

| JT-Bug-Id | Description |
|-----------|---|
| BP 1 | AM: Creates an empty jar file entry |
| BP 2 | AM: Creates an empty zip file entry |
| BP 3 | BC: Equals method should not assume anything about the type of its argument |
| BP 4 | BIT: Check for sign of bitwise operation |
| BP 5 | CN: Class implements Cloneable but does not define or use clone method |
| BP 6 | CN: clone method does not call super.clone() |
| BP 7 | CN: Class defines clone() but doesn't implement Cloneable |
| BP 8 | Co: Abstract class defines covariant compareTo() method |
| BP 9 | Co: Covariant compareTo() method defined |
| BP 10 | DE: Method might drop exception |
| BP 11 | DE: Method might ignore exception |
| BP 12 | DMI: Adding elements of an entry set may fail due to reuse of Entry objects |
| BP 13 | DMI: Random object created and used only once |
| BP 14 | DMI: Don't use removeAll to clear a collection |
| BP 15 | Dm: Method invokes System.exit(...) |
| BP 16 | Dm: Method invokes dangerous method runFinalizersOnExit |
| BP 17 | ES: Comparison of String parameter using == or != |
| BP 18 | ES: Comparison of String objects using == or != |
| BP 19 | Eq: Abstract class defines covariant equals() method |
| BP 20 | Eq: Equals checks for incompatible operand |
| BP 21 | Eq: Class defines compareTo(...) and uses Object.equals() |
| BP 22 | Eq: equals method fails for subtypes |
| BP 23 | Eq: Covariant equals() method defined |
| BP 24 | FI: Empty finalizer should be deleted |
| BP 25 | FI: Explicit invocation of finalizer |
| BP 26 | FI: Finalizer nulls fields |
| BP 27 | FI: Finalizer only nulls fields |
| BP 28 | FI: Finalizer does not call superclass finalizer |
| BP 29 | FI: Finalizer nullifies superclass finalizer |
| BP 30 | FI: Finalizer does nothing but call superclass finalizer |
| BP 31 | FS: Format string should use %n rather than \n |
| BP 32 | GC: Unchecked type in generic call |
| BP 33 | HE: Class defines equals() but not hashCode() |
| BP 34 | HE: Class defines equals() and uses Object.hashCode() |
| BP 35 | HE: Class defines hashCode() but not equals() |
| BP 36 | HE: Class defines hashCode() and uses Object.equals() |
| BP 37 | HE: Class inherits equals() and uses Object.hashCode() |
| BP 38 | IC: Superclass uses subclass during initialization |
| BP 39 | IMSE: Dubious catching of IllegalMonitorStateException |
| BP 40 | ISC: Needless instantiation of class that only supplies static methods |
| BP 41 | It: Iterator next() method can't throw NoSuchElementException |
| BP 42 | J2EE: Store of non serializable object into HttpSession |
| BP 43 | JCIP: Fields of immutable classes should be final |
| BP 44 | NP: Method with Boolean return type returns explicit null |
| BP 45 | NP: Clone method may return null |
| BP 46 | NP: equals() method does not check for null argument |
| BP 47 | NP: toString method may return null |
| BP 48 | Nm: Class names should start with an upper case letter |
| BP 49 | Nm: Class is not derived from an Exception, even though it is named as such |

| JT-Bug-Id | Description |
|-----------|---|
| BP 50 | Nm: Confusing method names |
| BP 51 | Nm: Field names should start with a lower case letter |
| BP 52 | Nm: Use of identifier that is a keyword in later versions of Java |
| BP 53 | Nm: Use of identifier that is a keyword in later versions of Java |
| BP 54 | Nm: Method names should start with a lower case letter |
| BP 55 | Nm: Class names shouldn't shadow simple name of implemented interface |
| BP 56 | Nm: Class names shouldn't shadow simple name of superclass |
| BP 57 | Nm: Very confusing method names (but perhaps intentional) |
| BP 58 | Nm: Method doesn't override method in superclass due to wrong package for parameter |
| BP 59 | ODR: Method may fail to close database resource |
| BP 60 | ODR: Method may fail to close database resource on exception |
| BP 61 | OS: Method may fail to close stream |
| BP 62 | OS: Method may fail to close stream on exception |
| BP 63 | PZ: Don't reuse entry objects in iterators |
| BP 64 | RC: Suspicious reference comparison to constant |
| BP 65 | RC: Suspicious reference comparison of Boolean values |
| BP 66 | RR: Method ignores results of InputStream.read() |
| BP 67 | RR: Method ignores results of InputStream.skip() |
| BP 68 | RV: Negating the result of compareTo()/compare() |
| BP 69 | RV: Method ignores exceptional return value |
| BP 70 | SI: Static initializer creates instance before all static final fields assigned |
| BP 71 | SW: Certain swing methods needs to be invoked in Swing thread |
| BP 72 | Se: Non-transient non-serializable instance field in serializable class |
| BP 73 | Se: Non-serializable class has a serializable inner class |
| BP 74 | Se: Non-serializable value stored into instance field of a serializable class |
| BP 75 | Se: Comparator doesn't implement Serializable |
| BP 76 | Se: Serializable inner class |
| BP 77 | Se: serialVersionUID isn't final |
| BP 78 | Se: serialVersionUID isn't long |
| BP 79 | Se: serialVersionUID isn't static |
| BP 80 | Se: Class is Serializable but its superclass doesn't define a void constructor |
| BP 81 | Se: Class is Externalizable but doesn't define a void constructor |
| BP 82 | Se: The readResolve method must be declared with a return type of Object. |
| BP 83 | Se: Transient field that isn't set by deserialization. |
| BP 84 | SnVI: Class is Serializable, but doesn't define serialVersionUID |
| BP 85 | UI: Usage of GetResource may be unsafe if class is extended |