

JT-Bug-Id	Description
Dodgy 1	<a href="#">BC: Questionable cast to abstract collection</a>
Dodgy 2	<a href="#">BC: Questionable cast to concrete collection</a>
Dodgy 3	<a href="#">BC: Unchecked/unconfirmed cast</a>
Dodgy 4	<a href="#">BC: Unchecked/unconfirmed cast of return value from method</a>
Dodgy 5	<a href="#">BC: instanceof will always return true</a>
Dodgy 6	<a href="#">BSHIFT: Unsigned right shift cast to short/byte</a>
Dodgy 7	<a href="#">CI: Class is final but declares protected field</a>
Dodgy 8	<a href="#">DB: Method uses the same code for two branches</a>
Dodgy 9	<a href="#">DB: Method uses the same code for two switch clauses</a>
Dodgy 10	<a href="#">DLS: Dead store to local variable</a>
Dodgy 11	<a href="#">DLS: Useless assignment in return statement</a>
Dodgy 12	<a href="#">DLS: Dead store of null to local variable</a>
Dodgy 13	<a href="#">DLS: Dead store to local variable that shadows field</a>
Dodgy 14	<a href="#">DMI: Code contains a hard coded reference to an absolute pathname</a>
Dodgy 15	<a href="#">DMI: Non serializable object written to ObjectOutputStream</a>
Dodgy 16	<a href="#">DMI: Invocation of substring(0), which returns the original value</a>
Dodgy 17	<a href="#">Dm: Thread passed where Runnable expected</a>
Dodgy 18	<a href="#">Eq: Class doesn't override equals in superclass</a>
Dodgy 19	<a href="#">Eq: Unusual equals method</a>
Dodgy 20	<a href="#">FE: Test for floating point equality</a>
Dodgy 21	<a href="#">FS: Non-Boolean argument formatted using %b format specifier</a>
Dodgy 22	<a href="#">IA: Ambiguous invocation of either an inherited or outer method</a>
Dodgy 23	<a href="#">IC: Initialization circularity</a>
Dodgy 24	<a href="#">ICAST: integral division result cast to double or float</a>
Dodgy 25	<a href="#">ICAST: Result of integer multiplication cast to long</a>
Dodgy 26	<a href="#">IM: Computation of average could overflow</a>
Dodgy 27	<a href="#">IM: Check for oddness that won't work for negative numbers</a>
Dodgy 28	<a href="#">INT: Integer remainder modulo 1</a>
Dodgy 29	<a href="#">INT: Vacuous bit mask operation on integer value</a>
Dodgy 30	<a href="#">INT: Vacuous comparison of integer value</a>
Dodgy 31	<a href="#">MTIA: Class extends Servlet class and uses instance variables</a>
Dodgy 32	<a href="#">MTIA: Class extends Struts Action class and uses instance variables</a>
Dodgy 33	<a href="#">NP: Dereference of the result of readLine() without nullcheck</a>
Dodgy 34	<a href="#">NP: Immediate dereference of the result of readLine()</a>
Dodgy 35	<a href="#">NP: Load of known null value</a>
Dodgy 36	<a href="#">NP: Possible null pointer dereference due to return value of called method</a>
Dodgy 37	<a href="#">NP: Possible null pointer dereference on branch that might be infeasible</a>
Dodgy 38	<a href="#">NP: Parameter must be nonnull but is marked as nullable</a>
Dodgy 39	<a href="#">NP: Read of unwritten public or protected field</a>
Dodgy 40	<a href="#">NS: Potentially dangerous use of non-short-circuit logic</a>
Dodgy 41	<a href="#">NS: Questionable use of non-short-circuit logic</a>
Dodgy 42	<a href="#">PZLA: Consider returning a zero length array rather than null</a>
Dodgy 43	<a href="#">QF: Complicated, subtle or wrong increment in for-loop</a>
Dodgy 44	<a href="#">RCN: Redundant comparison of non-null value to null</a>
Dodgy 45	<a href="#">RCN: Redundant comparison of two null values</a>
Dodgy 46	<a href="#">RCN: Redundant nullcheck of value known to be non-null</a>
Dodgy 47	<a href="#">RCN: Redundant nullcheck of value known to be null</a>
Dodgy 48	<a href="#">REC: Exception is caught when Exception is not thrown</a>
Dodgy 49	<a href="#">RI: Class implements same interface as superclass</a>

<b>JT-Bug-Id</b>	<b>Description</b>
Dodgy 50	<a href="#">RV: Method checks to see if result of String.indexOf is positive</a>
Dodgy 51	<a href="#">RV: Method discards result of readLine after checking if it is nonnull</a>
Dodgy 52	<a href="#">RV: Remainder of hashCode could be negative</a>
Dodgy 53	<a href="#">RV: Remainder of 32-bit signed random integer</a>
Dodgy 54	<a href="#">RV: Method ignores return value, is this OK?</a>
Dodgy 55	<a href="#">SA: Double assignment of field</a>
Dodgy 56	<a href="#">SA: Double assignment of local variable</a>
Dodgy 57	<a href="#">SA: Self assignment of local variable</a>
Dodgy 58	<a href="#">SF: Switch statement found where one case falls through to the next case</a>
Dodgy 59	<a href="#">SF: Switch statement found where default case is missing</a>
Dodgy 60	<a href="#">ST: Write to static field from instance method</a>
Dodgy 61	<a href="#">Se: private readResolve method not inherited by subclasses</a>
Dodgy 62	<a href="#">Se: Transient field of class that isn't Serializable.</a>
Dodgy 63	<a href="#">TQ: Value required to have type qualifier, but marked as unknown</a>
Dodgy 64	<a href="#">TQ: Value required to not have type qualifier, but marked as unknown</a>
Dodgy 65	<a href="#">UCF: Useless control flow</a>
Dodgy 66	<a href="#">UCF: Useless control flow to next line</a>
Dodgy 67	<a href="#">UrF: Unread public/protected field</a>
Dodgy 68	<a href="#">UuF: Unused public or protected field</a>
Dodgy 69	<a href="#">UwF: Field not initialized in constructor but dereferenced without null check</a>
Dodgy 70	<a href="#">UwF: Unwritten public or protected field</a>
Dodgy 71	<a href="#">XFB: Method directly allocates a specific implementation of xml interfaces</a>