

# 객체지향설계

## Fall 2018



임성수 교수  
Introduction: 과목 소개



# 수업 내용

1. 강의 소개 ←
2. 개발 환경 구축 및 예제

# 강의 소개



**담당교수:** 임성수

- 연구실: 공5612
- 전화: 821-5448
- E-mail: sungsu@cnu.ac.kr

**수업조교:** 01분반 - 권채운(rnjscodns99@naver.com)  
02분반 - 김용건(sasin\_sg@naver.com)

**시간 및 장소**

- 이론(공5404): 금요일 10:00~12:00 (01) / 13:00~15:00 (02)
- 실습(공5415): 월요일 09:00~11:00 (01) / 11:00~13:00 (02)

# 강의 소개



## 웹사이트

- 충남대학교 이러닝지원센터 (강의 자료, 실습 과제 업로드)
  - <http://e-learn.cnu.ac.kr/> (Web & App)
- CLASSUM (질문/답변, 익명 가능)
  - <https://classum.org/> (Web & App)

## 교재 및 참고문헌

- 교재: 강의노트
- Thinking in C++ (Ch. 1,2,4,5,14,15,16)
  - <https://www.micc.unifi.it/bertini/download/programmazione/TICPP-2nd-ed-Vol-one-printed.pdf>
- 클린 소프트웨어: 애자일 원칙과 패턴, 그리고 실천 방법 (디자인 패턴 관련)

# 강의 개요



## 대상

- Java / C 프로그래밍 기초 과정을 이수한 학생

## 개요

- 객체지향 프로그래밍 언어를 이용한 소프트웨어 설계 방법론 학습
- 복잡한 소프트웨어의 체계적 설계 과정 학습 (실습, Term Project)
- 학기 초/중반부 C++ 프로그래밍 언어 학습 (Thinking in C++)
- 학기 후반부 디자인 패턴 및 구현 사례 학습 (클린 소프트웨어)

# 강의 계획



주차	내용	
1	객체지향 개요 (lecture note)	개발 환경 구축
2	클래스 (lecture note)	C++ 기초 문법
3	객체 (Chap. 1)	각 주차 실습
4	객체 생성과 활용 (Chap. 2)	<b>프로젝트 팀 구성</b>
5	추상화 및 정보 은닉 (Chap. 4, Chap. 5)	
6	상속과 구성 (Chap. 14)	<b>프로젝트 제안서 제출</b>
7	다형성과 가상함수 (Chap. 15)	
8	<b>중간고사</b>	

# 강의 계획



주차	내용	
9	템플릿의 소개 (Chap. 16)	
10	예외 처리 (lecture note)	
11	C++ 프로그래밍 보충	<b>프로젝트 중간 발표</b>
12	디자인 패턴의 소개 (Ref. 2)	
13	주요 디자인 패턴 (Ref. 2)	
14	<b>프로젝트 최종 발표 및 데모 시연</b>	<b>프로젝트 보고서 제출</b>
15	기말고사	

➤ 강의 후반부 시간이 남으면 디자인 패턴 및 사례 추가로 다룰 예정

# 학교 규정



**출석** 출석시수 3/4미달자의 성적은 “F”로 처리함

(예시 : 3-3-0 교과목의 경우 → 총45시간 중 34시간 이상 출석하여야 성적부여 조건이 되며,  
34시간 미만인 경우 출석미달(출미)로 급제점 이상 성적부여 불가(F) [학칙 제30조]

**보충 강의:** 9/24 (실습) 휴강, 12/10 보충 강의 (학기말 과제 발표)  
11/23 (이론) 휴강, 추후 보강일 지정

※ 정기 휴업일에 다른 보충강의 지정일에 반드시 강의를 실시하여야 함, 부득이 보충강의 지정일에 강의를 실시할 수 없는 경우 별도의 결강 및 보강계획을 세워 진행 하여야 함

< 2018학년도 제2학기 정기휴업에 따른 보충강의 지정일>

정기 휴업일	추석연휴			개천절	한글날
	9. 24.(월)	9. 25.(화)	9. 26.(수)	10. 3.(수)	10. 9.(화)
보강 지정일	12.10.(월)	12. 11.(화)	12. 12.(수)	12. 13.(목)	12. 14.(금)

※ 2018. 12. 10.(월) ~ 12.14.(금)은 정기 휴업일에 다른 보충강의가 실시되므로 15주차 정규수업은 2018. 12. 17.(월) ~ 12. 21(금)로 진행되어야 함



# 평가 규정



## 세부 기준(100%)

- 중간고사 (25%): 8주차
- 기말고사 (25%): 15주차
- 출석 (10%): 3/4 미달 시 F 부여 (학교 규정)
- 실습 과제 (10%): 매 실습 시간 부여
- 학기말 과제(30%): Term Project 제안/발표/결과

## 규정

- 중간 및 기말고사 이전에 연습 문제 제공
- 출석인정(초안): 가족 사망, 입원(확인서), 파견 및 학회 참석(증명서), 입대 및 예비군 훈련(확인서), 생리공결(신청서) 등

# 실습 과제



## 제출(초안)

- 매주 실습 과제에 대한 결과를 정리하여 다음 실습 시간 전까지 업로드
  - <http://e-learn.cnu.ac.kr/>
- 내용 설명, 결과, 소스코드(comment 포함)
- Delay: 제출 당일 포함 2일까지만 평가
  - 1점씩 감점(당일 실습 이후: -1점, 2일차: -2점)
  - 실습 조교에게 직접 제출

## 평가

- 실습 과제 각 3점 만점

# 학기말 과제



## 일정(초안)

- 4주차(9/28) : 프로젝트 팀 구성(팀원 명단 제출)
  - 팀 구성: 팀 당 2~4명
- 6주차(10/12) : 주제 선정 및 제안서 제출
  - 제안서: A4 2쪽, 자율 양식
  - 팀 별 주제 선정 미팅: 10/1~10/12 중
- 11주차(11/16): 테스트 프로그램 및 실행 결과(중간 발표, 데모 포함)
  - 프로젝트 전체에 대한 use case(시나리오) 작성
  - Use case 별 테스트 프로그램 코드 및 실행 결과
- 14주차(12/10): 결과 보고서(최종 발표, 데모 포함)
  - 테스트 프로그램의 수정 내용 포함
  - 결과 보고서: A4, 분량 및 양식 자율

# 학기말 과제



## 고려사항(권장)

- 텍스트 입출력에 기반한 프로그램
  - 프로그램 핵심 기능에 충실 - GUI 등 부가적 기능은 배제
- 구현 프로그래밍 언어는 Java/C++ 모두 가능
- 프로그램의 복잡도 (지난해 기준, 필수 사항 아님)
  - 라인수 (테스트 케이스 제외) > 팀원 수 × 1,000 lines
  - 클래스 수 > 팀원 수 × 10개

# 내용



1. 강의 소개

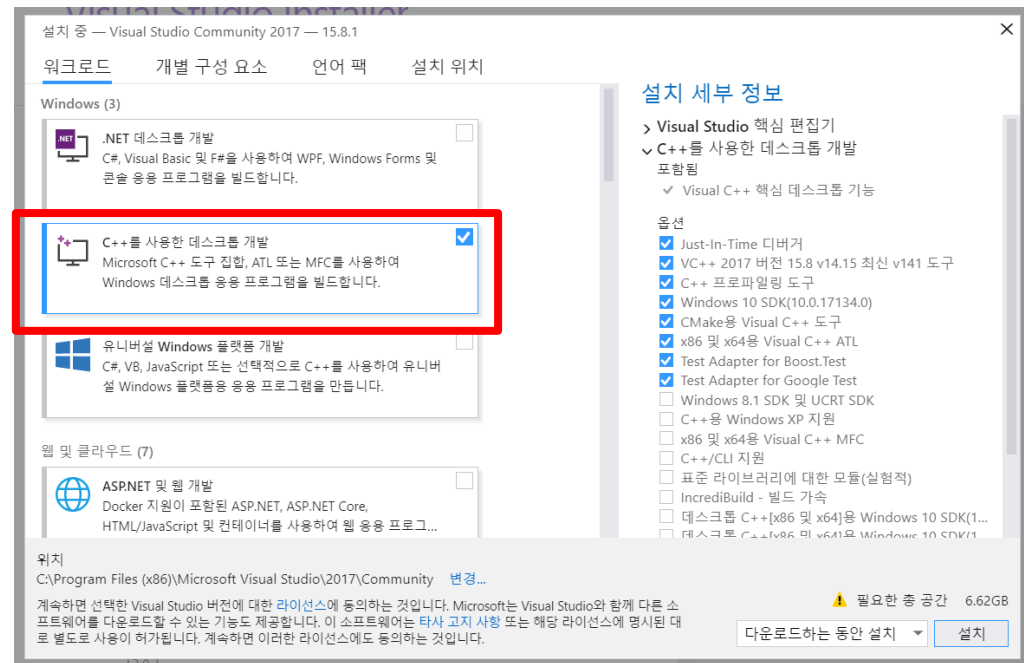
2. 개발 환경 구축 및 예제 

# 개발 환경 구축



## 통합개발환경 (IDE)

- Visual Studio 설치 (Eclipse 사용도 가능)
- <https://visualstudio.microsoft.com/ko/downloads/>



# 개발 환경 구축



## Visual Studio 설정 및 실행

- Microsoft 계정으로 로그인 및 Visual C++ 개발 설정

### Visual Studio

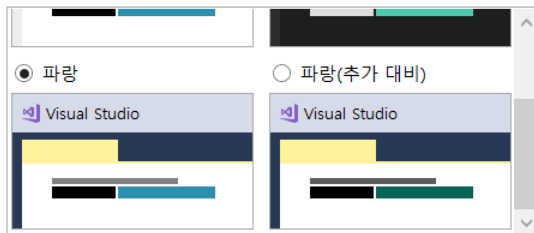
Sungsu Lim 시작

SL ssungssu@gmail.com  
Visual Studio 프로필 보기

친숙한 환경에서 시작

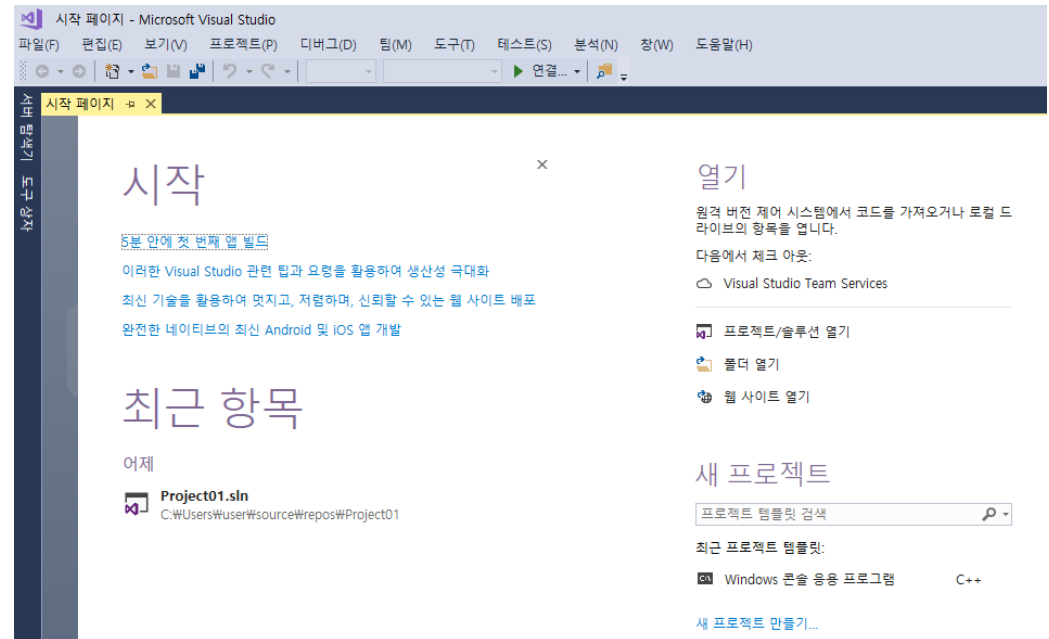
개발 설정(V): Visual C++

색 테마 선택



나중에 언제든지 이 설정을 변경할 수 있습니다.

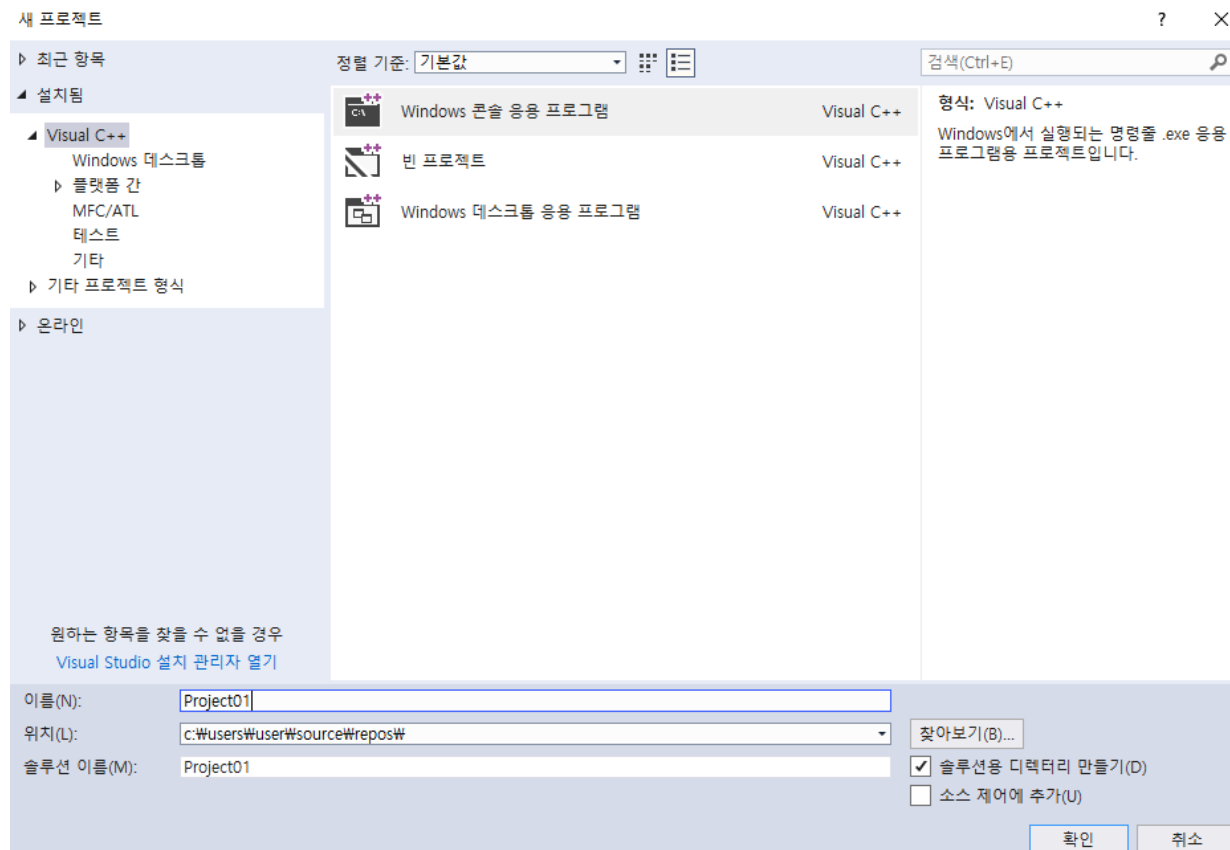
Visual Studio 시작(S)



# 프로그램 작성 - 프로젝트 생성



➤ [파일] → [새로 만들기] → [프로젝트]





# 프로그램 작성 - 프로젝트 생성



➤ [파일] → [새로 만들기] → [프로젝트]

Windows 데스크톱 프로젝트

응용 프로그램 종류(T):

콘솔 응용 프로그램(.exe) ▼

다음에 대한 일반 헤더 추가:

☐ ATL(A)

☐ MFC(M)

추가 옵션:

☐ 빈 프로젝트(E)

☐ 내보내기 기호(X)

☒ 미리 컴파일된 헤더(P)

☒ SDL(Security Development Lifecycle) 검사(C)

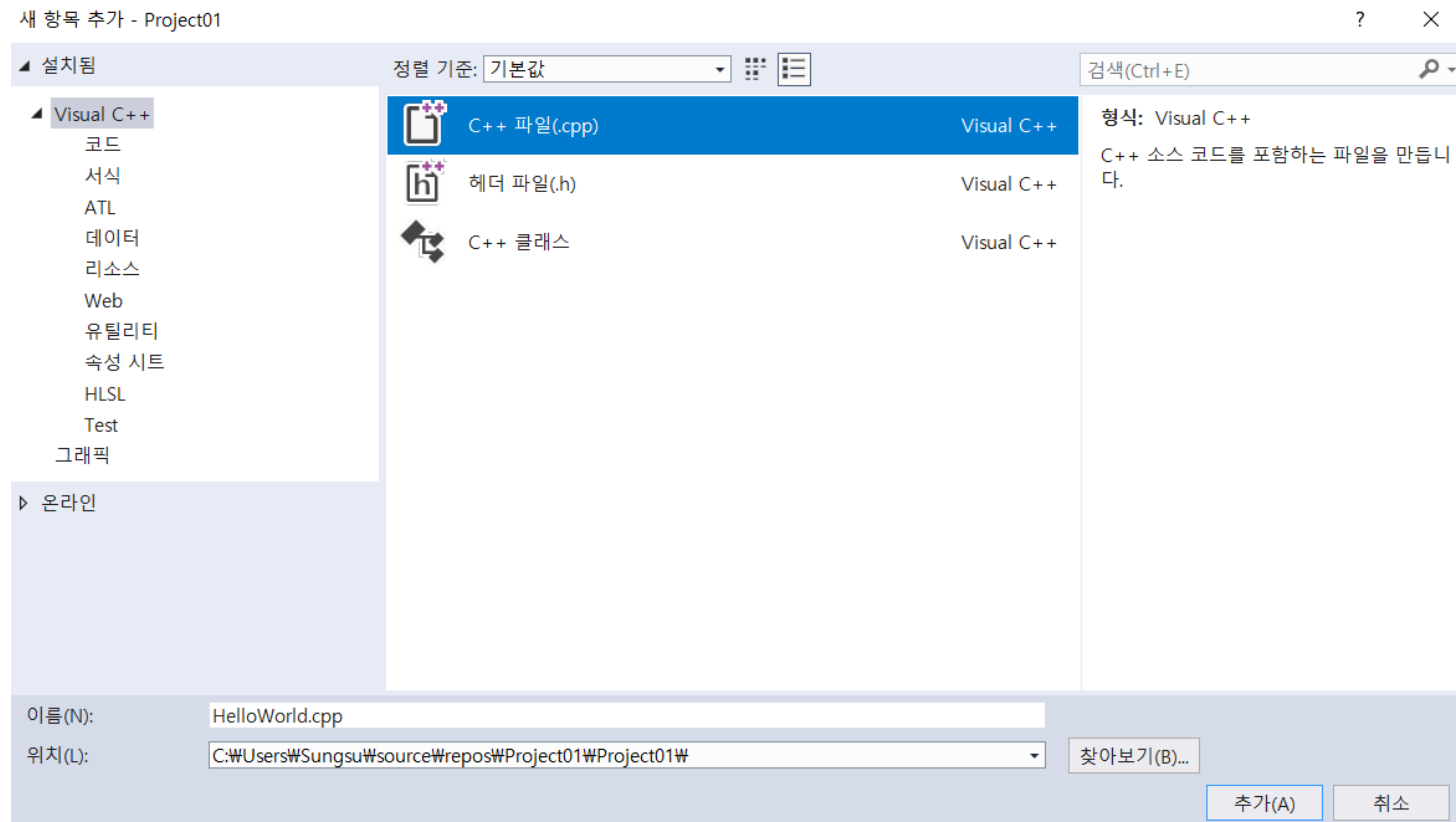
확인

취소

# 프로그램 작성 - 소스 파일 추가



➤ [소스 파일(우클릭)] → [추가] → [새 항목]



# 프로그램 작성 - 소스 파일 작성



➤ [소스 파일(우클릭)] → [추가] → [새 항목]

```
// Project01.cpp
```

```
#include "pch.h "
```

```
#include <iostream>
```

```
int main()
```

```
{
```

```
    std::cout << "Hello World " << std::endl;
```

```
    return 0;
```

```
}
```

```
// C는 c, C++는 cpp
```

```
// 미리 컴파일된 헤더
```

```
// 입출력을 위한 헤더
```

```
// 메인 함수, std: 네임스페이스
```

```
// cout: 콘솔 출력 담당 객체
```


# 프로그램 작성 - 실행



## ➤ 빌드: [빌드] → [컴파일]

```
출력
출력 보기 선택(S): 빌드
1>----- 빌드 시작: 프로젝트: Project01, 구성: Debug Win32 -----
1>Project01.cpp
===== 빌드: 성공 1, 실패 0, 최신 0, 생략 0 =====
```

## ➤ 실행: [디버그] → [디버그하지 않고 시작]

 Microsoft Visual Studio 디버그 콘솔

```
Hello World!
```



# C와 C++의 차이 - 객체

## C 프로그램

- 출력: `printf("%s\\n", "Hello, World!");`
- `printf()` 함수를 이용하여 문자열을 출력
- `printf()` 함수의 사용 방법, 형식 등에 대해서 알아야 함

## C++ 프로그램

- 출력: `std::cout << "Hello, World! " << std::endl;`
- `std` 네임스페이스에 속한 `cout` 객체에 문자열을 넘겨 출력
- 절차상의 흐름 하나 하나를 신경쓰기 보다는 전문 객체에게 맡김
- **객체지향 설계**: 적절한 객체를 생성하고 효율적으로 활용

# 인스턴스와 입/출력



```
#include "pch.h"
#include <string>
#include <iostream>

int main()
{
    int nAge;
    std::cout << "나이 입력" << std::endl;
    std::cin >> nAge;

    char szJob[32];
    std::cout << "직업 입력" << std::endl;
    std::cin >> szJob;
```

```
std::string strName;
std::cout << "이름 입력" << std::endl;
std::cin >> strName;

std::cout << "내 이름은 " << strName
<< "이고, " << "나이는 " << nAge
<< "살이며, " << "직업은 " << szJob
<< "입니다." << std::endl;

return 0;
}
```

Microsoft Visual Studio 디버그 콘솔

```
나이를 입력하세요.
31
직업을 입력하세요.
축구선수
이름을 입력하세요.
메시
내 이름은 메시이고, 나이는 31살이며, 직업은 축구선수입니다.
```



# 향후 실습

## C++ 기초 문법

- Java / C에서 C++로 문법 전환하기

## 이론 수업 내용 실습

- 객체지향 프로그래밍 주제 별 실습

## 객체지향 설계 실습

- 실습 과제 및 학기말 과제를 통한 구현

# 질문 및 답변

