

[lab11 보고서]

자율전공학과

2019111677

김지연

1. 소스코드

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

class Shape {
private:
    double height;
    double width;
public:
    Shape(double h, double w) {
        height = h;
        width = w;
    }
    void setHeight(double h) {
        height = h;
    }
    void setWidth(double w) {
        width = w;
    }
    double getHeight() {
        return height;
    }
    double getWidth() {
        return width;
    }
    double getArea() {
        return 0.0;
    }
};

class Rectangle : public Shape
{
public:
    Rectangle(double h, double w);
    ~Rectangle() {};

    double getArea() {
        return getHeight() * getWidth();
    }
};

Rectangle::Rectangle(double h, double w) :Shape(h, w) {}

class Triangle : public Shape
{
public:
    Triangle(double h, double w);
    ~Triangle() {};

    double getArea() {
        return getHeight() * getWidth() / 2;
    }
};
```

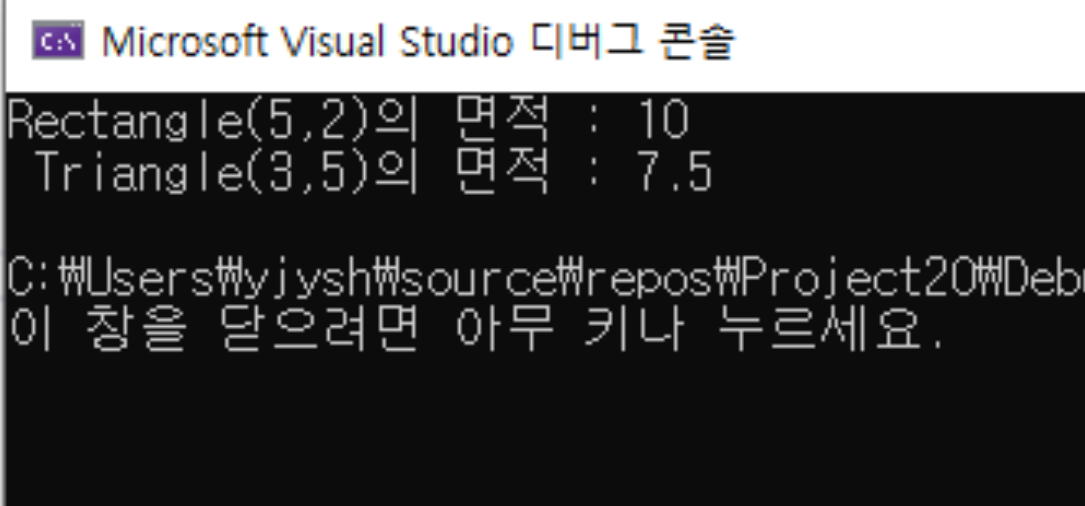
```

Triangle::Triangle(double h, double w) :Shape(h, w) {}

int main(){
    Rectangle r(5, 2);
    Triangle t(3, 5);
    cout << "Rectangle(5,2)의 면적 : " << r.getArea() << endl;
    cout << " Triangle(3,5)의 면적 : " << t.getArea() << endl;
    return 0;
}

```

2. 실행결과화면



```

C:\> Microsoft Visual Studio 디버그 콘솔

Rectangle(5,2)의 면적 : 10
Triangle(3,5)의 면적 : 7.5

C:\Users\jysh\source\repos\Project20\Deb
이 창을 닫으려면 아무 키나 누르세요.

```

3. 문제 정의 및 분석

사각형과 삼각형의 면적을 계산하는 문제

자식클래스에서 필요한 것은 가로와 세로 그리고 면적인데 중복되는 코드가 부모클래스에 있기 때문에 상속받아 사용한다.

면적은 Area의 계산식을 작성하여 반환값으로 받아준다.

가로와 세로는 상속받기 위하여 자식클래스의 생성자 헤더의 뒤에 콜론을 추가하여 부모클래스의 생성자를 적어준다.

1. 소스코드

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

class Animal
{
private:
    int age;
    int weight;
    string type;
public:
    Animal() {}
    Animal(int age, int weight, string type)
    {
        this->age = age;
        this->weight = weight;
        this->type = type;
    }
    void eat();
    void sleep();
    int getAge();
    void setAge(int age);
    int getWeight();
    void setWeight(int weight);
    string getType();
    void setType(string type);
    void showAnimal();
};

void Animal::eat() { cout << "Animal::eat()" << endl; }
void Animal::sleep() { cout << "Animal::sleep()" << endl; }
int Animal::getAge() { return age; }
void Animal::setAge(int age) { this->age = age; }
int Animal::getWeight() { return weight; }
void Animal::setWeight(int weight) { this->weight = weight; }
string Animal::getType() { return type; }
void Animal::setType(string type) { this->type = type; }

void Animal::showAnimal() {
    cout << "Age = " << age << endl;
    cout << "Weight = " << weight << endl;
    cout << "Type = " << type << endl;
}

class Cat : public Animal {
private:
    int tail_length;
public:
    Cat() {}
    Cat(int age = 0, int weight = 0, string type = " ", int t = 0);
    void Meow();
    int getTailLength();
    void setTailLength(int tail_length);
    void showCat();
};
```

```

};

Cat::Cat(int age, int weight, string type, int t) :Animal(age, weight, type), tail_length(t)
{};

int Cat::getTailLength() { return tail_length; }
void Cat::setTailLength(int tail_length) { this->tail_length = tail_length; }
void Cat::Meow() {};

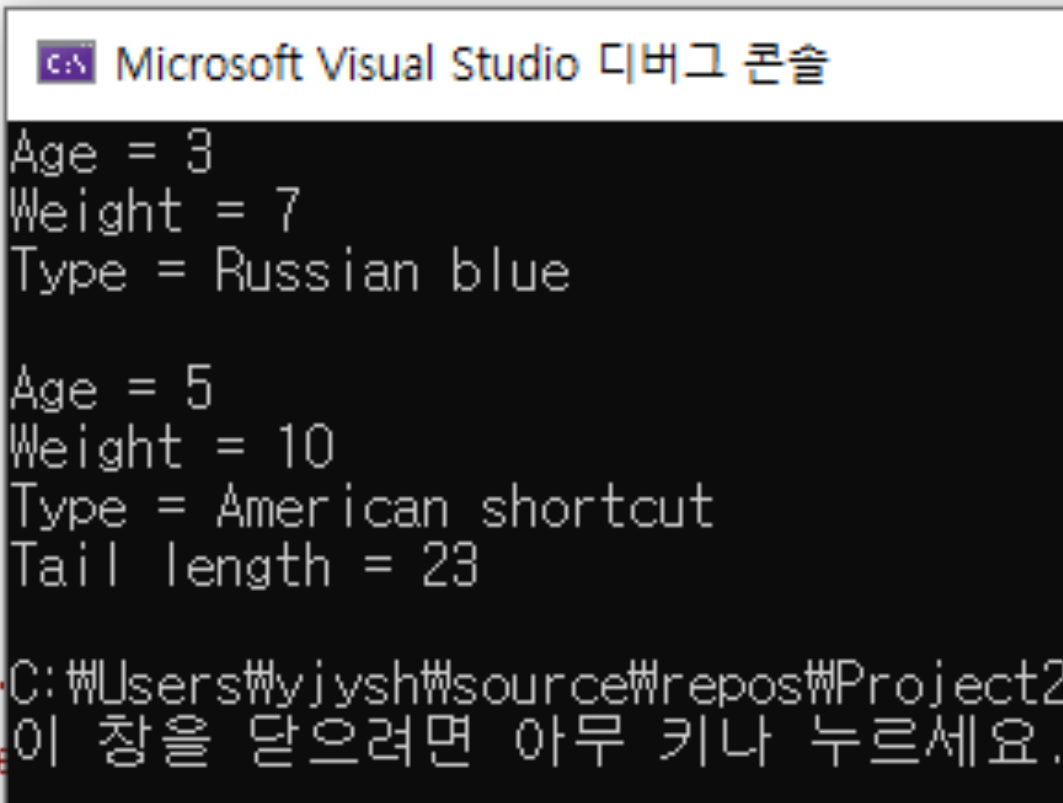
void Cat::showCat(){
    showAnimal();
    if (tail_length) {
        cout << "Tail length = " << getTailLength() << endl;
    }
};

void main()
{
    Cat frisky(3, 7, "Russian blue");
    frisky.showCat();
    cout << endl;

    Cat oreo(5, 10, "American shortcut", 23);
    oreo.showCat();
}

```

2. 실행결과화면



The screenshot shows the Microsoft Visual Studio Debug Console window. The title bar reads "C:\N Microsoft Visual Studio 디버그 콘솔". The console output displays the following information:

```

Age = 3
Weight = 7
Type = Russian blue

Age = 5
Weight = 10
Type = American shortcut
Tail length = 23

```

At the bottom of the console, the file path "C:\Users\myjysh\source\repos\Project2" is visible, followed by a red error message in Korean: "이 창을 닫으려면 아무 키나 누르세요." (Press any key to close this window).

3. 문제 정의 및 분석

Cat(자식클래스)은 나이, 무게, 종, 꼬리의 길이를 출력하는 클래스인데 그중 나이, 무게, 종은 부모클래스(Animal class)에 정의되어 있으므로 상속받아 사용하고 꼬리의 길이만 자식클래스에서 다시 정의해준다.

Tail_length의 함수의 정의를 적어준다.

자식클래스에 생성자를 만들어 주고 이 생성자가 Animal클래스를 상속받을 수 있도록 콜론을 붙여 Animal클래스의 생성자를 적어준다.

메인함수에서 showCat을 출력하고 있는데 showCat에 필요한 내용과 showAnimal이 가지고 있는 내용이 겹치므로 showCat함수에 showAnimal을 출력해준다.

Tail_length는 showAnimal에 있지 않으므로 if문을 사용하여 꼬리에 대한 내용이 있으면 출력하고 없으면 출력하지 않도록 만들어 준다.