Polymorphism

Polymorphism is integral to the world modelling aspect of object oriented design via generalisation and specialisation. Using polymorphism allows programmers to write code that is more flexible, extensible and adaptable by giving a common interface for similar objects with inheritance that can be used throughout the code without care of the actual underlying type whilst letting you specialise functionality based on types.

Abstraction

Abstraction is the art of mapping lower-level constructs and implementations to high-level ones that are easier to think about / work with. Object-Oriented Programming attempts to abstract procedural programming into objects that encapsulate both the data and functionality to make it easier to deal with large volumes of code and reduce code duplication with inheritance and automate certain control flow situations with polymorphism and virtual functions. One of the greatest abstractions is the programming language which can abstract away the concept of machine code and the actual CPU instructions needed to perform certain computation with higher-level concepts such as variables, if statements, loops and procedures. Languages like C# even abstract away and automate the computers memory with Garbage Collection allowing the programmer to have to know anything about the memory usage and constraints of hardware. Abstraction isn't just about classification and providing small helpers to make the code easier to write and less error prone, it's about altering how you think about the problem in the first place.