

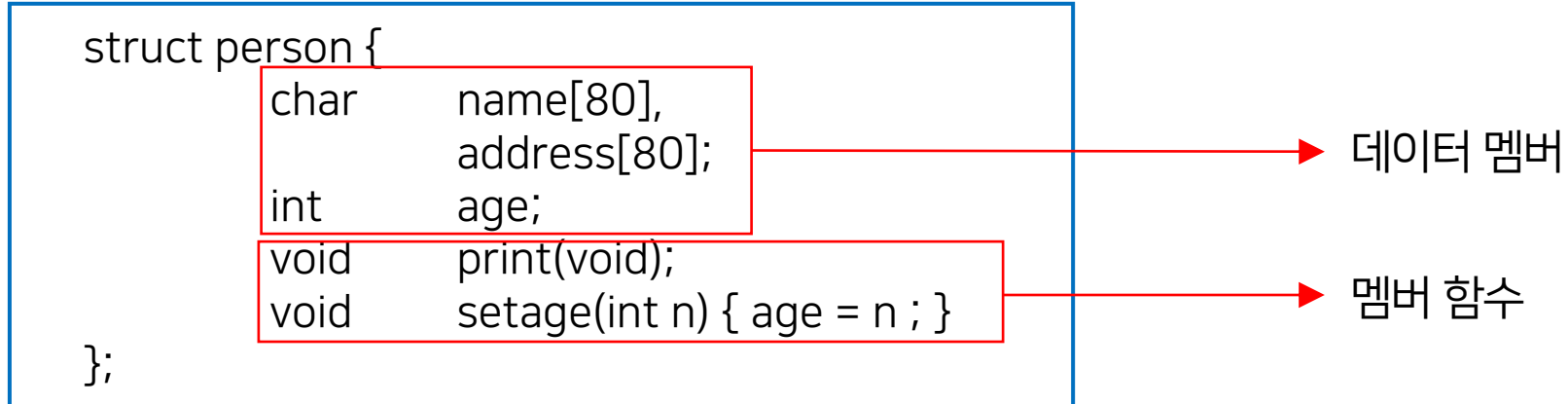
# 윈도우프로그래밍 Lab 04

- Lab03을 CStack 클래스 버전으로 수정하기 -

김한슬  
uo3359@sookmyung.ac.kr

# 구조체(struct)

- struct 안에 함수
  - C++에서는 struct 정의 안에 함수를 포함할 수 있음 - 멤버 함수
  - struct 안의 데이터 - 필드 or 데이터 멤버 or 멤버 변수



- 일반적으로 멤버 함수는 struct 밖에서 별도로 정의
  - 함수 이름 앞에 struct 이름과 범위 지정 연산자(::)를 덧붙임

```
void person::print( ) {  
    cout << "Name: " << name << endl << "Address: " << address << endl;  
}
```

# 클래스(class)

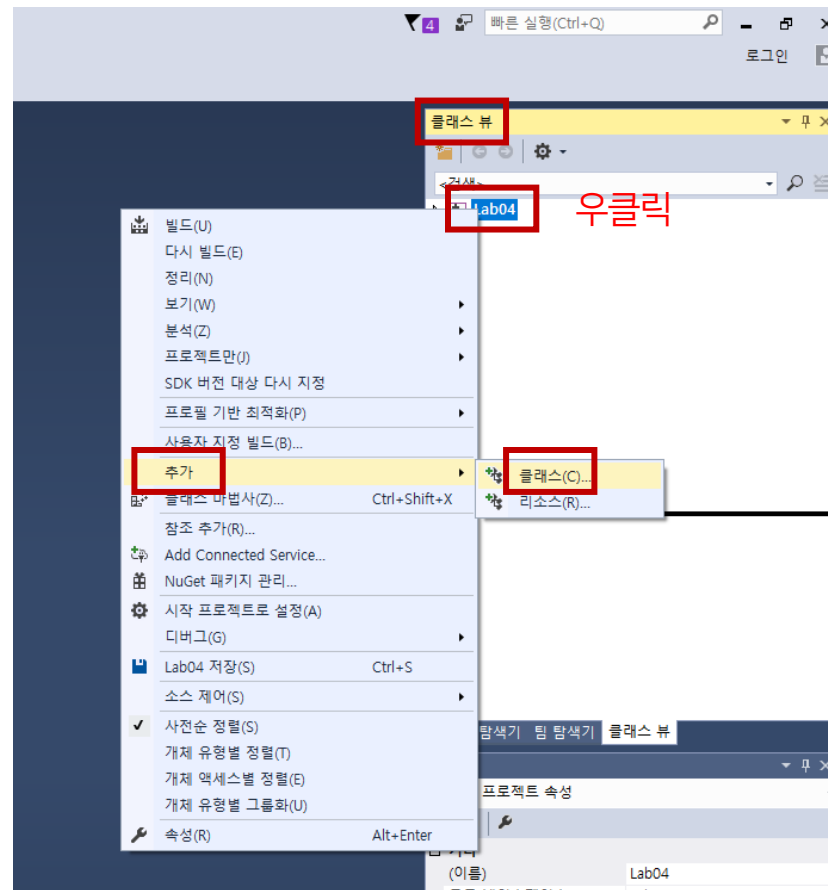
- C++에서 class와 struct는 기본적으로 같음
  - 차이 - 디폴트 가시성
    - class의 디폴트 가시성은 private
    - struct의 디폴트 가시성은 public
- public 가시성 - "public : " 이후의 필드나 멤버 함수는 어디에서나 접근 가능
- private 가시성 - "private: " 이후의 필드나 멤버 함수는 해당 struct 안에서만 접근 가능  
(해당 struct의 멤버 함수에서만 접근 가능)
- 일반적으로 데이터 멤버는 private, 멤버 함수는 public 가시성을 가짐

# 클래스(class)

- constructor( 생성자 )
  - 소속 클래스와 동일한 이름
  - 결과 값 타입 포함하지 않음 / void도 사용하지 않음
  - 기본 생성자: 파라미터를 갖지 않음 / 파라미터를 갖는 생성자도 있음
  - 클래스의 인스턴스( 객체 )가 생성될 때 그 클래스의 생성자가 호출됨
  - 데이터 멤버 초기화 / 메모리 할당 등
- destructor( 소멸자 )
  - 객체가 더 이상 존재하지 않게 될 때 호출( 생성자와 반대 )
  - 클래스 이름 앞에 ~ 기호를 붙여 표시
  - 리턴 값과 파라미터를 갖지 않음

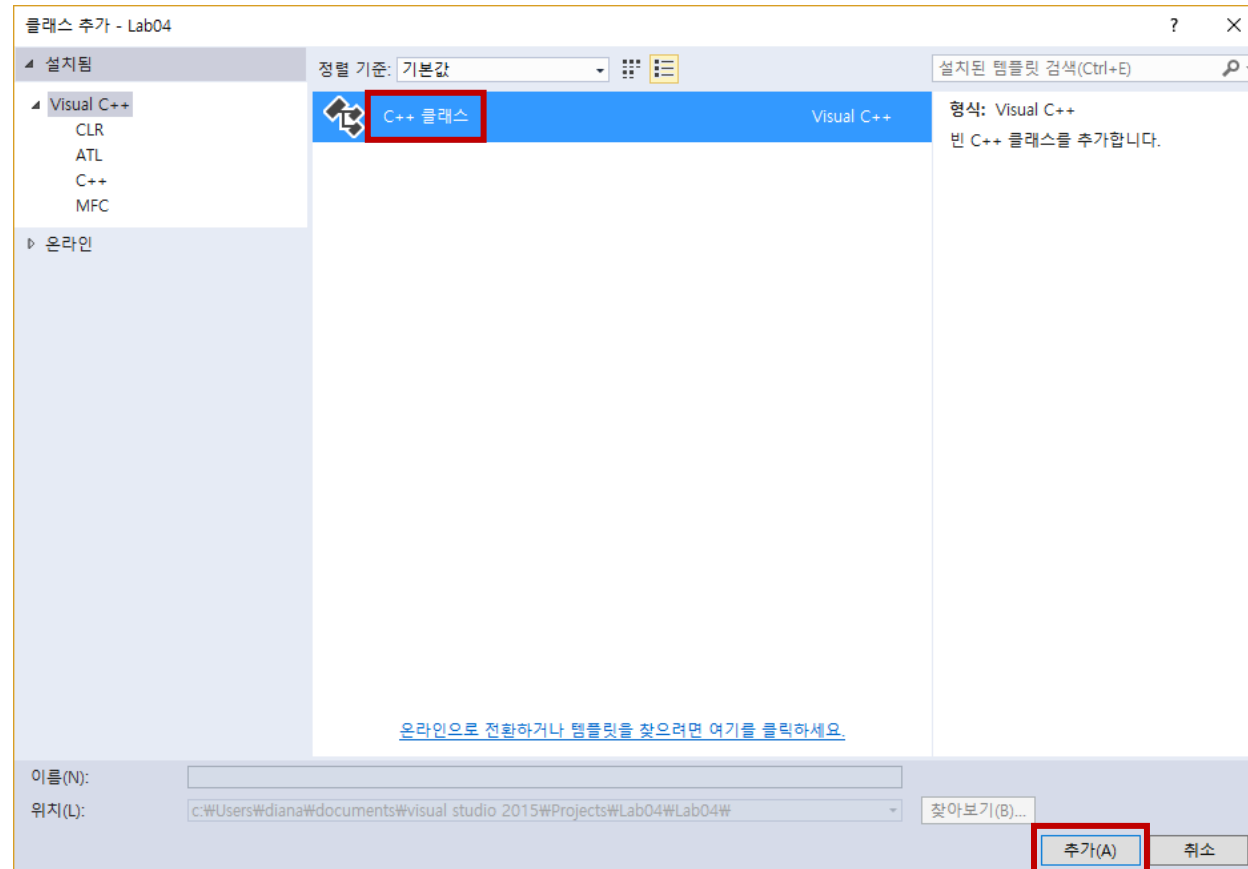
# visual studio 2015에서 클래스 추가

- [클래스 뷰] 에서 “프로젝트 이름” 우클릭 -> 추가 -> 클래스



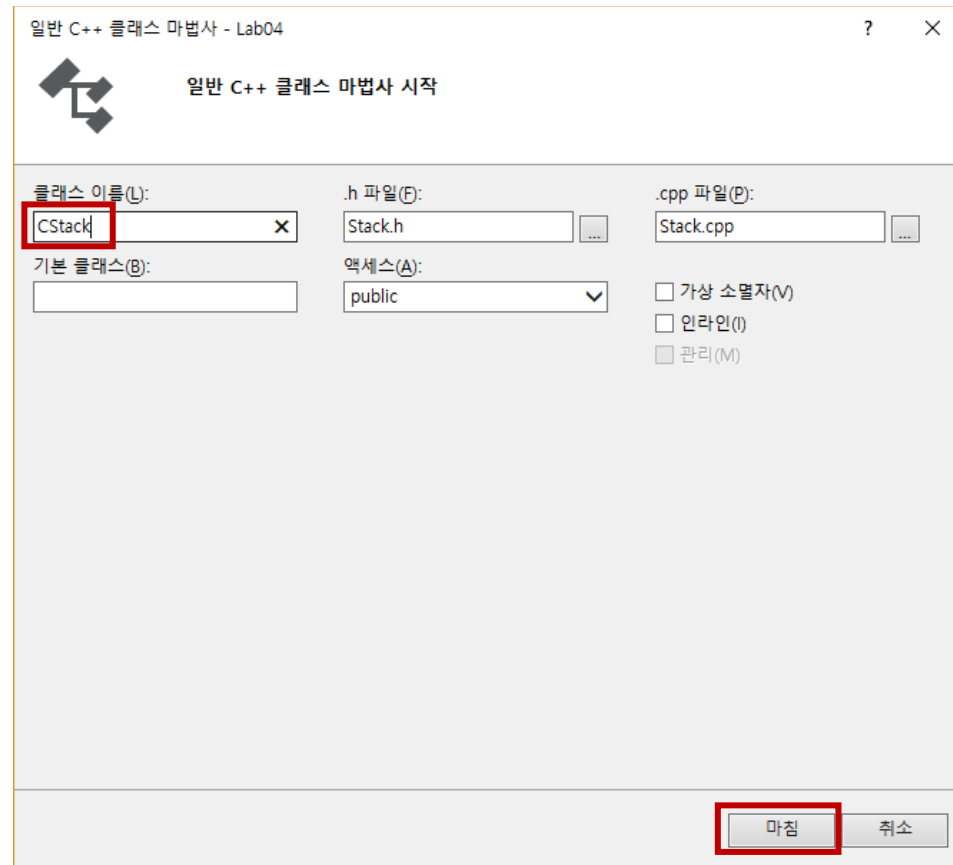
# visual studio 2015에서 클래스 추가

- [클래스 추가] 대화상자 -> [C++ 클래스] 선택 -> 추가



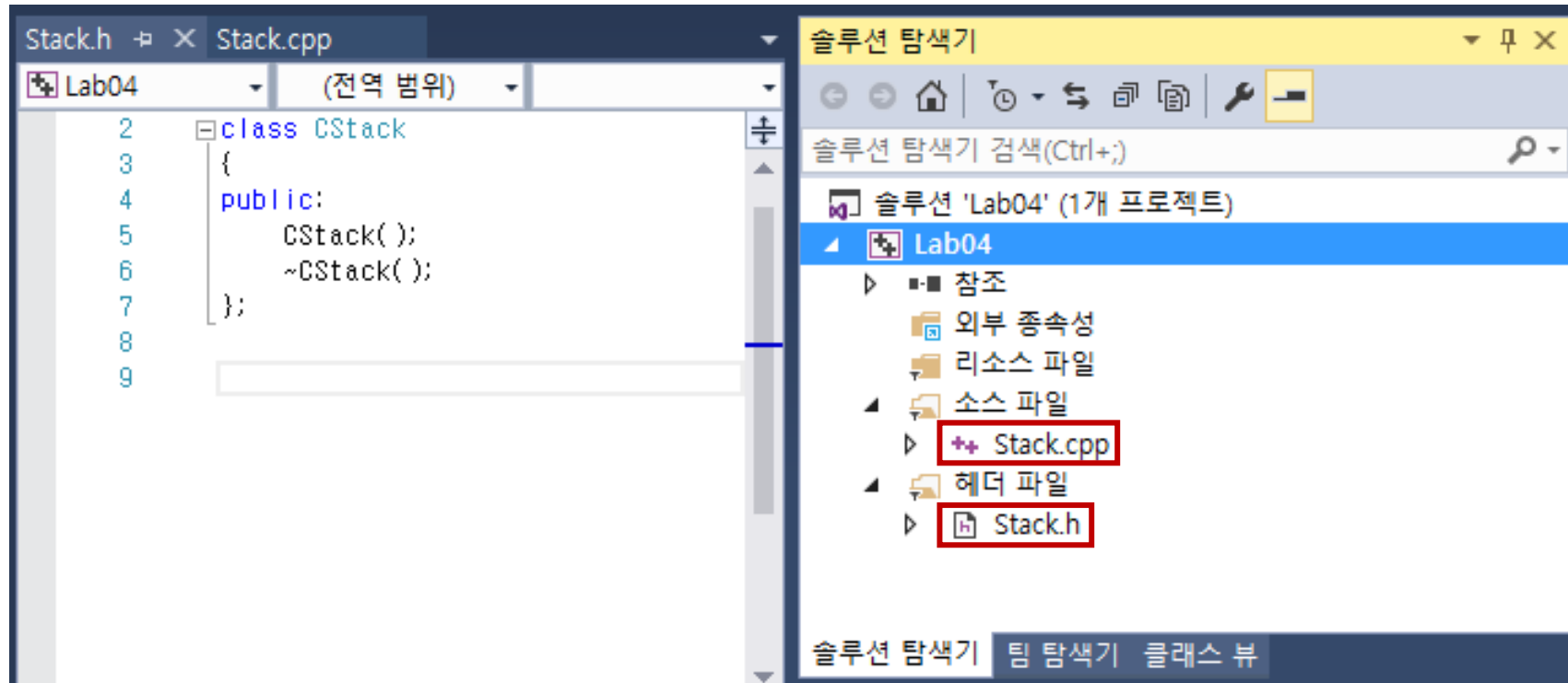
# visual studio 2015에서 클래스 추가

- [일반 C++ 클래스 마법사] -> 클래스 이름 "CStack" -> 마침



# visual studio 2015에서 클래스 추가

- 솔루션 탐색기에서 .h 파일과 .cpp 파일 생성 확인





# 과제

## 1. CStack 클래스 선언 (Stack.h)

- 데이터 멤버
  - 모두 private 가시성을 갖도록 선언함
  - friend 함수 사용 불가
- 생성자와 소멸자
  - Stack.h 파일을 만들면 자동으로 선언됨
- 멤버 함수
  - public 가시성을 갖도록 선언
  - display() 함수 생략
  - pop(), push(), 그 외 필요한 멤버함수들이 있다면 추가로 선언

```
#pragma once
class CStack
{
private:
    //데이터 멤버 선언

public:
    CStack();
    ~CStack();

    //pop(), push() 멤버함수 선언
    //추가할 멤버함수가 있다면 추가로 선언
};
```

# 과제

## 2. 멤버 함수 정의 (Stack.cpp)

- Stack.cpp 파일의 앞부분에 `#include "Stack.h"` 문장 포함
- 생성자와 소멸자
  - 생성자는 예전 Stack 구현의 `create()` 함수 기능을 담당함
  - 생성자에서는 필요한 저장 공간의 동적 생성 및 `top` 변수를 초기화 함
  - `main()`에서는 `create()` 함수 대신 생성자 사용
  - 소멸자는 생성자에서 동적으로 할당된 공간 해제
- `pop()`, `push()`와 추가로 선언한 멤버함수들이 있다면 이를 정의함

# 과제

## 2. 멤버 함수 정의 (Stack.cpp)

```
#include "Stack.h"
#include <iostream>
using namespace std;

// 생성자
CStack::CStack()
{
    // 생성자 정의(기존 Create() 함수의 기능을 하도록 함)
}

// 소멸자
CStack::~CStack()
{
    // 소멸자 정의(생성자에서 동적으로 할당한 공간 해제)
}

//멤버함수
// push() 함수 정의

// pop() 함수 정의

// 추가한 함수가 있다면 해당 함수를 정의
```

# 과제

## 3. main() 함수를 포함할 C++ 소스 파일 (main.cpp)

- CStack 클래스를 사용하도록 main() 함수를 완성함
- 파일의 앞부분에 `#include "Stack.h"` 문장 포함

# 과제 제출: 보고서

- 내용
  - 각 과제의 프로그램 소스
  - 각 과제의 실행 화면 캡처
- 제출 형식
  - 스노우보드에서 Lab04\_이름\_학번.doc 다운로드
  - 스노우보드에 제출: Lab04\_이름\_학번.zip (아래 두 파일 반드시 포함하도록)
    1. 솔루션 폴더
    2. 보고서: Lab04\_이름\_학번.doc (설명 없을 경우 감점)
- 제출 기한
  - 4월 6일(화) PM 11:59 까지