Polymorphism is shown when one line of code can have different behavior depending on the context. This is done through overriding a method in the child class that is inherited from a parent class. This means that the name of the method stays the same, but the code is different.

Programmers are faced with code that are duplicated in children’s classes. With Polymorphism it allows shared behaviors to be defined in a base class and then overridden in the children classes when needed, so they don’t have to repeat themselves.

The benefit of polymorphism is ensuring that programs are flexible and ready for change. For example, in the Eternal Quest Program I have a parent class method called public abstract void Display() and in my three child classes I had them display something differently (See screenshots below). Another key word you can use polymorphism is virtual, this allows you to still use the code in the parent class, but if you specify the method as an abstract you have to override the method in the child.

A screen shot of a computer

AI-generated content may be incorrect.

A screen shot of a computer

AI-generated content may be incorrect.

