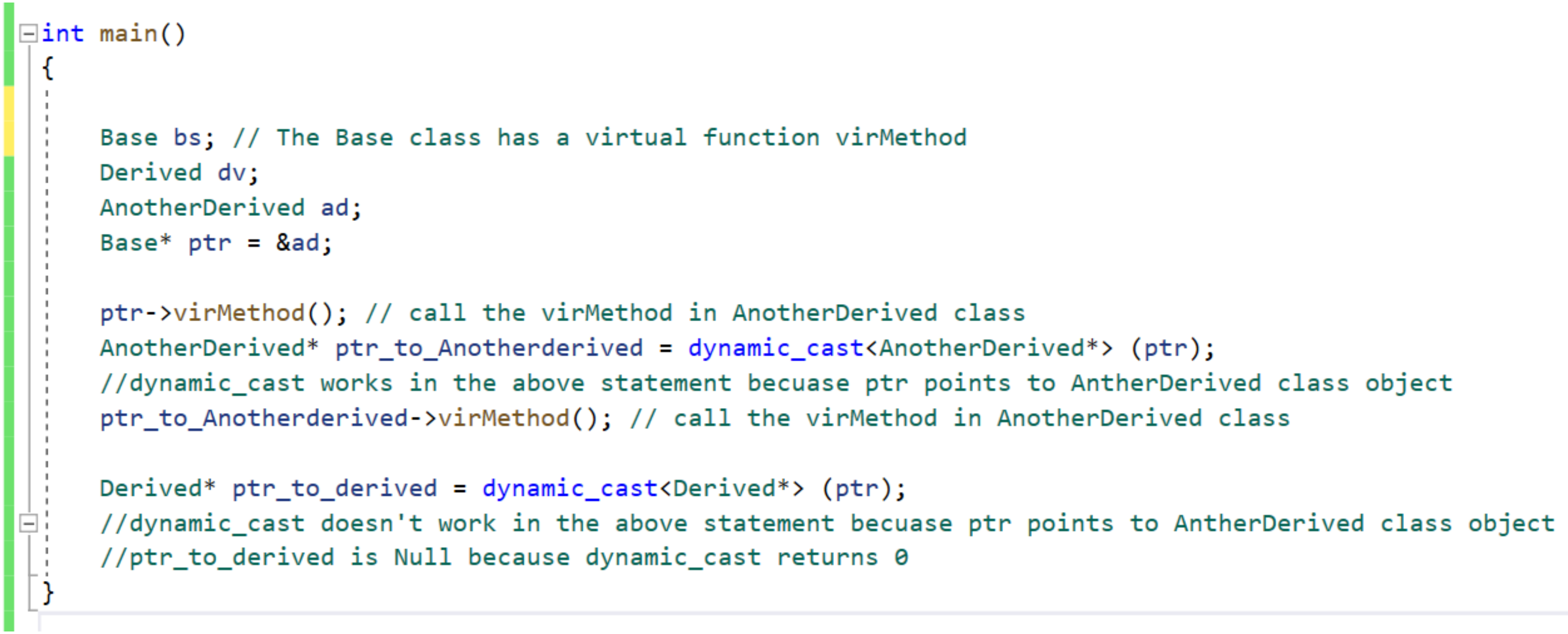
Type Casting

Continue

Example for **dynamic_cast**



```
int main()
{

    Base bs; // The Base class has a virtual function virMethod
    Derived dv;
    AnotherDerived ad;
    Base* ptr = &ad;

    ptr->virMethod(); // call the virMethod in AnotherDerived class
    AnotherDerived* ptr_to_Anotherderived = dynamic_cast<AnotherDerived*> (ptr);
    //dynamic_cast works in the above statement becuase ptr points to AntherDerived class object
    ptr_to_Anotherderived->virMethod(); // call the virMethod in AnotherDerived class

    Derived* ptr_to_derived = dynamic_cast<Derived*> (ptr);
    //dynamic_cast doesn't work in the above statement becuase ptr points to AntherDerived class object
    //ptr_to_derived is Null because dynamic_cast returns 0
}
```

4- reinterpret_cast

This type casting works by reinterpreting the underlying bit pattern, and thus is considered a **dangerous** type casting operator. For example, it can be used to convert a pointer of particular type to another pointer of any type.

reinterpret_cast <new-type> (expression)

Example) char *ptr = reinterpret_cast <char *> (65); //caution!

//the above statement converts an int number to a memory address. No error but "unable to read memory" at runtime

int num = reinterpret_cast <int> (ptr); //num is equal to 65