

# C++ 과제5

동물원 관리 시스템 만들기 문제

로봇게임단

19기 예비인턴

이원준

## 1. 코드

- 헤더 파일

```
#ifndef ANIMAL_H
```

```
#define ANIMAL_H
```

```
using namespace std;
```

```
#include <iostream>
```

```
#include <string>
```

```
class Animal{
```

```
protected:
```

```
    std::string name;
```

```
    int age;
```

```
public:
```

```
    Animal(std::string name, int age);
```

```
    virtual void makeSound(){};
```

```

        void showinfo();

};

Animal::Animal(std::string name, int age) { this->name = name; this->age = age; };

void Animal::showinfo()
{
    cout << "name: " << name << endl;
    cout << "age: " << age << endl;
}

class Dog : public Animal{

public:
    Dog(std::string name, int age) : Animal(name,age) {};
    virtual void makesound(){
        cout << "Bark" << endl;
    }
};

class Cat : public Animal{

public:
    Cat(std::string name, int age) : Animal(name,age) {};
    virtual void makesound(){
        cout << "Meow" << endl;
    }
};

```

```

    }

};

class Bird : public Animal{
public:
    Bird(std::string name, int age) : Animal(name,age) {};

    virtual void makesound(){
        cout << "Chrip" << endl;
    }
};

#endif // ANIMAL_H

```

- 메인 코드

```

#include "Animal.h"

int main()
{
    Dog* newdog = new Dog("dog1", 4);
    newdog->showinfo();
    newdog->makesound();

    delete newdog;
}

```

```
Cat* newcat = new Cat("Cat1", 7);  
  
newcat->showinfo();  
  
newcat->makesound();  
  
  
delete newcat;  
  
  
Bird* newbird = new Bird("Bird1", 1);  
  
newbird->showinfo();  
  
newbird->makesound();  
  
  
delete newbird;  
  
}
```

## 2. 코드 설명

### 2-1. 헤더 파일 설명

- Animal 클래스를 상속 시켜서 효율적으로 하위 클래스의 공통 구성 요소를 만들었다.
- Animal 클래스의 생성자를 구현할 때, this를 사용하여 코드 구현시 충돌을 방지하게 만들었다.

```
Animal::Animal(std::string name, int age) { this->name = name; this->age = age; };
```

## 2-2. 메인 코드 설명

- 클래스의 멤버 변수를 초기화 시키고 출력하는 모습을 사용자가 확인할 수 있게 만들었다.

## 3. 느낀점 및 고찰

- 클래스 설정을 할 때 클래스 멤버 함수를 단순히 선언만 하고 재정의하지 않으면

코드가 돌아가지 않는데, 이러한 현상을 컴파일러에서 잡아주지 못하기 때문에 상당히 디버깅하기 골치 아프다. 클래스 멤버 함수의 재정의 여부를 항상 확인해주자.