Question: Consider the following class definitions:

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

class Animal {
    int limbs;
    string eatinghabits;
    public:
        Animal(int limbs_, string eh): eatinghabits(eh){limbs=limbs_;}
        int getLimbs() const { return limbs; }
        string getEatingHabits() const {return eatinghabits; }
        void print() {
            cout << "Animal's print function" << endl;
            cout << "It is a " << eatinghabits << endl;
            cout << "It has " << limbs << " limbs" << endl;
        }
};

class Cat: public Animal {
    string breed;
    string color;
    public:
        Cat(string color, string breed): Animal(4, "carnivorous"),
            breed(breed), color(color) {} 
        void print() {
            cout << "Cat's print function" << endl;
            cout << "It is a " << getEatingHabits() << endl;
            cout << "It has " << getLimbs() << " limbs" << endl;
        }
        void printBreed() {cout << "The Cat's breed is: " << breed << endl;}
};

class Rabbit: public Animal {
    string type;
    string color;
    public:
        Rabbit(string type, string color): Animal(4, "herbivorous"),
            type(type), color(color) {} 
        void print() {
            cout << "Rabbit's print function" << endl;
            cout << "It is a " << getEatingHabits() << endl;
            cout << "It has " << getLimbs() << " limbs" << endl;
            cout << "It is a " << type << endl;
            cout << "Its color is " << color << endl;
        }
};
```
We are providing you two `main()` functions. For each of these, indicate whether the `main()` will run correctly. If so, what would be the output? And if not, what type of error (compile time, run time, or logical) will be encountered?

<table>
<thead>
<tr>
<th>Function</th>
<th>Run Correctly? (Yes/No)</th>
<th>What kind of error(s)?</th>
<th>Reason</th>
<th>Output</th>
</tr>
</thead>
</table>
| `int main()` { 
    Cat kitty("persian","white");  
    Cat &myCat=kitty;  
    Rabbit bunny("wild","white");  
    Rabbit &rabbit=bunny;  
    Animal &kitten=kitty;  
    kitten.printBreed();  
    return 0; } | No | Compile Time | kitten is an Animal& and the printBreed() method is not declared/defined for Animal. In other words, static type of kitten is Animal which does not implement the method printBreed(), hence the 2nd last statement in main fails. | N/A |
| `int main()` { 
    Cat* myCat;  
    Rabbit *rabbit;  
    Animal* animal;  
    myCat=new Cat("siamese","brown");  
    rabbit=new Rabbit("wild","white");  
    animal=myCat;  
    animal->print();  
    rabbit->print();  
    return 0; } | Yes | N/A | N/A | Animal's print function  
It is a carnivorous  
It has 4 limbs  
Rabbit's print function  
It is a herbivorous  
It has 4 limbs  
It is a wild  
Its color is white |