**Polymorphism** is a core concept in object-oriented programming. It allows methods to act differently based on the object that is calling them. Here is an example of polymorphism in C# using the keywords **virtual**, **override**, and **new**.

In C#, **virtual** is used to modify a method, property, indexer, or event declaration and allow it to be overridden in a derived class. The **override** keyword is used to extend or modify a virtual/abstract method, property, indexer, or event of a base class into the derived class. The **new** keyword can be used to hide a method, property, etc. from the parent class.

Here is an example:

![A picture containing text, menu, screenshot

Description automatically generated]()

In this example, the **Dog** class overrides the **MakeSound()** method using **override**, which changes the behavior of the method when called on a **Dog** object. On the other hand, the **Cat** class hides the base class method using the **new** keyword, but when the object is treated as an **Animal** type, it behaves as an **Animal** rather than a **Cat** which is why it outputs "The animal makes a sound". However, when the object is a **Cat** it behaves as a **Cat**.