

Quick Links for Java
Recent Articles
MCQ / Quizzes
Java Collections
Practice Problems
Commonly Asked Questions Set 1 & Set 2

Basics
Identifiers, Data types & Variables
Scope of Variables
Operators
Loops and Decision Making
Explore More...
Input / Output
Ways to read Input from Console
Scanner VS BufferedReader Class
Fast I/O in Java in Competitive Programming
Command Line arguments
Explore More...

Arrays
Arrays in Java
Default array values in Java
Compare two arrays
Final Arrays & Jagged Arrays
Array IndexOutOfBounds Exception
Explore More...

Strings
String Class in Java
StringBuffer , StringTokenizer & StringJoiner
Initialize and Compare Strings
String vs StringBuilder vs StringBuffer
Integer to String & String to Integer
Search, Reverse and Split()
Explore More...

OOP in Java
Classes and Objects in Java
Different ways to create objects
Access Modifiers in Java
Object class in Java
Encapsulation & Inheritance
Method Overloading & Overriding
Explore More...

Constructors
Constructors & Constructor Chaining
Constructor Overloading
Private Constructors and Singleton Classes
Explore More...

Methods
Parameter Passing
Returning Multiple Values
Private and Final Methods
Default Methods
Explore More...

Exception Handling
Exceptions & Types of Exceptions
Flow control in try-catch & Multicatch
throw and throws
Explore More...

Multithreading
Multithreading
Lifecycle and States of a Thread
Main Thread
Synchronization
Inter-thread Communication & Java Concurrency
Explore More...

File Handling
File Class
File Permissions
Different ways of Reading a text file
Delete a File
Explore more...

Garbage Collection
Garbage Collection
Mark and Sweep
Explore more...

Java Packages
Packages
Java.io Package
Java.lang package
Java.util Package
Networking
Socket Programming
URL class in Java
Reading from a URL
Inet Address Class
A Group Chat Application
Explore more...

Can we Overload or Override static methods in java ?

WooCommerce

Need eCommerce For WordPress?
WooCommerce is the open source, fully customizable eCommerce platform built for WordPress.

Get Started >

Let us first define Overloading and Overriding.

Overriding : Overriding is a feature of OOP languages like Java that is related to run-time polymorphism. A subclass (or derived class) provides a specific implementation of a method in superclass (or base class).

The implementation to be executed is decided at run-time and decision is made according to the object used for call. Note that signatures of both methods must be same. Refer Overriding in Java for details.

Overloading: Overloading is also a feature of OOP languages like Java that is related to compile time (or static) polymorphism. This feature allows different methods to have same name, but different signatures, especially number of input parameters and type of input paramaters. Note that in both C++ and Java, methods cannot be overloaded according to return type.

Can we overload static methods?

Overloading in Java is possible. For example, consider the following Java program.

```
// filename Test.java
public class Test {
    public static void foo() {
        System.out.println("Test.foo() called ");
    }
    public static void foo(int a) {
        System.out.println("Test.foo(int) called ");
    }
    public static void main(String args[])
    {
        Test.foo();
        Test.foo(10);
    }
}
```

Run on IDE

Output:

```
Test.foo() called
Test.foo(int) called
```

Can we overload methods that differ only by static keyword?

We cannot overload two methods in Java if they differ only by static keyword (number of parameters and types of parameters is same). See following Java program for example. This behaviour is same in C++ (See point 2 of this).

```
// filename Test.java
public class Test {
    public static void foo() {
        System.out.println("Test.foo() called ");
    }
    public void foo() { // Compiler Error: cannot redefine foo()
        System.out.println("Test.foo(int) called ");
    }
    public static void main(String args[]) {
        Test.foo();
    }
}
```

Run on IDE

Output: Compiler Error, cannot redefine foo()

Can we Override static methods in java?

We can declare static methods with same signature in subclass, but it is not considered overriding as there won't be any run-time polymorphism. Hence the answer is 'No'.

If a derived class defines a static method with same signature as a static method in base class, the method in the derived class hides the method in the base class.

```
/* Java program to show that if static method is redefined by
a derived class, then it is not overriding. */

// Superclass
class Base {

    // Static method in base class which will be hidden in subclass
    public static void display() {
        System.out.println("Static or class method from Base");
    }

    // Non-static method which will be overridden in derived class
    public void print() {
        System.out.println("Non-static or Instance method from Base");
    }
}

// Subclass
class Derived extends Base {

    // This method hides display() in Base
    public static void display() {
        System.out.println("Static or class method from Derived");
    }

    // This method overrides print() in Base
    public void print() {
        System.out.println("Non-static or Instance method from Derived");
    }
}

// Driver class
public class Test {
    public static void main(String args[] ) {
        Base obj1 = new Derived();

        // As per overriding rules this should call to class Derive's static
        // overridden method. Since static method can not be overridden, it
        // calls Base's display()
        obj1.display();

        // Here overriding works and Derive's print() is called
        obj1.print();
    }
}
```

Run on IDE

Output:

```
Static or class method from Base
Non-static or Instance method from Derived
```

Following are some important points for method overriding and static methods in Java.

- 1) For class (or static) methods, the method according to the type of reference is called, not according to the abject being referred, which means method call is decided at compile time.
- 2) For instance (or non-static) methods, the method is called according to the type of object being referred, not according to the type of reference, which means method calls is decided at run time.
- 3) An instance method cannot override a static method, and a static method cannot hide an instance method. For example, the following program has two compiler errors.

```
/* Java program to show that if static methods are redefined by
a derived class, then it is not overriding but hiding. */

// Superclass
class Base {

    // Static method in base class which will be hidden in subclass
    public static void display() {
        System.out.println("Static or class method from Base");
    }

    // Non-static method which will be overridden in derived class
    public void print() {
        System.out.println("Non-static or Instance method from Base");
    }
}

// Subclass
class Derived extends Base {

    // Static is removed here (Causes Compiler Error)
    public void display() {
        System.out.println("Non-static method from Derived");
    }

    // Static is added here (Causes Compiler Error)
    public static void print() {
        System.out.println("Static method from Derived");
    }
}
```

Run on IDE

- 4) In a subclass (or Derived Class), we can overload the methods inherited from the superclass. Such overloaded methods neither hide nor override the superclass methods — they are new methods, unique to the subclass.

References:

<http://docs.oracle.com/javase/tutorial/java/landl/override.html>

This article is contrubuted by **Chandra Prakash**. Please write comments if you find anything incorrect, or you want to share more information about the topic discussed above.

WooCommerce

Build Your Online Store, Your Way.
Get the most customizable eCommerce platform for building your online business.

Get Started >

Java

Improve this Article

Please write to us at contribute@geeksforgeeks.org to report any issue with the above content.

Recommended Posts:

- Overriding in Java
- Overloading in Java
- Blank Final in Java
- Function overloading and return type
- Can we override private methods in Java?
- Trigonometric Functions in Java with Examples
- Stream filter() in Java with examples
- Stream findFirst() in Java with examples
- Stream flatMapToInt() in Java with examples
- Stream mapToInt() in Java with examples

<< Previous Post

Next Post >>

Logged in as SahilAnand(Logout)

3.5 Average Difficulty : 3.5/5.0
Based on 43 vote(s)

Basic Easy Medium Hard Expert

Add to TODO List
Mark as DONE

Get Your Ads On Google Today - Try AdWords & Get Rs.2000

Looking To Gain New Customers? Learn How AdWords Can Help You Maximise Your ROI.

adwords.google.com



Trending Content
Python List
Set to Array in Java
Breadth First Traversal or BFS
School Programming
Longest Repeated Subsequence
Longest Palindromic Subsequence
Detect a negative cycle in a Graph (Bellman Ford)
GATE CS Notes
Depth First Traversal or DFS for a Graph
Reverse a linked list

Manage Chrome with group policies

Most Visited Posts
Top 10 Algorithms and Data Structures for Competitive Programming
Top 10 algorithms in Interview Questions
How to begin with Competitive Programming?
Step by Step Guide for Placement Preparation
How to prepare for ACM-ICPC?
Insertion Sort , Binary Search , QuickSort , MergeSort , HeapSort

Need to get vision on board?

Popular Categories
Interview Experiences
Advanced Data Structures
Dynamic Programming
Greedy Algorithms
Backtracking
Pattern Searching
Divide & Conquer
Geometric Algorithms
Searching
Sorting
Analysis of Algorithms
Mathematical Algorithms
Randomized Algorithms
Recursion
Game Theory
Statistical Algorithms

Advertise Here

Tags

Advanced Data Structure Amazon Aptitude Aptitude Arrays Bit Magic C C C++ C++ Quiz Computer Networks CPP-Library C Quiz Data Structures Data Structures Dynamic Programming Experienced GBlog Geometric Graph Hash Internship Interview Experiences Java Linked Lists Mathematical Matrix Microsoft number-digits PHP Program Output Project Puzzles Python QA - Placement Quizzes QA - Placement Quizzes School Programming Searching series Sorting STL Strings Technical Scripter Trees Web Technologies

Advertise Here

Recent Comments