

```
1 //=====
2 // Name      : Animal.cpp
3 // Author    : Sotheanith Sok
4 // Version   : 1.0
5 // Description : This is an abstract class contains: virtual destructor, count of ↗
6 //             all object created from Animal, operator<< overload,...etc.
7 //=====
8 #include "Animal.h"
9 #include<iostream>
10
11 //Precondition:
12 // _None.
13 //Postcondition:
14 // _This variable created on the number of Animal existed.
15 int Animal::count = 0;
16
17 //Precondition:
18 // _None.
19 //Postcondition:
20 // _Virtual destructor used to deallocated the memory used to initilize variable.
21 Animal::~Animal()
22 {
23     delete animalType;
24     count--;
25 }
26
27 //Precondition:
28 // _None.
29 //Postcondition:
30 // _Return the name of this animal.
31 std::string Animal::getAnimalType()
32 {
33     return "["+*animalType+"]";
34 }
35
36 //Precondition:
37 // _None.
38 //Postcondition:
39 // _Return the number of this object existed.
40 int Animal::getCount()
41 {
42     return count;
43 }
44
45 //Precondition:
46 // _None.
47 //Postcondition:
48 // _Return os contains animalType, animal's talk, animal's move.
49 std::ostream & operator<<(std::ostream & os, Animal & obj)
50 {
51     os << *(obj.animalType) << ", " << obj.talk() << ", " << obj.move();
```

```
52     return os;  
53 }  
54
```