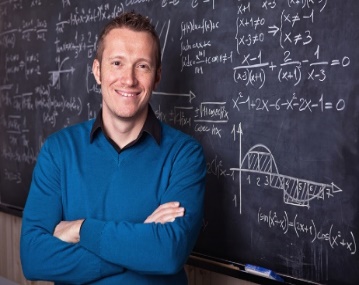
**POLYMORPHISM IN C#**

* Polymorphism is one of the four pillars of Object Oriented Programming.
* Polymorphism in C# is a concept by which we can perform a single action by different ways.
* Polymorphism is derived from 2 Greek words: POLY and MORPHS.
* The word "poly" means many and "morphs" means forms.
* So polymorphism means many forms.

**REAL WORLD EXAMPLE**

FATHER

SON

HUSBAND

TEACHER

**THERE ARE TWO TYPES OF POLYMORPHISM**

1. STATIC POLYMORPHISM (COMPILE TIME POLYMORPHISM)
2. DYNAMIC POLYMORPHISM (RUN TIME POLYMORPHISM)

**STATIC POLYMORPHISM (COMPILE TIME POLYMORPHISM) IN C#**

* The mechanism of linking a function with an object during compile time is called static polymorphism or early binding.
* It is also called static binding.

C# provides two techniques to implement static polymorphism. They are −

* METHOD OR FUNCTION OVERLOADING
* OPERATOR OVERLOADING

**METHOD OR FUNCTION OVERLOADING**

* You can have multiple definitions for the same function name in the same scope.
* The definition of the function must differ from each other by the types and/or the number of arguments in the argument list.
* You cannot overload function declarations that differ only by return type.

**DYNAMIC OR RUNTIME POLYMORPHISM IN C#**

* RUN TIME POLYMORPISM IS ACHIEVED BY METHOD OVERRIDING.
* METHOD OVERRIDING ALLOWS US TO HAVE VIRTUAL AND ABSTRACT METHODS IN THE BASE USING DERIVED CLASSES WITH THE SAME NAME AND THE SAME PARAMETER.

**C# METHOD OVERRIDING**

* If derived class defines same method as defined in its base class, it is known as method overriding in C#.
* It is used to achieve runtime polymorphism.
* It enables you to provide specific implementation of the method in child class which is already provided by its base class.
* To perform method overriding in C#, you need to use **virtual** keyword with base class method and **override** keyword with derived class method.
* A method declared using the virtual keyword is referred to as a virtual method.
* In the derived class, you need to declare the inherited virtual method using the override keyword.
* In the derived class, you need to declare the inherited virtual method using the override keyword which is mandatory for any virtual method that is inherited in the derived class.
* The override keyword overrides the base class method in the derived class.