4.5 Nested Interfaces

➤ An interface can be declared a member of a class or another interface.
➤ Such an interface is called a *member interface* or a *nested interface*.
➤ A nested interface can be declared as **public**, **private**, or **protected**.
➤ This differs from a top-level interface, which must either be declared as **public** or use the default access level, as previously described.
➤ When a nested interface is used outside of its enclosing scope, it must be qualified by the name of the class or interface of which it is a member.
➤ Thus, outside of the class or interface in which a nested interface is declared, its name must be fully qualified.
➤ Interface inside Interface **Syntax:**

```
interface interface_name
{

        interface nested_interface_name
        {

        }
}
```