Polymorphism

The word polymorphism is a combination of two words i.e. **ploy** and **morphs**. The word poly means **many** and morphs means **different forms**. In short, a mechanism by which we can perform a single action in different ways.

Let's understand the meaning of polymorphism with a real-world example.

A person in a shop is a customer, in an office, he is an employee, in the home he is husband/ father/son, in a party he is guest. So, the same person possesses different roles in different places. It is called polymorphism.

53.7M

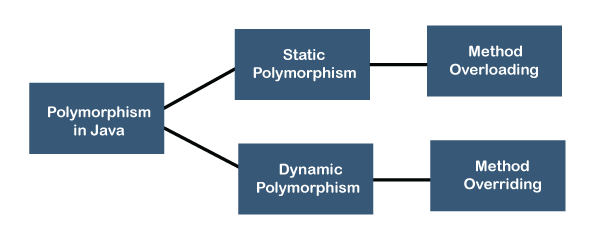
1.1K

OOPs Concepts in Java

Types of Polymorphism

There are two types of polymorphism in [Java](https://www.javatpoint.com/java-tutorial):

* Static Polymorphism (Compile Time Polymorphism)
* Dynamic Polymorphism (Run Time Polymorphism)



Dynamic Polymorphism

**Dynamic polymorphism** is a process or mechanism in which a call to an overridden method is to resolve at runtime rather than compile-time. It is also known as [**runtime polymorphism**](https://www.javatpoint.com/runtime-polymorphism-in-java) or **dynamic method dispatch**. We can achieve dynamic polymorphism by using the [**method overriding**](https://www.javatpoint.com/method-overriding-in-java).

In this process, an overridden method is called through a reference variable of a superclass. The determination of the method to be called is based on the object being referred to by the reference variable.

Properties of Dynamic Polymorphism

* It decides which method is to execute at runtime.
* It can be achieved through dynamic binding.
* It happens between different classes.
* It is required where a subclass object is assigned to a super-class object for dynamic polymorphism.
* Inheritance involved in dynamic polymorphism.