

```

// Online IDE - Code Editor, Compiler, Interpreter

#include<iostream>

using namespace std;

class Fish
{
public:
    virtual void Swim()
    {
        cout << "The fish swims in the water" << endl;
    }
    virtual ~Fish(){}
};

class Tuna: public Fish
{
public:
    void Swim()
    {
        cout << "The tuna swims really fast into the sea" << endl;
    }
    void BecomeDinner()
    {
        cout << "The tuna has become dinner with shushi" << endl;
    }
};

class Carp: public Fish
{
public:
    void Swim()
    {
        cout << "The carp swims really slow into the lake" << endl;
    }

    void Talk()
    {
        cout << "The carp spoke carpish!" << endl;
    }
};

void DetectFishType(Fish* objFish)
{
    Tuna* objTuna = dynamic_cast <Tuna*>(objFish);
    if (objTuna)
    {
        cout << "Tuna has been found. The dinner will be with tuna: " <<
endl;
        objTuna->BecomeDinner();
    }
    Carp* objCarp = dynamic_cast <Carp*>(objFish);
    if(objCarp)
    {
        cout << "Carp has been found. We make the carp to speak: " << endl;
        objCarp->Talk();
    }
}

```

```
    }  
    cout << "Verifying the form by using virtual Fish::Swim: " << endl;  
    objFish->Swim();  
}  
  
int main()  
{  
    Carp myLunch;  
    Tuna myDinner;  
  
    DetectFishType(&myDinner);  
    cout << endl;  
    DetectFishType(&myLunch);  
  
    return 0;  
}
```