

JT-Bug-Id	Description
Perf 1	Bx: Primitive value is boxed and then immediately unboxed
Perf 2	Bx: Primitive value is boxed then unboxed to perform primitive coercion
Perf 3	Bx: Boxed value is unboxed and then immediately reboxed
Perf 4	Bx: Method allocates a boxed primitive just to call toString
Perf 5	Bx: Method invokes inefficient floating-point Number constructor; use static valueOf instead
Perf 6	Bx: Method invokes inefficient Number constructor; use static valueOf instead
Perf 7	Dm: The equals and hashCode methods of URL are blocking
Perf 8	Dm: Maps and sets of URLs can be performance hogs
Perf 9	Dm: Method invokes inefficient Boolean constructor; use Boolean.valueOf(...) instead
Perf 10	Dm: Explicit garbage collection; extremely dubious except in benchmarking code
Perf 11	Dm: Method allocates an object, only to get the class object
Perf 12	Dm: Use the nextInt method of Random rather than nextDouble to generate a random integer
Perf 13	Dm: Method invokes inefficient new String(String) constructor
Perf 14	Dm: Method invokes toString() method on a String
Perf 15	Dm: Method invokes inefficient new String() constructor
Perf 16	HSC: Huge string constants is duplicated across multiple class files
Perf 17	ITA: Method uses toArray() with zero-length array argument
Perf 18	SBSC: Method concatenates strings using + in a loop
Perf 19	SIC: Should be a static inner class
Perf 20	SIC: Could be refactored into a named static inner class
Perf 21	SIC: Could be refactored into a static inner class
Perf 22	SS: Unread field: should this field be static?
Perf 23	UM: Method calls static Math class method on a constant value
Perf 24	UPM: Private method is never called
Perf 25	UrF: Unread field
Perf 26	UuF: Unused field
Perf 27	WMI: Inefficient use of keySet iterator instead of entrySet iterator