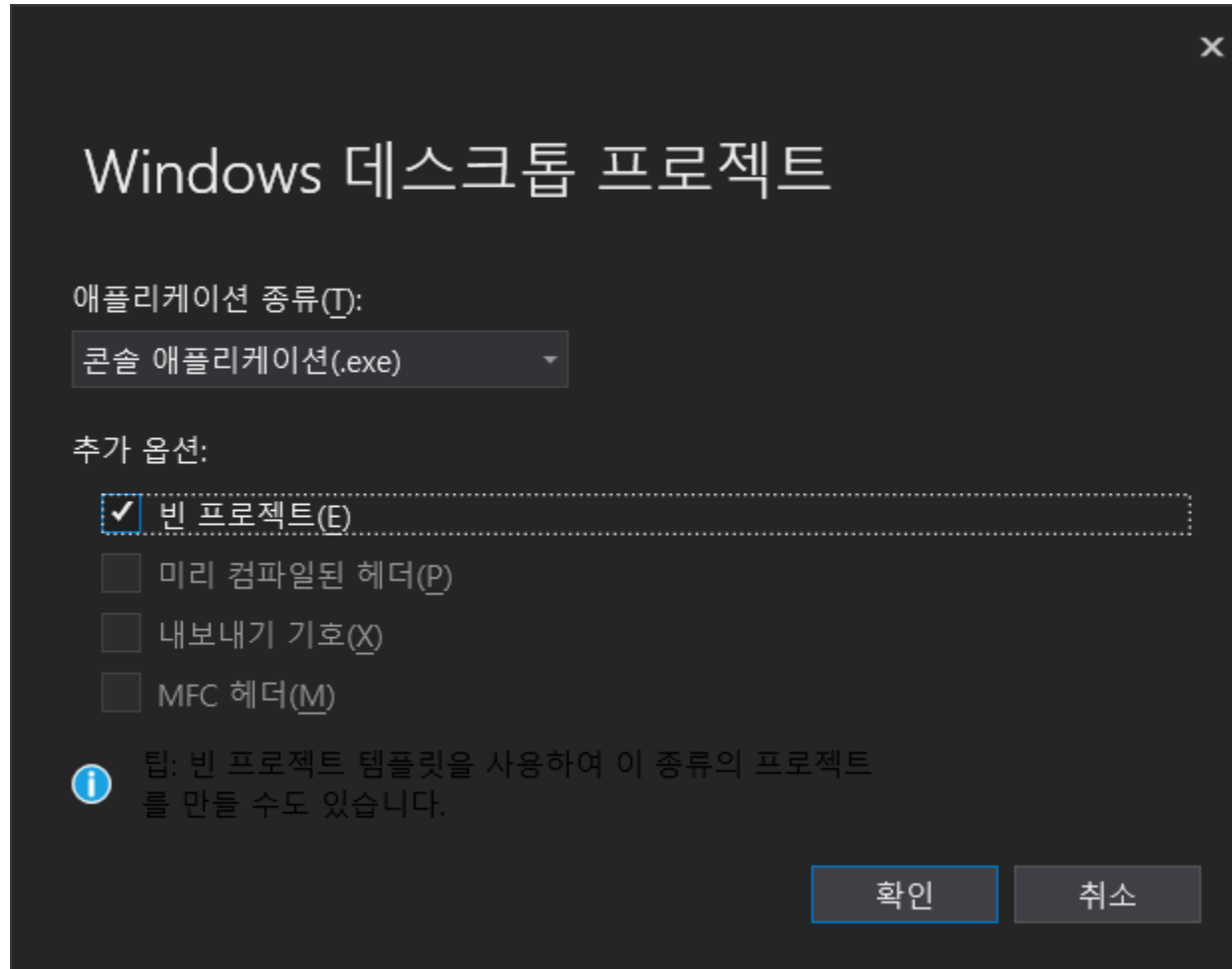
☐ 솔루션 및 프로젝트를 같은 디렉터리에 배치

뒤로(B)

만들기(C)

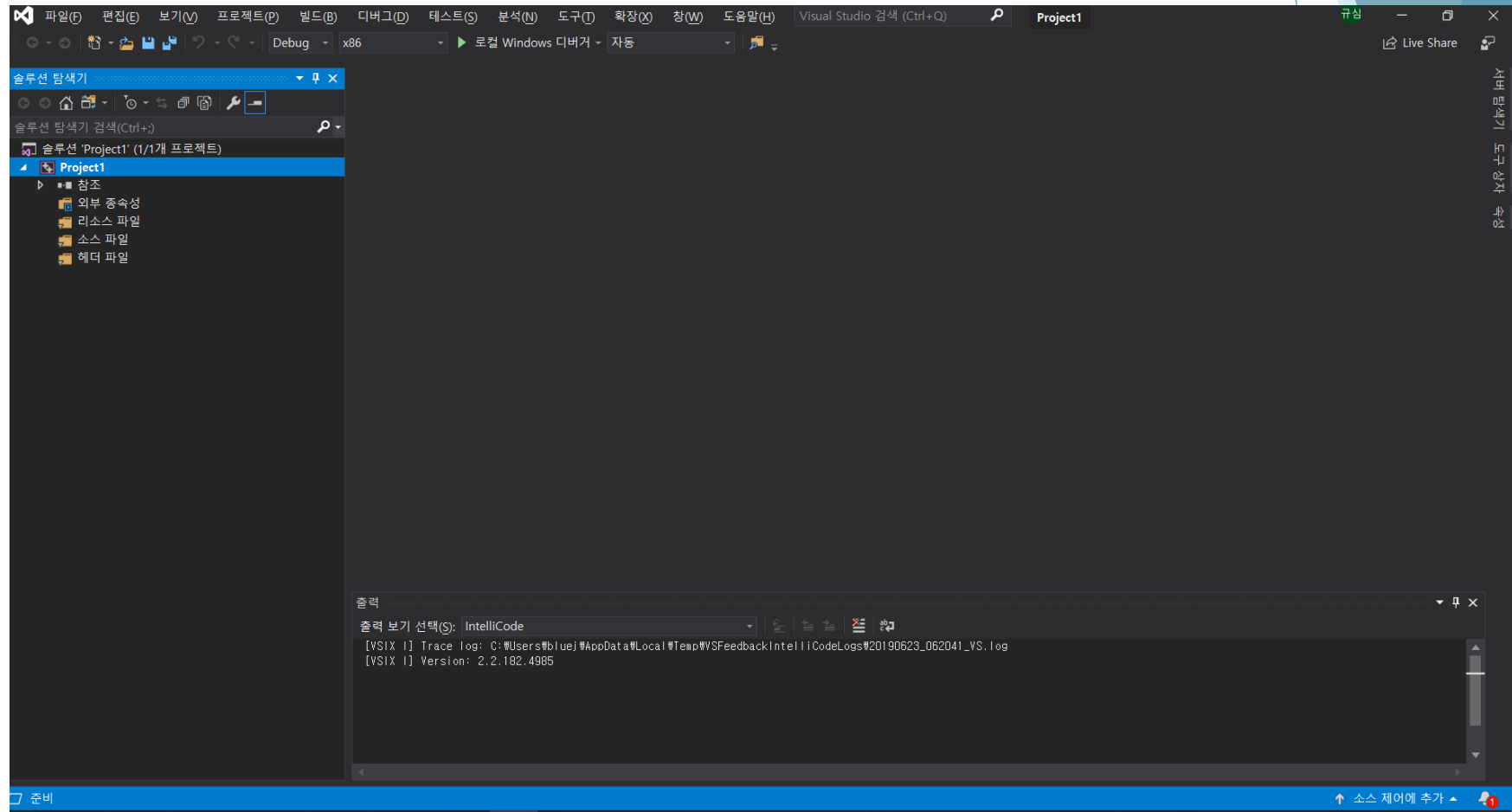
프로젝트 시작 하는 법

- 빈 프로젝트를 체크하고 확인을 누른다.



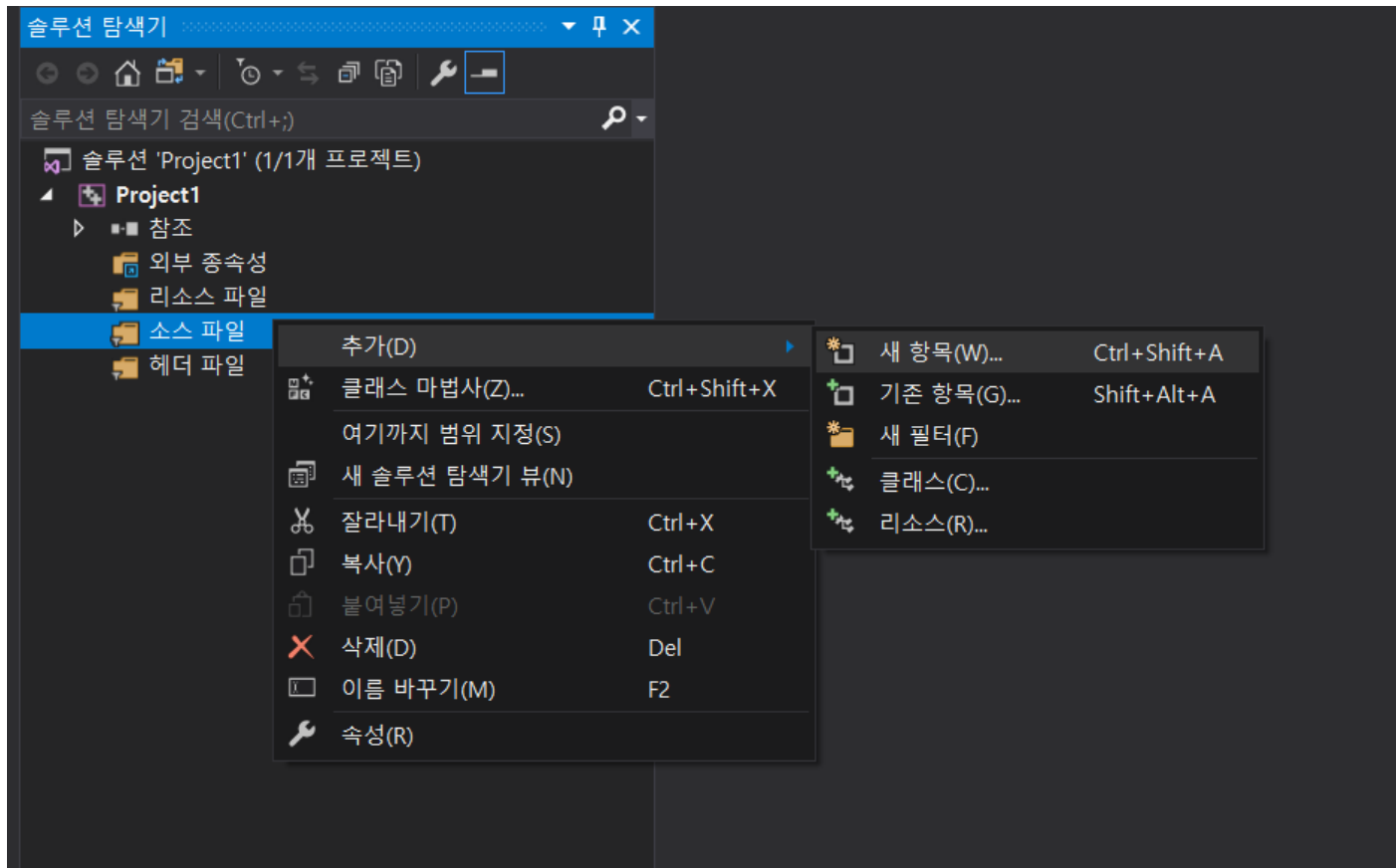
프로젝트 시작 하는 법

- 다했다면 이런 화면이 나온다.



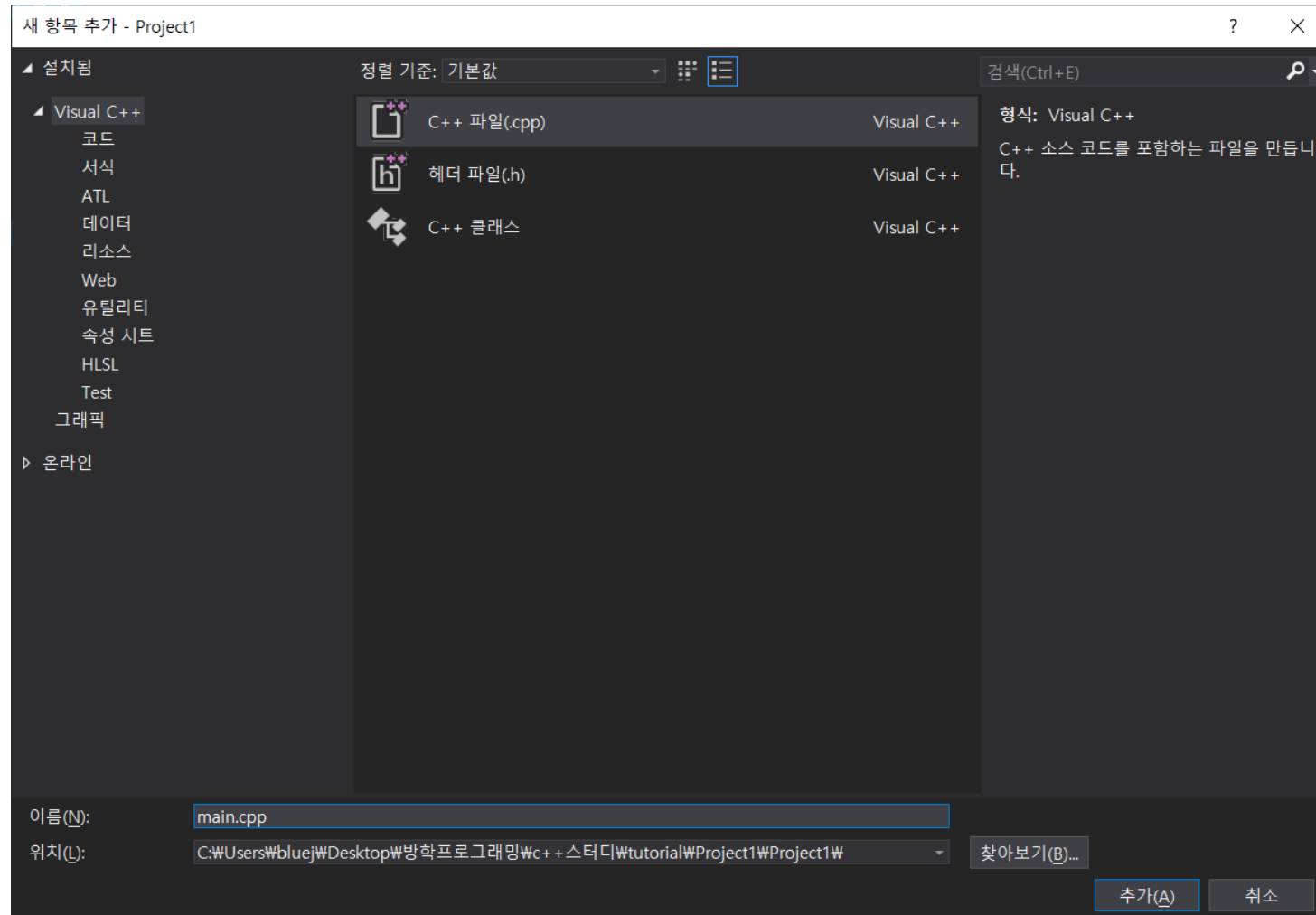
프로젝트 시작 하는 법

- 소스파일에 우클릭을 누르고 추가-새항목을 눌러준다.



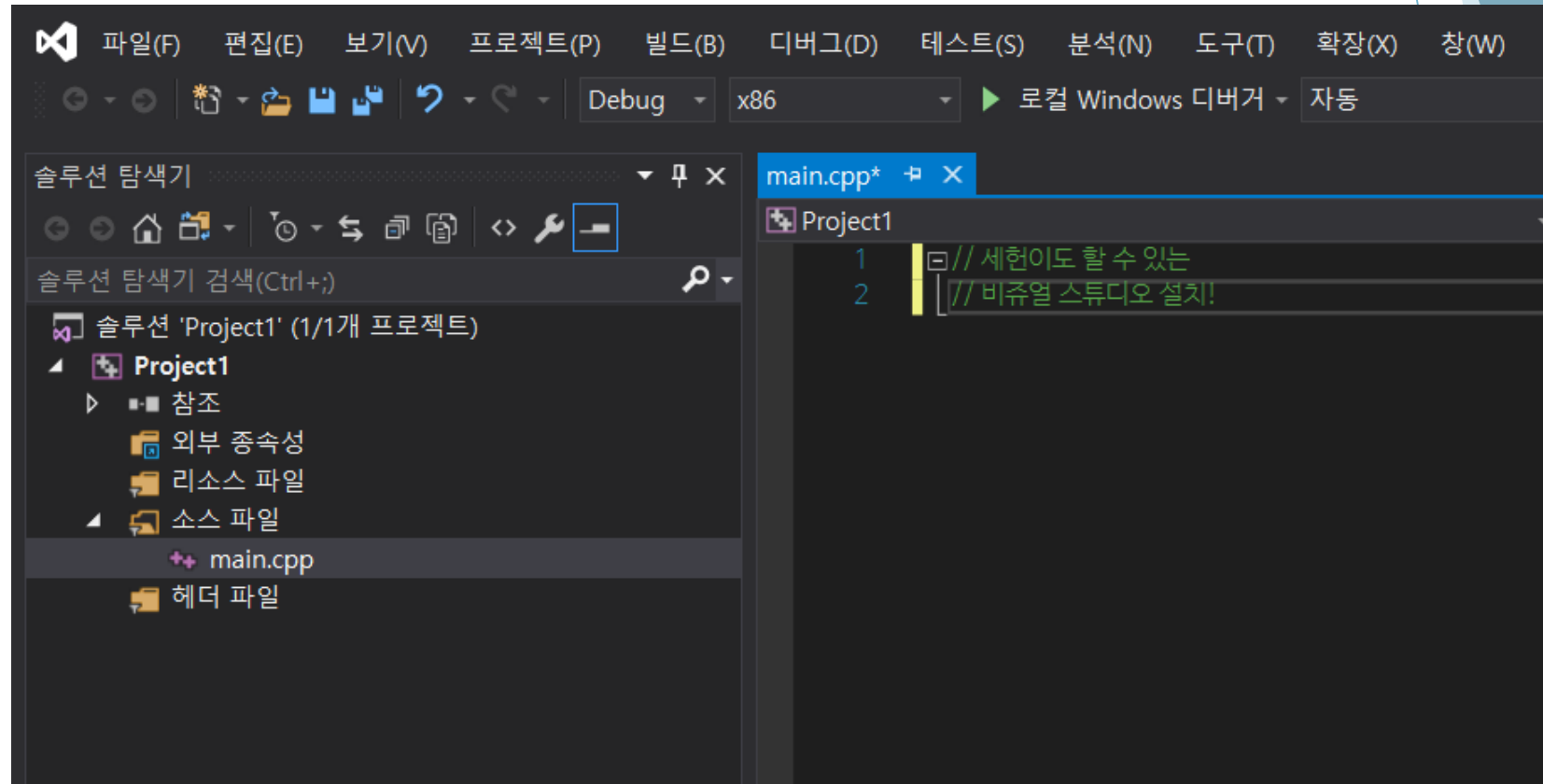
프로젝트 시작 하는 법

- C++파일을 클릭하고 이름을 main.cpp로 설정해준후 추가해준다.



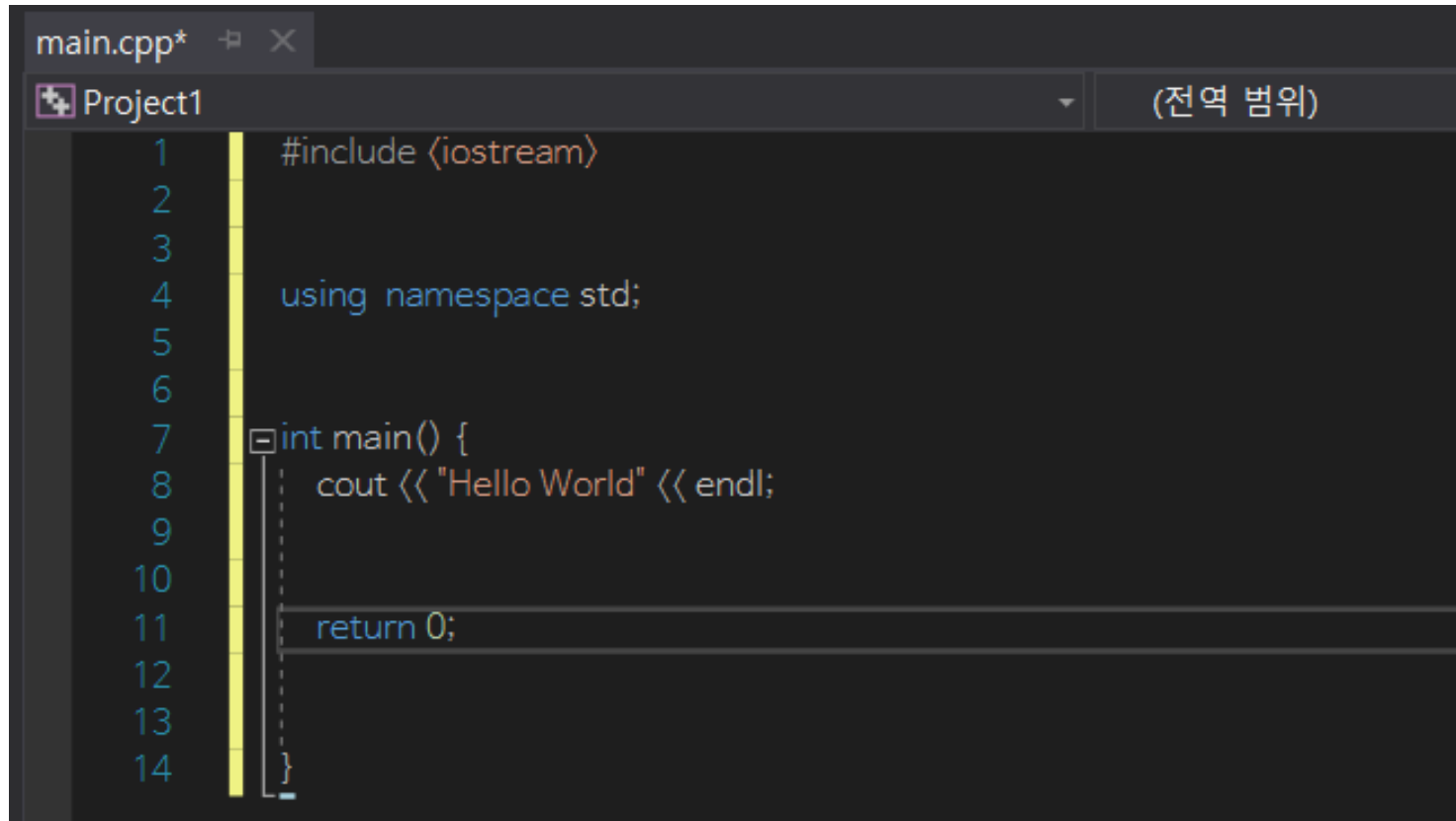
프로젝트 시작 하는 법

- 드디어 소스코드를 칠 수 있다.



첫 C++ 프로그램

- 우선 이대로 따라쳐보자. 실행하고 싶다면 Ctrl + F5를 눌러주자.



The screenshot shows a code editor window with a tab labeled 'main.cpp*'. Below the tab is a toolbar with a 'Project1' dropdown menu and a '(전역 범위)' (Global Scope) button. The code is written in C++ and is as follows:

```
1  #include <iostream>
2
3
4  using namespace std;
5
6
7  int main() {
8      cout << "Hello World" << endl;
9
10
11     return 0;
12
13
14 }
```

실행 결과

- 정상적으로는 아래 출력창이 다음과 같이 변하게 된다.

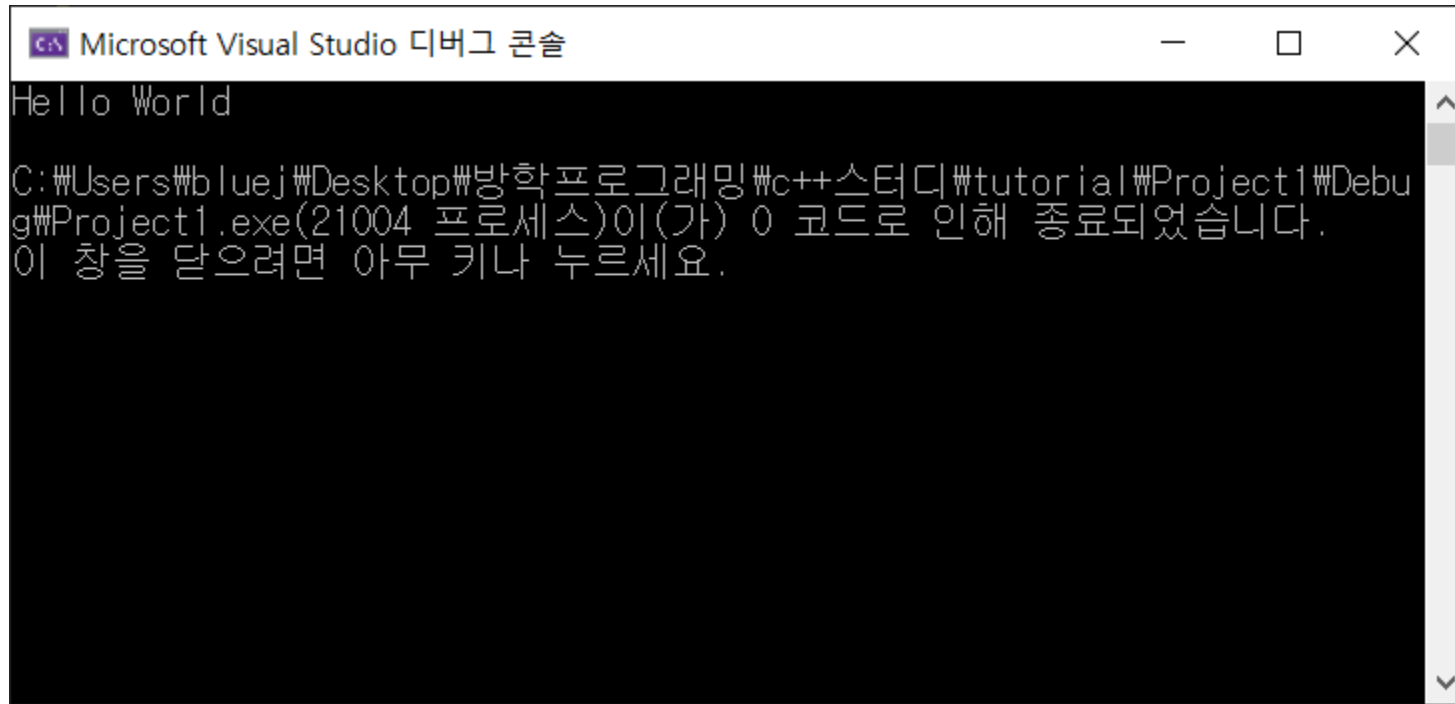
```
출력
출력 보기 선택(S): 빌드
1>----- 빌드 시작: 프로젝트: Project1, 구성: Debug Win32 -----
1>Project1.vcxproj -> C:\Users\bluej\Desktop\방학프로그래밍\c++스터디\tutorial\Project1\Debug\Project1.exe
===== 빌드: 성공 1, 실패 0, 최신 0, 생략 0 =====
```

- 만약 코드를 잘못 쳤을 경우 출력창에서 밑의 사진처럼 간단하게 원인을 살필 수 있다.

```
출력
출력 보기 선택(S): 빌드
1>----- 빌드 시작: 프로젝트: Project1, 구성: Debug Win32 -----
1>main.cpp
1>C:\Users\bluej\Desktop\방학프로그래밍\c++스터디\tutorial\Project1\Project1\main.cpp(11,2): error C2143: 구문 오류: ';'이(가) 'return' 앞에 없습니다.
1>"Project1.vcxproj" 프로젝트를 빌드했습니다. - 실패
===== 빌드: 성공 0, 실패 1, 최신 0, 생략 0 =====
```

실행 결과

- 이후 이런 콘솔 창이 뜰 것이다. 만약 뜨지 않는다면 아래 링크를 참고하자
<https://blog.bsk.im/2013/10/24/visual-studio-console-remained/>



```
Microsoft Visual Studio 디버그 콘솔
Hello World
C:\Users\bluej\Desktop\방학프로그래밍\c++스터디\tutorial\Project1\Debug\Project1.exe(21004 프로세스)이(가) 0 코드로 인해 종료되었습니다.
이 창을 닫으려면 아무 키나 누르세요.
```


첫 C++ 프로그램 분석

- `#include <iostream>`
- `<iostream>`은 C++ 표준 라이브러리 중 하나로 입출력에 관여하는 라이브러리이다.
- 앞의 `#include`는 파이썬의 `import`와 비슷하다고 생각하면 된다.
- 라이브러리가 없이 기본으로 지원되는 명령어도 있지만 `iostream`이 없다면 입력과 출력 (python의 `input`, `print`)을 못한다.
- `cout`, `cin`등이 `iostream`의 객체들이다.

```
main.cpp*  +  x
Project1  (전역 범위)
1  #include <iostream>
2
3
4  using namespace std;
5
6
7  int main() {
8      cout << "Hello World" << endl;
9
10
11     return 0;
12
13
14 }
```

첫 C++ 프로그램 분석

- using namespace std;
- 여기서 C++은 한 문장이 끝났음을 세미콜론으로 알려줄 수 있다.
- 중괄호 {}는 함수의 구현 부분인데 클래스라는 예외를 제외하면 {}에는 ;을 붙이지 않는다.
- 위 문장은 std라는 네임스페이스를 생략하고 사용하겠다는 것이다.
- std는 C++ 표준 라이브러리의 네임스페이스로 여기도 표준 라이브러리 중 하나인 iostream이 있다.
- cout 을 std::cout 처럼 써야하는 것을 생략하게 해준다.

```
main.cpp*  Project1  (전역 범위)
1  #include <iostream>
2
3
4  using namespace std;
5
6
7  int main() {
8      cout << "Hello World" << endl;
9
10
11      return 0;
12
13
14 }
```

첫 C++ 프로그램 분석

- `int main() {}`
- 메인 함수는 다른 함수들과 다르게 프로그램을 실행하면 가장 먼저 실행되는 함수로 필수적으로 있어야 한다.
- 앞의 `int`는 리턴 형식이 `int` 타입임을 알려준다.
- `()`는 사실 `(void)`의 줄임말로 함수의 인자로 아무것도 받지 않는다는 것을 알려준다.
- `{}` 안에 실행될 내용을 적어주면 된다. (함수의 구현)
- 중괄호 안에서 11번째 줄의 `return 0;` 을 통해 `int` 타입을 반환해줘 함수의 형식을 맞춰주고 있다.
- `main`에서는 `return` 생략 가능. `int` 타입인 이유는 에러코드 리턴을 통해 디버깅 위해.

```
main.cpp*  + - x
Project1  (전역 범위)
1  #include <iostream>
2
3
4  using namespace std;
5
6
7  int main() {
8      cout << "Hello World" << endl;
9
10
11     return 0;
12
13
14 }
```

첫 C++ 프로그램 분석

- `cout << "Hello World" << endl;`
- `cout`이라는 stream 객체에 "Hello World"라는 문자열과 `endl`이라는 명령을 넘겨줘 출력하게 한다.
- `endl`은 엔터와 같은 줄바꿈 역할을 해준다. 'Wn'으로 대체해도 무방하다.
- C++은 큰 따옴표 `"`를 여러 개의 문자들 (string)을 나타내는데 사용하고 작은 따옴표 `'`를 1개의 문자(char)를 나타낼 때 사용하기 때문에 주의해야한다.
- 씨-아웃 이라고 읽는다.

```
main.cpp*  + - x
Project1  (전역 범위)
1  #include <iostream>
2
3
4  using namespace std;
5
6
7  int main() {
8      cout << "Hello World" << endl;
9
10
11      return 0;
12
13
14 }
```

입력을 받아보자!

- 이런식으로 cin과 cout을 이용해 입력과 출력을 할 수 있다.
- 단 선언한 변수의 타입과 입력한 정보의 타입이 일치해야 한다.

```
main.cpp  + x
Project1  (전역 범위)  main()
1  #include <iostream>
2
3  using namespace std;
4
5  int main() {
6      int num; //C++은 변수를 선언할때에 타입을 명시해줘야 한다.
7      cout << "수를 입력해 주세요 : ";
8      cin >> num; // cin 객체 이용 입력받은 값을 num에 넣어줌.
9      cout << "입력한 수는 " << num << "입니다!" << endl;
10
11      return 0;
12  }
```

```
Microsoft Visual Studio 디버그 콘솔
수를 입력해 주세요 : 1422
입력한 수는 1422입니다!

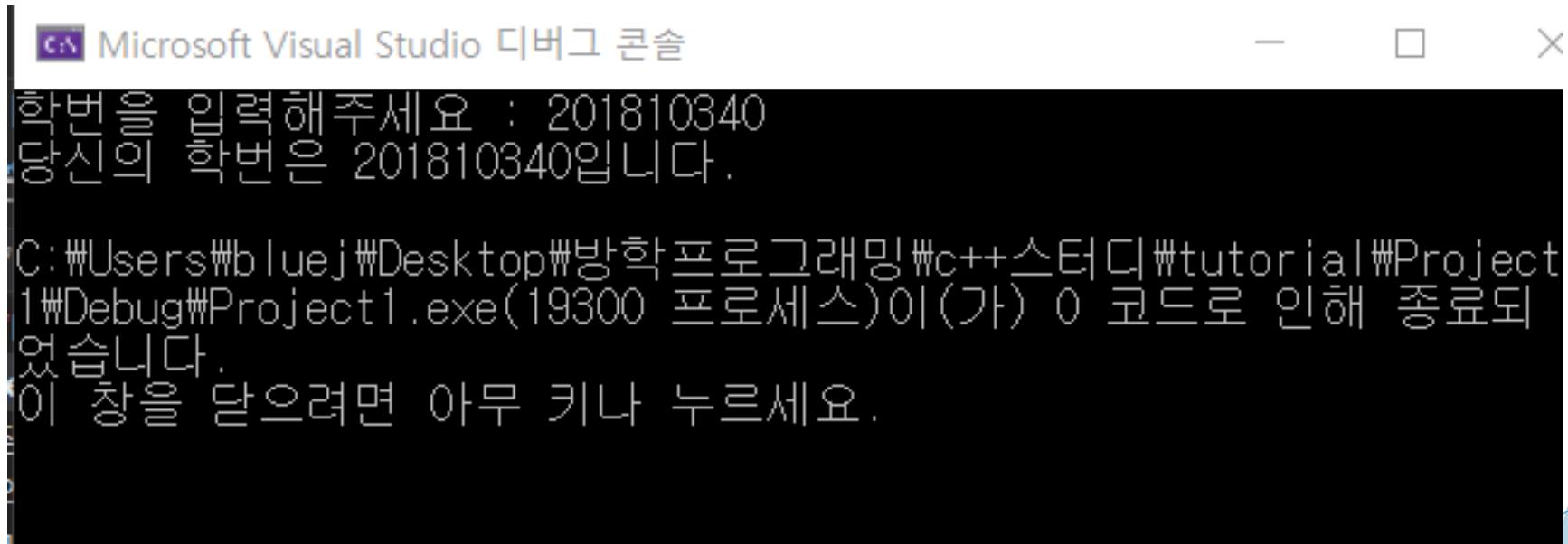
C:\Users\bluej\Desktop\방학프로그래밍\c++스터디\tutorial\Project1\Debug\Project1.exe(17372 프로세스)이(가) 0 코드로 인해 종료되었습니다.
이 창을 닫으려면 아무 키나 누르세요.
```

과제 0-1 번

- C++ 표준 라이브러리에 대해 간략히 알아보기.
- 최소 3개의 라이브러리 이름과 간단한 설명(1~2줄)로 적어서 제출해주세요.

과제 0-2번

- 자신의 학번을 입력 받아 학번을 출력하는 프로그램을 만드세요.
- 제출시에는 cpp 파일 (main.cpp) 하나만 제출하시면 됩니다.
- 입출력 예시



```
Microsoft Visual Studio 디버그 콘솔
학번을 입력해주세요 : 201810340
당신의 학번은 201810340입니다.

C:\Users\bluej\Desktop\방학프로그래밍\c++스터디\tutorial\Project1\Debug\Project1.exe(19300 프로세스)이(가) 0 코드로 인해 종료되었습니다.
이 창을 닫으려면 아무 키나 누르세요.
```