https://youtu.be/LVWK1aJiN40

Inherited Access Specifiers

Just as access specifiers (i.e. `public`, `protected`, and `private`) define which class members *users* can access, the same access modifiers also define which class members *users of a derived classes* can access.

> **Public inheritance:** the public and protected members of the base class listed after the specifier keep their member access in the derived class
>
> **Protected inheritance:** the public and protected members of the base class listed after the specifier are protected members of the derived class
>
> **Private inheritance:** the public and protected members of the base class listed after the specifier are private members of the derived class
>
> Source: **C++ reference**

In the exercise below, you'll experiment with access modifiers.

Instructions

1. Update the derived classes so that one has `protected` inheritance and one has `private` inheritance.
2. Try to access a `protected` member from `main()`. Is it possible?
3. Try to access a `private` member from `main()`. Is it possible?
4. Try to access a member of the base class from within the derived class that has `protected` inheritance. Is it possible?
5. Try to access a member of the base class from within the derived class that has `private` inheritance. Is it possible?