

06. 생성자와 소멸자 그리고 분할 컴파일

실습내용

■ Practice#6 : improve the Grade Management Program #5

▪ **Filename:** PracticeAssignment#6_2022199002_HongGildong

- Convert structures to classes
- Add functions to the Subject class
- Add constructors and destructors
- Apply separate compilation

Practice #6-1: Class Implementation (1)

- As shown in the diagram, proceed with class implementation.

- Member variables of Subject: string SubName, int Hakjum, string Grade, float GPA
- Additional member functions for Subject:

```
main.cpp
└── class Subject
    ├── 멤버 변수
    └── 멤버 함수
└── class Student
    ├── 멤버 변수
    └── 멤버 함수
└── 전역(외부) 함수
    ├── PrintOneData
    ├── PrintAllData
    ├── StdSearch
    ├── PrintMenu
    ├── PrintAllStdList
    ├── ModifyStdInfo
    └── main
```

함수명	설명
Subject()	기본 생성자, 멤버 변수 초기화
void InputData()	과목명, 학점, 등급을 사용자로부터 입력받음
void CalcGPA()	등급에 따라 GPA 계산
void PrintData() const	과목 정보를 한 줄로 출력 (const = 수정 X)
float GetGPA() const	GPA 값을 반환 (const = 수정 X)

- Subject(): Constructor for Subject, initializes member variables(specified by the student).
- InputData(): Function to receive input for subject name, credits, and grade (must be added)
- CalcGPA(): Replaces the function CalcGPA(Subject& sub) in the Student class (Do not delete the original CalcGPA(Subject&) function)
- GetGPA(): Returns the value of the GPA member variable in the Subject class

Practice #6-1: Class Implementation (2)

■ As shown in the diagram, proceed with class implementation.

- Member variables of Student: string StdName, int Hakbun, int SubNum, Subject* Sub, float AveGPA
- Member functions of Student:

main.cpp

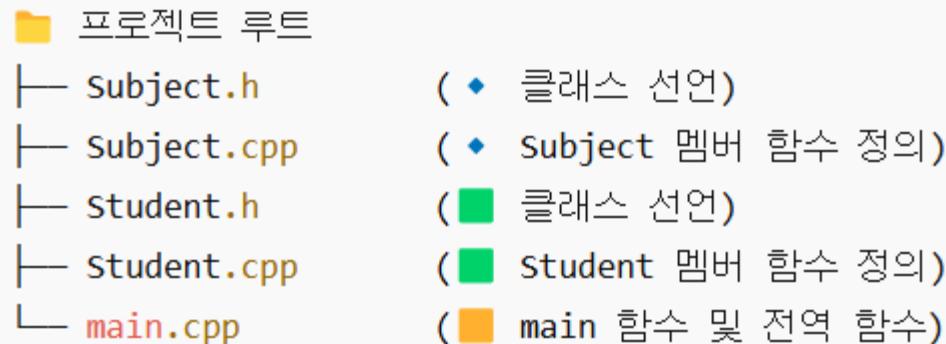
- class Subject
 - 멤버 변수
 - 멤버 함수
- class Student
 - 멤버 변수
 - 멤버 함수
- 전역(외부) 함수
 - PrintOneData
 - PrintAllData
 - StdSearch
 - PrintMenu
 - PrintAllStdList
 - ModifyStdInfo
- main

함수명	설명
Student()	기본 생성자, 멤버 변수 초기화
~Student()	소멸자, Sub 배열 메모리 해제
void CalcGPA(Subject& sub)	Subject 객체의 CalcGPA() 호출
float CalcAveGPA(Subject* sub, int SubNum)	모든 과목 GPA 평균 계산
void InputData(Student* pSt, int Student_Num)	학생 데이터 일괄 입력
inline void InputValue(int& i)	사용자로부터 int 입력받고 cin.ignore() 처리
inline void InputValue(string& str)	사용자로부터 string 한 줄 입력받음

- Student(): Constructor that initializes member variables with values(specified by the student).
- ~Student(): Destructor that deletes dynamically allocated Subject objects

Practice #6-2: Separate Compilation

- As shown in the diagram, divide the



- Make sure to verify the inclusion relationships (#include) in the diagram
- Prevent header file duplication using header guards (#pragma once **or** #ifndef SUBJECT_H... #endif)

