

Distinguish, differentiate, compare and explain what is the Difference between Method Overloading and Overriding. Comparison and Differences.

Method Overloading

Overloading allows us to use functions or methods with the same name, but different arguments. The decision on which function to use (overload resolution) is done by the compiler when the program is compiled. There is NO dynamic binding.

Method Overriding

Overriding allows a derived class to provide a different implementation for a method declared in the base class. It is only possible with inheritance and dynamic binding without inheritance, there is no overriding. The decision which method to use is delayed until runtime and is done at the moment that the method is called. It only applies to member methods, not free functions.

Overriding vs Overloading

Do not confuse overriding a method in a derived class with overloading a method name. When a method is overridden, the new method definition given in the derived class has the exact same number and types of parameters as in the base class. When a method in a derived class has a different signature from the method in the base class, that is overloading. Note that when the derived class overloads the original method, it still inherits the original method from the base class as well.

Difference between Method Overloading and Overriding

1. Method overloading is used to increase the readability of the program. Method overriding is used to provide the specific implementation of the method that is already provided by its superclass.
2. Method overloading is performed within a class. Method overriding occurs in two classes that have the IS-A (inheritance) relationship.
3. In the case of method overloading, the parameter must be different. In the case of method overriding, the parameter must be same.
4. Method signature must be different. Method signature must be the same.
5. Private, static and final methods can be overloaded. Private, static and final methods can not be overridden.