

## 07. 접근지정자, const 객체, static 멤버

# 실습내용

## ■ 실습#7 : 성적처리프로그램#6 개선

- 파일명 : 실습과제#7\_2022199002\_홍길동
  - 멤버변수 캡슐화 (private)
  - 캡슐화에 따른 접근자 함수 및 수정자 함수 생성
  - 학생수를 카운팅 할 수 있는 기능추가(static 변수 생성)

## ■ 중간고사 과제안내

- 평가방법
  - 실습#7로 평가
  - 퀴즈(10%)는 과제 제출 여부 및 코드 정상 작동으로 평가
  - 과제(20%)는 2번의 평가 실시(각 10% 반영)되며, 기능구현, 활용도, 완성도로 평가

## 실습#7-1 : 캡슐화(1)

- 다음 그림과 같이 캡슐화를 진행한다
- Subject의 접근자 함수는 다음과 같음(구현되어 있지 않다면 만들 것)

```
class Subject
{
private:
    SubName : string
    Hakjum  : int
    Grade   : string
    GPA     : float
public:
    Subject()
    void InputData()
    void CalcGPA()
    void PrintData() const
    float GetGPA() const
}
```

- 변수들의 캡슐화 진행
- float GetGPA() const : GPA를 반환하는 접근자 함수

## 실습#7-2 : 캡슐화(2)

- 다음 그림과 같이 캡슐화 및 접근자와 수정자 함수를 생성한다

```
class Student
{
private:
    StdName      : string
    Hakbun       : int
    SubNum       : int
    Sub          : Subject*
    AveGPA       : float

public:
    Student()
    ~Student()
    void InputData(Student*, int)
    void CalcGPA(Subject&)
    float CalcAveGPA(Subject*, int)
    void InputValue(int&)
    void InputValue(string&)
    string GetStdName() const
    int GetHakbun() const
    int GetSubNum() const
    Subject* GetSubjects() const
    float GetAveGPA() const
    void SetStdName(const string&)
    void SetHakbun(int)
```

- 변수의 캡슐화
- 캡슐화된 변수에 접근하기 위한 접근자 함수와 수정자 함수 생성
- 접근자에서 const를 사용하는 이유는 이 함수에서 값을 변경하지 않으려는 의도
- 수정자에서 const를 사용하는 이유는 넘겨받은 인자를 수정하지 않기 위함

### ★ Student 클래스 - 접근자 & 수정자 함수 요약

함수 이름	유형	반환형	설명
GetStdName() const	접근자	string	학생의 이름 반환 (출력, 검색 시 사용)
GetHakbun() const	접근자	int	학생의 학번 반환 (출력, 목록 표시 시 사용)
GetSubNum() const	접근자	int	수강 과목 수 반환 (반복 출력 시 사용)
GetSubjects() const	접근자	Subject*	과목 배열 반환 (출력 시 사용)
GetAveGPA() const	접근자	float	평균 평점 반환 (출력 시 사용)
SetStdName(const string&)	수정자	void	학생 이름 변경 (InputData, ModifyStdInfo 사용)
SetHakbun(int)	수정자	void	학생 학번 변경 (InputData, ModifyStdInfo 사용)

## 실습#7-3 : static 변수 및 함수 생성

### ■ static 멤버를 활용해 본다.

- 전체 학생 수 확인을 위한 기능 추가 (7.번)
- Student 클래스에 static 변수 studentCount를 선언 (**static** 변수의 접근지정자는 **private**)
- 메뉴에서 7번을 입력하면, 현재 총 학생수가 출력되게 할 것

```
===== 메뉴 =====
1. 학생 성적 입력
2. 전체 학생 성적 보기
3. 학생 이름으로 검색
4. 전체 학생 목록 보기
5. 학생 정보 수정
6. 종료
7. 전체 학생 수 확인
선택 : 7
현재 등록된 학생 수 : 1명
```