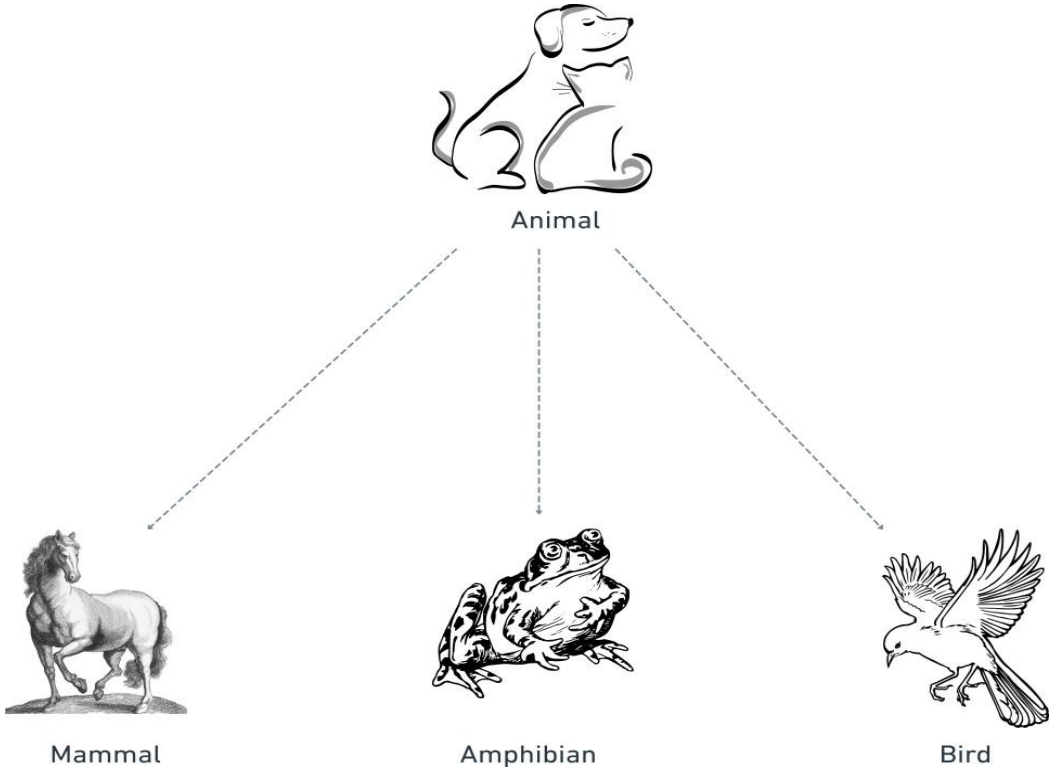
Real-world class modeling

Realworld class modeling example:-

Let's take a real-world example of Animal as a class to understand concepts better. And Mammals, birds, and amphibians as objects of the class.




```
#include <iostream>
Using namespace std;
// creating Animal class
class Animal{
        bool gives_birth;
        bool lay_egg;
```

```
        bool live_in_ground;
        bool live_in_water;
        bool have_wings;
};
int main(){
        // creating an object of animal class
        Animal mammal;

        //assign values to instance variables
        mammal.gives_birth = true;
        mammal.lay_egg = false;
        mammal.live_in_ground = true;
        mammal.live_in_water = false;
        mammal.have_wings = false;

        Animal amphibian;
        amphibian.gives_birth = false;
        amphibian.lay_egg = true;
        amphibian.live_in_ground = true;
        amphibian.live_in_water = true;
        amphibian.have_wings = false;

        Animal bird;
        bird.gives_birth = false;
        bird.lay_egg = true;
        bird.live_in_ground = true;
        bird.live_in_water = false;
        bird.have_wings = true;
}
```

Here we create a class Animal and definesome animal characters (properties) that may be shared for different kinds of animals. We defined all the properties for each object, like whether they give birth or not, whether they live in water, etc.

Here, Animal class provides a template or blueprint for creating objects (mammal, bird, and amphibian).

Why do we need object-oriented programming?

- To make the development and maintenance of projects more effortless.
- To provide the feature of data hiding that is good for security concerns.
- We can solve real-world problems if we are using object-oriented programming.
- It ensures code reusability.
- It lets us write generic code: which will work with a range of data, so we don't have to write basic stuff over and over again.