

[Lab #02]

고급C++프로그래밍 2주차

-영화 클래스, 사각형 클래스 작성-

학과: 소프트웨어융합학과

학번: 2020111694

이름: 이기연

1) 영화 클래스 작성

1. 소스코드

```
#include <iostream>
#include <string>
using namespace std;

class Movie{
private:
    string title;
    string director;
    string actors;
    string grade;
public:
    // setter
    void setTitle(string t);
    void setDirector(string d);
    void setActors(string a);
    void setGrade(string g);

    //getter
    string getTitle();
    string getDirector();
    string getActors();
    string getGrade();
};

// setter body 구현
void Movie::setTitle(string t)
{
    title = t;
}
void Movie::setDirector(string d)
{
    director = d;
}
void Movie::setActors(string a)
{
    actors = a;
}
void Movie::setGrade(string g)
{
    grade = g;
}
```

```

// getter body 구현
string Movie::getTitle()
{
    return title;
}
string Movie::getDirector()
{
    return director;
}

string Movie::getActors()
{
    return actors;
}

string Movie::getGrade()
{
    return grade;
}

int main() {
    Movie mv;

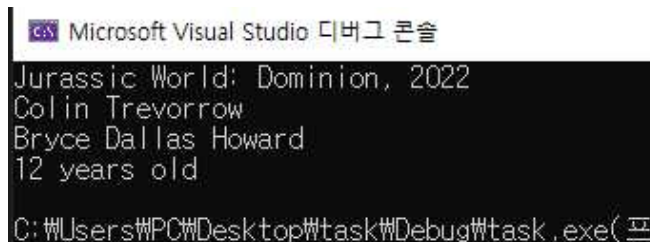
    mv.setTitle("Jurassic World: Dominion, 2022 ");
    mv.setDirector("Colin Trevorrow ");
    mv.setActors("Bryce Dallas Howard ");
    mv.setGrade("12 years old");

    cout << mv.getTitle() << endl;
    cout << mv.getDirector() << endl;
    cout << mv.getActors() << endl;
    cout << mv.getGrade() << endl;

    return 0;
}

```

2. 실행 결과 화면



Microsoft Visual Studio 디버그 콘솔

```

Jurassic World: Dominion, 2022
Colin Trevorrow
Bryce Dallas Howard
12 years old
C:\Users\PC\Desktop\task\Debug\task.exe(프

```

3. 문제 정의 및 분석

- 문제의 정의

- Movie라는 클래스에서 만든 setter와 getter를 이용해 main 함수에서 제목, 감독, 배우, 그리고 상영등급을 입력하고 출력할 수 있도록 하는 프로그램을 만들었다.

- 필요 사항(이 프로그램을 작성하기 위해 사용되는 개념에 대하여)

- Movie의 멤버 변수가 private으로 선언되었기 때문에, 이를 외부 함수에서 사용할 수 있도록 하기 위해 접근자와 설정자 함수를 만들도록 한다.
- 접근자와 설정자 함수를 만드는 과정에서, 클래스의 과도한 집중을 막기 위해 멤버 함수를 외부 정의하여 함수의 header와 body를 분리한다.
- main함수에서 Movie클래스를 mv로 새롭게 정의 후, 점 연산자(.)를 이용해 멤버 함수에 접근하여 쓰기, 읽기를 진행한다.

- 기타 소감

- 멤버 함수를 외부정의하는 과정에서, 작성되고 있는 영역은 클래스의 외부이다. 하지만 작성하고 있는 멤버 함수는 클래스 내에 정의되어 있고, 이를 영역 지정 연산자(::)를 이용해 작성하고 있는 것이기 때문에 private으로 선언된 멤버 변수를 그대로 이용할 수 있다.

2) 사각형 클래스 작성

1. 소스코드

```
#include <iostream>
using namespace std;

class Rectangle
{
private:
    int width;
    int height;
    int area;

public:
    //setter
    void setWidth(int w);
    void setHeight(int h);
```

```

    //getter
    int getWidth();
    int getHeight();
    int getArea();
};

// setter body 구현
void Rectangle::setWidth(int w)
{
    width = w;
}

void Rectangle::setHeight(int h)
{
    height = h;
}

//getter body 구현
int Rectangle::getWidth()
{
    return width;
}

int Rectangle::getHeight()
{
    return height;
}

int Rectangle::getArea()
{
    area = width * height;
    return area;
}

int main()
{
    Rectangle r1;

    r1.setHeight(3);
    r1.setWidth(7);

    cout << "Width = " << r1.getWidth() << endl;
    cout << "Height = " << r1.getHeight() << endl;
}

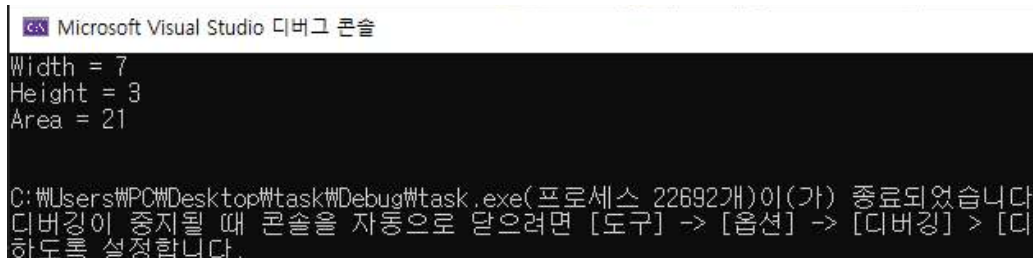
```

```

    cout << "Area = " << r1.getArea() << endl << endl;
    return 0;
}

```

2. 실행 결과 화면



```

Microsoft Visual Studio 디버그 콘솔
Width = 7
Height = 3
Area = 21

C:\Users\PC\Desktop\Task\Debug\task.exe(프로세스 22692개)이(가) 종료되었습니다.
디버깅이 중지될 때 콘솔을 자동으로 닫으려면 [도구] -> [옵션] -> [디버깅] > [디
버깅 콘솔] 하도록 설정합니다.

```

3. 문제 정의 및 분석

- 문제의 정의(지금 어떤 프로그램을 하는 것인지에 대하여)

- Rectangle 이라고 하는 클래스에서 만든 setter와 getter 함수를 이용하여 main 함수에서 width와 height를 입력받아 width, height, area를 콘솔에 출력하는 프로그램을 만든다.

- 필요 사항(이 프로그램을 작성하기 위해 사용되는 개념에 대하여)

- Rectangle의 멤버 변수가 private으로 선언되었기 때문에, 이를 외부에서 이용하기 위해서는 public으로 정의된 setter와 getter라고 하는 설정자, 접근자 멤버 함수를 새롭게 만들어 외부에서도 이용할 수 있도록 해야 한다.
- setter와 getter 함수를 만드는 과정에서, 클래스의 과도한 집중을 막기 위해 멤버 함수를 외부 정의하여 함수의 header와 body를 분리한다.
- main함수에서 Rectangle 클래스를 r1으로 새롭게 정의 후, 점 연산자(.)를 이용해 멤버 함수에 접근하여 쓰기, 읽기를 진행한다.

- 기타 소감(기록)

- area를 구하기 위해서는 width와 height를 이용하면 되므로, 값 설정을 위한 setter 함수를 만들 필요는 없다. 다만 이후 콘솔 출력을 위해 값을 리턴하는 getter 함수는 만들어야 한다.