Range Pit Test Coverage Report

Pit Test Coverage Report
Package Summary
org.jfree.data

Number of Classes	Line Coverage	Mutation Coverage
15	5% 58/1116	8% 536/6839

Breakdown by Class

Name	Line Coverage	Mutation Coverage
ComparableObjectItem.java	0% 0/28	0% 0/146
ComparableObjectSeries.java	0% 0/117	0% 0/886
DataUtilities.java	0% 0/80	0% 0/687
DefaultKeyedValue.java	0% 0/25	0% 0/153
DefaultKeyedValues.java	0% 0/123	0% 0/661
DefaultKeyedValues2D.java	0% 0/152	0% 0/862
DomainOrder.java	0% 0/25	0% 0/91
KeyToGroupMap.java	0% 0/94	0% 0/322
KeyedObject.java	0% 0/23	0% 0/87
KeyedObjects.java	0% 0/95	0% 0/437
KeyedObjects2D.java	0% 0/157	0% 0/871
KeyedValueComparator.java	0% 0/52	0% 0/230
KeyedValueComparatorType.java	0% 0/17	0% 0/56
Range.java	56% 58/103	43% 536/1259
RangeType.java	0% 0/25	0% 0/91

Report generated by PIT 1.4.11

Range.java

1		/* ===
2		* JFreeChart : a free chart library for the Java(tm) platform
3		* ===
4		*
5		* (C) Copyright 2000-2014, by Object Refinery Limited and Contributors.
6		*
7		* Project Info: http://www.jfree.org/jfreechart/index.html
8		*
9		* This library is free software; you can redistribute it and/or modify it
10		* under the terms of the GNU Lesser General Public License as published by
11		* the Free Software Foundation; either version 2.1 of the License, or
12		* (at your option) any later version.
13		*
14		* This library is distributed in the hope that it will be useful, but
15		* WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY
16		* or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public
17		* License for more details.
18		*
19		* You should have received a copy of the GNU Lesser General Public
20		* License along with this library; if not, write to the Free Software
21		* Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301,
22		* USA.
23		*
24		* [Oracle and Java are registered trademarks of Oracle and/or its affiliates.
25		* Other names may be trademarks of their respective owners.]
26		*
27		* ----------
28		* Range.java
29		* ----------
30		* (C) Copyright 2002-2014, by Object Refinery Limited and Contributors.
31		*
32		* Original Author: David Gilbert (for Object Refinery Limited);
33		* Contributor(s): Chuanhao Chiu;
34		* Bill Kelemen;
35		* Nicolas Brodu;

36		* Sergei Ivanov;
37		*
38		* Changes (from 23-Jun-2001)
39		* -------------------------
40		* 22-Apr-2002 : Version 1, loosely based by code by Bill Kelemen (DG);
41		* 30-Apr-2002 : Added getLength() and getCentralValue() methods. Changed
42		* argument check in constructor (DG);
43		* 13-Jun-2002 : Added contains(double) method (DG);
44		* 22-Aug-2002 : Added fix to combine method where both ranges are null, thanks
45		* to Chuanhao Chiu for reporting and fixing this (DG);
46		* 07-Oct-2002 : Fixed errors reported by Checkstyle (DG);
47		* 26-Mar-2003 : Implemented Serializable (DG);
48		* 14-Aug-2003 : Added equals() method (DG);
49		* 27-Aug-2003 : Added toString() method (BK);
50		* 11-Sep-2003 : Added Clone Support (NB);
51		* 23-Sep-2003 : Fixed Checkstyle issues (DG);
52		* 25-Sep-2003 : Oops, Range immutable, clone not necessary (NB);
53		* 05-May-2004 : Added constrain() and intersects() methods (DG);
54		* 18-May-2004 : Added expand() method (DG);
55		* ------------- JFreeChart 1.0.x ---
56		* 11-Jan-2006 : Added new method expandToInclude(Range, double) (DG);
57		* 18-Dec-2007 : New methods intersects(Range) and scale(...) thanks to Sergei
58		* Ivanov (DG);
59		* 08-Jan-2012 : New method combineIgnoringNaN() (DG);
60		* 23-Feb-2014 : Added isNaNRange() method (DG);
61		*
62		*/
63		
64		package org.jfree.data;
65		
66		import java.io.Serializable;
67		import org.jfree.chart.util.ParamChecks;
68		
69		/**
70		* Represents an immutable range of values.
71		*/
72		public strictfp class Range implements Serializable {
73		
74		/** For serialization. */
75		private static final long serialVersionUID = -906333695431863380L;
76		

77		/** The lower bound of the range. */
78		private double lower;
79		
80		/** The upper bound of the range. */
81		private double upper;
82		
83		/**
84		* Creates a new range.
85		*
86		* @param lower the lower bound (must be <= upper bound).
87		* @param upper the upper bound (must be >= lower bound).
88		*/
89		public Range(double lower, double upper) {
90	19	`        if (lower > upper) {`
91	9	`            String msg = "Range(double, double): require lower (" + lower`
92	11	`                + ") <= upper (" + upper + ").";`
93	1	`            throw new IllegalArgumentException(msg);`
94		`        }`
95	6	`        this.lower = lower;`
96	6	`        this.upper = upper;`
97		`    }`
98		
99		/**
100		* Returns the lower bound for the range.
101		*
102		* @return The lower bound.
103		*/
104		public double getLowerBound() {
105	7	`        return this.lower;`
106		`    }`
107		
108		/**
109		* Returns the upper bound for the range.
110		*
111		* @return The upper bound.
112		*/
113		public double getUpperBound() {
114	7	`        return this.upper;`
115		`    }`
116		
117		/**

118		* Returns the length of the range.
119		*
120		* @return The length.
121		*/
122		public double getLength() {
123	19	`        return this.upper - this.lower;`
124		}
125		
126		/**
127		* Returns the central value for the range.
128		*
129		* @return The central value.
130		*/
131		public double getCentralValue() {
132	47	`        return this.lower / 2.0 + this.upper / 2.0;`
133		}
134		
135		/**
136		* Returns <code>true</code> if the range contains the specified value and
137		* <code>false</code> otherwise.
138		*
139		* @param value the value to lookup.
140		*
141		* @return <code>true</code> if the range contains the specified value.
142		*/
143		public boolean contains(double value) {
144	53	`        return (value >= this.lower && value <= this.upper);`
145		}
146		
147		/**
148		* Returns <code>true</code> if the range intersects with the specified
149		* range, and <code>false</code> otherwise.
150		*
151		* @param b0 the lower bound (should be <= b1).
152		* @param b1 the upper bound (should be >= b0).
153		*
154		* @return A boolean.
155		*/
156		public boolean intersects(double b0, double b1) {
157	19	`        if (b0 <= this.lower) {`
158	34	`            return (b1 > this.lower);`

159		}
160		else {
161	53	`return (b0 < this.upper && b1 >= b0);`
162		}
163		}
164		
165		/**
166		* Returns <code>true</code> if the range intersects with the specified
167		* range, and <code>false</code> otherwise.
168		*
169		* @param range another range (<code>null</code> not permitted).
170		*
171		* @return A boolean.
172		*
173		* @since 1.0.9
174		*/
175		public boolean intersects(Range range) {
176	6	`return intersects(range.getLowerBound(),` `range.getUpperBound());`
177		}
178		
179		/**
180		* Returns the value within the range that is closest to the specified
181		* value.
182		*
183		* @param value the value.
184		*
185		* @return The constrained value.
186		*/
187		public double constrain(double value) {
188	5	`double result = value;`
189	14	`if (!contains(value)) {`
190	19	`if (value > this.upper) {`
191	5	`result = this.upper;`
192		`}`
193	19	`else if (value < this.lower) {`
194	5	`result = this.lower;`
195		`}`
196		`}`
197	7	`return result;`
198		}
199		

200		/**
201		* Creates a new range by combining two existing ranges.
202		* <P>
203		* Note that:
204		*
205		* either range can be <code>null</code>, in which case the other
206		* range is returned;
207		* if both ranges are <code>null</code> the return value is
208		* <code>null</code>.
209		*
210		*
211		* @param range1 the first range (<code>null</code> permitted).
212		* @param range2 the second range (<code>null</code> permitted).
213		*
214		* @return A new range (possibly <code>null</code>).
215		*/
216		public static Range combine(Range range1, Range range2) {
217	4	`        if (range1 == null) {`
218	2	`            return range2;`
219		`        }`
220	4	`        if (range2 == null) {`
221	2	`            return range1;`
222		`        }`
223	4	`        double l = Math.min(range1.getLowerBound(), range2.getLowerBound());`
224	4	`        double u = Math.max(range1.getUpperBound(), range2.getUpperBound());`
225	13	`        return new Range(l, u);`
226		`    }`
227		
228		/**
229		* Returns a new range that spans both <code>range1</code> and
230		* <code>range2</code>. This method has a special handling to ignore
231		* Double.NaN values.
232		*
233		* @param range1 the first range (<code>null</code> permitted).
234		* @param range2 the second range (<code>null</code> permitted).
235		*
236		* @return A new range (possibly <code>null</code>).
237		*
238		* @since 1.0.15
239		*/

240		public static Range combineIgnoringNaN(Range range1, Range range2) {
241	4	if (range1 == null) {
242	13	if (range2 != null && range2.isNaNRange()) {
243	1	return null;
244		}
245	2	return range2;
246		}
247	4	if (range2 == null) {
248	9	if (range1.isNaNRange()) {
249	1	return null;
250		}
251	2	return range1;
252		}
253	4	double l = min(range1.getLowerBound(), range2.getLowerBound());
254	4	double u = max(range1.getUpperBound(), range2.getUpperBound());
255	28	if (Double.isNaN(l) && Double.isNaN(u)) {
256	1	return null;
257		}
258	13	return new Range(l, u);
259		}
260		
261		/**
262		* Returns the minimum value. If either value is NaN, the other value is
263		* returned. If both are NaN, NaN is returned.
264		*
265		* @param d1 value 1.
266		* @param d2 value 2.
267		*
268		* @return The minimum of the two values.
269		*/
270		private static double min(double d1, double d2) {
271	14	if (Double.isNaN(d1)) {
272	7	return d2;
273		}
274	14	if (Double.isNaN(d2)) {
275	7	return d1;
276		}
277	14	return Math.min(d1, d2);
278		}
279		
280		private static double max(double d1, double d2) {

281	14	`        if (Double.isNaN(d1)) {`
282	7	`            return d2;`
283		`        }`
284	14	`        if (Double.isNaN(d2)) {`
285	7	`            return d1;`
286		`        }`
287	14	`        return Math.max(d1, d2);`
288		`    }`
289		
290		`    /**`
291		`     * Returns a range that includes all the values in the specified`
292		`     * <code>range</code> AND the specified <code>value</code>.`
293		`     *`
294		`     * @param range  the range (<code>null</code> permitted).`
295		`     * @param value  the value that must be included.`
296		`     *`
297		`     * @return A range.`
298		`     *`
299		`     * @since 1.0.1`
300		`     */`
301		`    public static Range expandToInclude(Range range, double value) {`
302	4	`        if (range == null) {`
303	13	`            return new Range(value, value);`
304		`        }`
305	15	`        if (value < range.getLowerBound()) {`
306	9	`            return new Range(value, range.getUpperBound());`
307		`        }`
308	15	`        else if (value > range.getUpperBound()) {`
309	9	`            return new Range(range.getLowerBound(), value);`
310		`        }`
311		`        else {`
312	2	`            return range;`
313		`        }`
314		`    }`
315		
316		`    /**`
317		`     * Creates a new range by adding margins to an existing range.`
318		`     *`
319		`     * @param range  the range (<code>null</code> not permitted).`
320		`     * @param lowerMargin  the lower margin (expressed as a percentage of the`
321		`     *               range length).`
322		`     * @param upperMargin  the upper margin (expressed as a percentage of the`

323		* range length).
324		*
325		* @return The expanded range.
326		*/
327		public static Range expand(Range range,
328		double lowerMargin, double upperMargin) {
329	1	`ParamChecks.nullNotPermitted(range, "range");`
330	1	`double length = range.getLength();`
331	25	`double lower = range.getLowerBound() - length * lowerMargin;`
332	25	`double upper = range.getUpperBound() + length * upperMargin;`
333	19	`if (lower > upper) {`
334	45	`lower = lower / 2.0 + upper / 2.0;`
335	5	`upper = lower;`
336		`}`
337	13	`return new Range(lower, upper);`
338		`}`
339		
340		/**
341		* Shifts the range by the specified amount.
342		*
343		* @param base the base range (<code>null</code> not permitted).
344		* @param delta the shift amount.
345		*
346		* @return A new range.
347		*/
348		public static Range shift(Range base, double delta) {
349	14	`return shift(base, delta, false);`
350		`}`
351		
352		/**
353		* Shifts the range by the specified amount.
354		*
355		* @param base the base range (<code>null</code> not permitted).
356		* @param delta the shift amount.
357		* @param allowZeroCrossing a flag that determines whether or not the
358		* bounds of the range are allowed to cross
359		* zero after adjustment.
360		*
361		* @return A new range.
362		*/
363		public static Range shift(Range base, double delta,

364		boolean allowZeroCrossing) {
365	1	`        ParamChecks.nullNotPermitted(base, "base");`
366	13	`        if (allowZeroCrossing) {`
367	16	`            return new Range(base.getLowerBound() + delta,`
368	13	`                    base.getUpperBound() + delta);`
369		}
370		else {
371	6	`            return new Range(shiftWithNoZeroCrossing(base.getLowerBound(),`
372	8	`                    delta), shiftWithNoZeroCrossing(base.getUpperBound(),`
373	5	`                    delta));`
374		}
375		}
376		
377		/**
378		* Returns the given <code>value</code> adjusted by <code>delta</code> but
379		* with a check to prevent the result from crossing <code>0.0</code>.
380		*
381		* @param value the value.
382		* @param delta the adjustment.
383		*
384		* @return The adjusted value.
385		*/
386		private static double shiftWithNoZeroCrossing(double value, double delta) {
387	19	`        if (value > 0.0) {`
388	26	`            return Math.max(value + delta, 0.0);`
389		}
390	19	`        else if (value < 0.0) {`
391	26	`            return Math.min(value + delta, 0.0);`
392		}
393		else {
394	19	`            return value + delta;`
395		}
396		}
397		
398		/**
399		* Scales the range by the specified factor.
400		*
401		* @param base the base range (<code>null</code> not permitted).
402		* @param factor the scaling factor (must be non-negative).
403		*

404		* @return A new range.
405		*
406		* @since 1.0.9
407		*/
408		public static Range scale(Range base, double factor) {
409	1	`ParamChecks.nullNotPermitted(base, "base");`
410	19	`if (factor < 0) {`
411	1	`throw new IllegalArgumentException("Negative 'factor' argument.");`
412		`}`
413	16	`return new Range(base.getLowerBound() * factor,`
414	13	`base.getUpperBound() * factor);`
415		`}`
416		
417		/**
418		* Tests this object for equality with an arbitrary object.
419		*
420		* @param obj the object to test against (<code>null</code> permitted).
421		*
422		* @return A boolean.
423		*/
424		@Override
425		public boolean equals(Object obj) {
426	8	`if (!(obj instanceof Range)) {`
427	7	`return false;`
428		`}`
429		Range range = (Range) obj;
430	18	`if (!(this.lower == range.lower)) {`
431	7	`return false;`
432		`}`
433	18	`if (!(this.upper == range.upper)) {`
434	7	`return false;`
435		`}`
436	8	`return true;`
437		`}`
438		
439		/**
440		* Returns <code>true</code> if both the lower and upper bounds are
441		* <code>Double.NaN</code>, and <code>false</code> otherwise.
442		*
443		* @return A boolean.
444		*

445		* @since 1.0.18
446		*/
447		public boolean isNaNRange() {
448	<u>43</u>	`return Double.isNaN(this.lower) && Double.isNaN(this.upper);`
449		}
450		
451		/**
452		* Returns a hash code.
453		*
454		* @return A hash code.
455		*/
456		@Override
457		public int hashCode() {
458		int result;
459		long temp;
460	<u>6</u>	`temp = Double.doubleToLongBits(this.lower);`
461	<u>19</u>	`result = (int) (temp ^ (temp >>> 32));`
462	<u>6</u>	`temp = Double.doubleToLongBits(this.upper);`
463	<u>45</u>	`result = 29 * result + (int) (temp >>> 32));`
464	<u>7</u>	`return result;`
465		}
466		
467		/**
468		* Returns a string representation of this Range.
469		*
470		* @return A String "Range[lower,upper]" where lower=lower range and
471		* upper=upper range.
472		*/
473		@Override
474		public String toString() {
475	<u>22</u>	`return ("Range[" + this.lower + "," + this.upper + "]");`
476		}
477		
478		}

Mutations

<u>90</u>		1. changed conditional boundary → KILLED 2. negated conditional → KILLED 3. removed conditional - replaced comparison check with false → SURVIVED 4. removed conditional - replaced comparison check with true → KILLED

		5. Negated double local variable number 1 → KILLED 6. Negated double local variable number 3 → KILLED 7. Less or equal to less than → KILLED 8. Less or equal to greater than → KILLED 9. Less or equal to greater or equal → KILLED 10. Less or equal to equal → KILLED 11. Less or equal to not equal → KILLED 12. Incremented (a++) double local variable number 1 → KILLED 13. Incremented (a++) double local variable number 3 → KILLED 14. Decremented (a--) double local variable number 1 → KILLED 15. Decremented (a--) double local variable number 3 → KILLED 16. Incremented (++a) double local variable number 1 → KILLED 17. Incremented (++a) double local variable number 3 → KILLED 18. Decremented (--a) double local variable number 1 → KILLED 19. Decremented (--a) double local variable number 3 → KILLED
91		1. removed call to java/lang/StringBuilder::<init> → NO_COVERAGE 2. removed call to java/lang/StringBuilder::append → NO_COVERAGE 3. removed call to java/lang/StringBuilder::toString → NO_COVERAGE 4. replaced call to java/lang/StringBuilder::append with receiver → NO_COVERAGE 5. Negated double local variable number 1 → NO_COVERAGE 6. Incremented (a++) double local variable number 1 → NO_COVERAGE 7. Decremented (a--) double local variable number 1 → NO_COVERAGE 8. Incremented (++a) double local variable number 1 → NO_COVERAGE 9. Decremented (--a) double local variable number 1 → NO_COVERAGE
92		1. removed call to java/lang/StringBuilder::append → NO_COVERAGE 2. removed call to java/lang/StringBuilder::append → NO_COVERAGE 3. removed call to java/lang/StringBuilder::append → NO_COVERAGE 4. replaced call to java/lang/StringBuilder::append with receiver → NO_COVERAGE 5. replaced call to java/lang/StringBuilder::append with receiver → NO_COVERAGE 6. replaced call to java/lang/StringBuilder::append with receiver → NO_COVERAGE 7. Negated double local variable number 3 → NO_COVERAGE 8. Incremented (a++) double local variable number 3 → NO_COVERAGE 9. Decremented (a--) double local variable number 3 → NO_COVERAGE 10. Incremented (++a) double local variable number 3 →

		NO_COVERAGE 11. Decremented (--a) double local variable number 3 → NO_COVERAGE
93		1. removed call to java/lang/IllegalArgumentException::<init> → NO_COVERAGE
95		1. Removed assignment to member variable lower → KILLED 2. Negated double local variable number 1 → KILLED 3. Incremented (a++) double local variable number 1 → SURVIVED 4. Decremented (a--) double local variable number 1 → SURVIVED 5. Incremented (++a) double local variable number 1 → KILLED 6. Decremented (--a) double local variable number 1 → KILLED
96		1. Removed assignment to member variable upper → KILLED 2. Negated double local variable number 3 → KILLED 3. Incremented (a++) double local variable number 3 → SURVIVED 4. Decremented (a--) double local variable number 3 → SURVIVED 5. Incremented (++a) double local variable number 3 → KILLED 6. Decremented (--a) double local variable number 3 → KILLED
105		1. replaced double return with 0.0d for org/jfree/data/Range::getLowerBound → KILLED 2. replaced return of double value with -(x + 1) for org/jfree/data/Range::getLowerBound → KILLED 3. Negated double field lower → KILLED 4. Incremented (a++) double field lower → KILLED 5. Decremented (a--) double field lower → KILLED 6. Incremented (++a) double field lower → KILLED 7. Decremented (--a) double fieldlower → KILLED
114		1. replaced double return with 0.0d for org/jfree/data/Range::getUpperBound → KILLED 2. replaced return of double value with -(x + 1) for org/jfree/data/Range::getUpperBound → KILLED 3. Negated double field upper → KILLED 4. Incremented (a++) double field upper → KILLED 5. Decremented (a--) double field upper → KILLED 6. Incremented (++a) double field upper → KILLED 7. Decremented (--a) double fieldupper → KILLED
123		1. Replaced double subtraction with addition → NO_COVERAGE 2. replaced double return with 0.0d for org/jfree/data/Range::getLength → NO_COVERAGE 3. replaced return of double value with -(x + 1) for org/jfree/data/Range::getLength → NO_COVERAGE 4. Negated double field upper → NO_COVERAGE 5. Negated double field lower → NO_COVERAGE 6. Replaced double operation with first member → NO_COVERAGE 7. Replaced double operation by second member → NO_COVERAGE 8. Replaced double subtraction with addition → NO_COVERAGE 9. Replaced double subtraction with multiplication → NO_COVERAGE

		10. Replaced double subtraction with division → NO_COVERAGE 11. Replaced double subtraction with modulus → NO_COVERAGE 12. Incremented (a++) double field upper → NO_COVERAGE 13. Incremented (a++) double field lower → NO_COVERAGE 14. Decremented (a--) double field upper → NO_COVERAGE 15. Decremented (a--) double field lower → NO_COVERAGE 16. Incremented (++a) double field upper → NO_COVERAGE 17. Incremented (++a) double field lower → NO_COVERAGE 18. Decremented (--a) double fieldupper → NO_COVERAGE 19. Decremented (--a) double fieldlower → NO_COVERAGE
132		1. Substituted 2.0 with 1.0 → NO_COVERAGE 2. Substituted 2.0 with 1.0 → NO_COVERAGE 3. Replaced double division with multiplication → NO_COVERAGE 4. Replaced double division with multiplication → NO_COVERAGE 5. Replaced double addition with subtraction → NO_COVERAGE 6. replaced double return with 0.0d for org/jfree/data/Range::getCentralValue → NO_COVERAGE 7. replaced return of double value with -(x + 1) for org/jfree/data/Range::getCentralValue → NO_COVERAGE 8. Negated double field lower → NO_COVERAGE 9. Negated double field upper → NO_COVERAGE 10. Replaced double operation with first member → NO_COVERAGE 11. Replaced double operation with first member → NO_COVERAGE 12. Replaced double operation with first member → NO_COVERAGE 13. Replaced double operation by second member → NO_COVERAGE 14. Replaced double operation by second member → NO_COVERAGE 15. Replaced double operation by second member → NO_COVERAGE 16. Replaced double division with multiplication → NO_COVERAGE 17. Replaced double division with multiplication → NO_COVERAGE 18. Replaced double addition with subtraction → NO_COVERAGE 19. Replaced double division with modulus → NO_COVERAGE 20. Replaced double division with modulus → NO_COVERAGE 21. Replaced double addition with multiplication → NO_COVERAGE 22. Replaced double division with addition → NO_COVERAGE 23. Replaced double division with addition → NO_COVERAGE 24. Replaced double addition with division → NO_COVERAGE 25. Replaced double division with subtraction → NO_COVERAGE 26. Replaced double division with subtraction → NO_COVERAGE 27. Replaced double addition with modulus → NO_COVERAGE 28. Substituted 2.0 with 1.0 → NO_COVERAGE 29. Substituted 2.0 with 1.0 → NO_COVERAGE 30. Substituted 2.0 with 0.0 → NO_COVERAGE 31. Substituted 2.0 with 0.0 → NO_COVERAGE 32. Substituted 2.0 with -1.0 → NO_COVERAGE 33. Substituted 2.0 with -1.0 → NO_COVERAGE 34. Substituted 2.0 with -2.0 → NO_COVERAGE 35. Substituted 2.0 with -2.0 → NO_COVERAGE

		36. Substituted 2.0 with 3.0 → NO_COVERAGE 37. Substituted 2.0 with 3.0 → NO_COVERAGE 38. Substituted 2.0 with 1.0 → NO_COVERAGE 39. Substituted 2.0 with 1.0 → NO_COVERAGE 40. Incremented (a++) double field lower → NO_COVERAGE 41. Incremented (a++) double field upper → NO_COVERAGE 42. Decremented (a--) double field lower → NO_COVERAGE 43. Decremented (a--) double field upper → NO_COVERAGE 44. Incremented (++a) double field lower → NO_COVERAGE 45. Incremented (++a) double field upper → NO_COVERAGE 46. Decremented (--a) double fieldlower → NO_COVERAGE 47. Decremented (--a) double fieldupper → NO_COVERAGE
144		1. replaced boolean return with false for org/jfree/data/Range::contains → NO_COVERAGE 2. replaced boolean return with true for org/jfree/data/Range::contains → NO_COVERAGE 3. changed conditional boundary → NO_COVERAGE 4. changed conditional boundary → NO_COVERAGE 5. Substituted 1 with 0 → NO_COVERAGE 6. Substituted 0 with 1 → NO_COVERAGE 7. negated conditional → NO_COVERAGE 8. negated conditional → NO_COVERAGE 9. removed conditional - replaced comparison check with false → NO_COVERAGE 10. removed conditional - replaced comparison check with false → NO_COVERAGE 11. removed conditional - replaced comparison check with true → NO_COVERAGE 12. removed conditional - replaced comparison check with true → NO_COVERAGE 13. replaced return of integer sized value with (x == 0 ? 1 : 0) → NO_COVERAGE 14. replaced return of integer sized value with (x == 0 ? 1 : 0) → NO_COVERAGE 15. Negated double local variable number 1 → NO_COVERAGE 16. Negated double field lower → NO_COVERAGE 17. Negated double local variable number 1 → NO_COVERAGE 18. Negated double field upper → NO_COVERAGE 19. Substituted 0 with 1 → NO_COVERAGE 20. Substituted 1 with 0 → NO_COVERAGE 21. Substituted 1 with -1 → NO_COVERAGE 22. Substituted 0 with -1 → NO_COVERAGE 23. Substituted 1 with -1 → NO_COVERAGE 24. Substituted 1 with 2 → NO_COVERAGE 25. Substituted 0 with 1 → NO_COVERAGE 26. Substituted 1 with 0 → NO_COVERAGE 27. Substituted 0 with -1 → NO_COVERAGE 28. Less than to less or equal → NO_COVERAGE 29. greater than to less than → NO_COVERAGE 30. Less than to greater than → NO_COVERAGE 31. greater than to less or equal → NO_COVERAGE 32. Less than to greater or equal → NO_COVERAGE 33. greater than to greater or equal → NO_COVERAGE 34. Less than to equal → NO_COVERAGE 35. greater than to equal → NO_COVERAGE 36. Less than to not equal → NO_COVERAGE 37. greater than to not equal → NO_COVERAGE 38. Incremented (a++) double local variable number 1 →

NO_COVERAGE
39. Incremented (a++) double field lower → NO_COVERAGE
40. Incremented (a++) double local variable number 1 →
NO_COVERAGE
41. Incremented (a++) double field upper → NO_COVERAGE
42. Decremented (a--) double local variable number 1 →
NO_COVERAGE
43. Decremented (a--) double field lower → NO_COVERAGE
44. Decremented (a--) double local variable number 1 →
NO_COVERAGE
45. Decremented (a--) double field upper → NO_COVERAGE
46. Incremented (++a) double local variable number 1 →
NO_COVERAGE
47. Incremented (++a) double field lower → NO_COVERAGE
48. Incremented (++a) double local variable number 1 →
NO_COVERAGE
49. Incremented (++a) double field upper → NO_COVERAGE
50. Decremented (--a) double local variable number 1 →
NO_COVERAGE
51. Decremented (--a) double fieldlower → NO_COVERAGE
52. Decremented (--a) double local variable number 1 →
NO_COVERAGE
53. Decremented (--a) double fieldupper → NO_COVERAGE

| 157 | | 1. changed conditional boundary → SURVIVED
2. negated conditional → KILLED
3. removed conditional - replaced comparison check with false → KILLED
4. removed conditional - replaced comparison check with true → KILLED
5. Negated double local variable number 1 → KILLED
6. Negated double field lower → KILLED
7. greater than to less than → KILLED
8. greater than to less or equal → KILLED
9. greater than to greater or equal → SURVIVED
10. greater than to equal → KILLED
11. greater than to not equal → KILLED
12. Incremented (a++) double local variable number 1 → KILLED
13. Incremented (a++) double field lower → KILLED
14. Decremented (a--) double local variable number 1 → KILLED
15. Decremented (a--) double field lower → KILLED
16. Incremented (++a) double local variable number 1 → KILLED
17. Incremented (++a) double field lower → KILLED
18. Decremented (--a) double local variable number 1 → KILLED
19. Decremented (--a) double fieldlower → KILLED |
| 158 | | 1. replaced boolean return with false for org/jfree/data/Range::intersects → KILLED
2. replaced boolean return with true for org/jfree/data/Range::intersects → KILLED
3. changed conditional boundary → KILLED
4. Substituted 1 with 0 → KILLED
5. Substituted 0 with 1 → KILLED
6. negated conditional → KILLED
7. removed conditional - replaced comparison check with false → KILLED |

		8. removed conditional - replaced comparison check with true → KILLED
		9. replaced return of integer sized value with (x == 0 ? 1 : 0) → KILLED
		10. replaced return of integer sized value with (x == 0 ? 1 : 0) → KILLED
		11. Negated double local variable number 3 → KILLED
		12. Negated double field lower → KILLED
		13. Substituted 0 with 1 → KILLED
		14. Substituted 1 with 0 → KILLED
		15. Substituted 1 with -1 → SURVIVED
		16. Substituted 0 with -1 → KILLED
		17. Substituted 1 with -1 → SURVIVED
		18. Substituted 1 with 2 → KILLED
		19. Substituted 0 with 1 → KILLED
		20. Substituted 1 with 0 → KILLED
		21. Substituted 0 with -1 → KILLED
		22. Less or equal to less than → KILLED
		23. Less or equal to greater than → KILLED
		24. Less or equal to greater or equal → KILLED
		25. Less or equal to equal → SURVIVED
		26. Less or equal to not equal → KILLED
		27. Incremented (a++) double local variable number 3 → SURVIVED
		28. Incremented (a++) double field lower → SURVIVED
		29. Decremented (a--) double local variable number 3 → SURVIVED
		30. Decremented (a--) double field lower → SURVIVED
		31. Incremented (++a) double local variable number 3 → KILLED
		32. Incremented (++a) double field lower → KILLED
		33. Decremented (--a) double local variable number 3 → KILLED
		34. Decremented (--a) double fieldlower → KILLED
161		1. replaced boolean return with false for org/jfree/data/Range::intersects → KILLED
		2. replaced boolean return with true for org/jfree/data/Range::intersects → KILLED
		3. changed conditional boundary → KILLED
		4. changed conditional boundary → KILLED
		5. Substituted 1 with 0 → KILLED
		6. Substituted 0 with 1 → KILLED
		7. negated conditional → KILLED
		8. negated conditional → KILLED
		9. removed conditional - replaced comparison check with false → KILLED
		10. removed conditional - replaced comparison check with false → KILLED
		11. removed conditional - replaced comparison check with true → KILLED
		12. removed conditional - replaced comparison check with true → SURVIVED
		13. replaced return of integer sized value with (x == 0 ? 1 : 0) → KILLED
		14. replaced return of integer sized value with (x == 0 ? 1 : 0) → KILLED
		15. Negated double local variable number 1 → KILLED
		16. Negated double field upper → KILLED

17. Negated double local variable number 3 → KILLED
18. Negated double local variable number 1 → SURVIVED
19. Substituted 0 with 1 → KILLED
20. Substituted 1 with 0 → KILLED
21. Substituted 1 with -1 → SURVIVED
22. Substituted 0 with -1 → KILLED
23. Substituted 1 with -1 → SURVIVED
24. Substituted 1 with 2 → KILLED
25. Substituted 0 with 1 → KILLED
26. Substituted 1 with 0 → KILLED
27. Substituted 0 with -1 → KILLED
28. greater or equal to less than → KILLED
29. Less than to less or equal → KILLED
30. greater or equal to less or equal → KILLED
31. Less than to greater than → KILLED
32. greater or equal to greater than → KILLED
33. Less than to greater or equal → KILLED
34. greater or equal to equal → SURVIVED
35. Less than to equal → KILLED
36. greater or equal to not equal → KILLED
37. Less than to not equal → KILLED
38. Incremented (a++) double local variable number 1 → KILLED
39. Incremented (a++) double field upper → SURVIVED
40. Incremented (a++) double local variable number 3 → SURVIVED
41. Incremented (a++) double local variable number 1 → SURVIVED
42. Decremented (a--) double local variable number 1 → SURVIVED
43. Decremented (a--) double field upper → SURVIVED
44. Decremented (a--) double local variable number 3 → SURVIVED
45. Decremented (a--) double local variable number 1 → SURVIVED
46. Incremented (++a) double local variable number 1 → KILLED
47. Incremented (++a) double field upper → KILLED
48. Incremented (++a) double local variable number 3 → SURVIVED
49. Incremented (++a) double local variable number 1 → KILLED
50. Decremented (--a) double local variable number 1 → KILLED
51. Decremented (--a) double fieldupper → KILLED
52. Decremented (--a) double local variable number 3 → KILLED
53. Decremented (--a) double local variable number 1 → SURVIVED

| 176 | | 1. replaced boolean return with false for org/jfree/data/Range::intersects → NO_COVERAGE
2. replaced boolean return with true for org/jfree/data/Range::intersects → NO_COVERAGE
3. removed call to org/jfree/data/Range::getLowerBound → NO_COVERAGE
4. removed call to org/jfree/data/Range::getUpperBound → NO_COVERAGE
5. removed call to org/jfree/data/Range::intersects → |

		NO_COVERAGE 6. replaced return of integer sized value with (x == 0 ? 1 : 0) → NO_COVERAGE
188		1. Negated double local variable number 1 → NO_COVERAGE 2. Incremented (a++) double local variable number 1 → NO_COVERAGE 3. Decremented (a--) double local variable number 1 → NO_COVERAGE 4. Incremented (++a) double local variable number 1 → NO_COVERAGE 5. Decremented (--a) double local variable number 1 → NO_COVERAGE
189		1. negated conditional → NO_COVERAGE 2. removed call to org/jfree/data/Range::contains → NO_COVERAGE 3. removed conditional - replaced equality check with false → NO_COVERAGE 4. removed conditional - replaced equality check with true → NO_COVERAGE 5. Negated double local variable number 1 → NO_COVERAGE 6. not equal to less than → NO_COVERAGE 7. not equal to less or equal → NO_COVERAGE 8. not equal to greater than → NO_COVERAGE 9. not equal to greater or equal → NO_COVERAGE 10. not equal to equal → NO_COVERAGE 11. Incremented (a++) double local variable number 1 → NO_COVERAGE 12. Decremented (a--) double local variable number 1 → NO_COVERAGE 13. Incremented (++a) double local variable number 1 → NO_COVERAGE 14. Decremented (--a) double local variable number 1 → NO_COVERAGE
190		1. changed conditional boundary → NO_COVERAGE 2. negated conditional → NO_COVERAGE 3. removed conditional - replaced comparison check with false → NO_COVERAGE 4. removed conditional - replaced comparison check with true → NO_COVERAGE 5. Negated double local variable number 1 → NO_COVERAGE 6. Negated double field upper → NO_COVERAGE 7. Less or equal to less than → NO_COVERAGE 8. Less or equal to greater than → NO_COVERAGE 9. Less or equal to greater or equal → NO_COVERAGE 10. Less or equal to equal → NO_COVERAGE 11. Less or equal to not equal → NO_COVERAGE 12. Incremented (a++) double local variable number 1 → NO_COVERAGE 13. Incremented (a++) double field upper → NO_COVERAGE 14. Decremented (a--) double local variable number 1 → NO_COVERAGE 15. Decremented (a--) double field upper → NO_COVERAGE 16. Incremented (++a) double local variable number 1 → NO_COVERAGE 17. Incremented (++a) double field upper → NO_COVERAGE 18. Decremented (--a) double local variable number 1 → NO_COVERAGE

		19. Decremented (--a) double fieldupper → NO_COVERAGE
191		1. Negated double field upper → NO_COVERAGE 2. Incremented (a++) double field upper → NO_COVERAGE 3. Decremented (a--) double field upper → NO_COVERAGE 4. Incremented (++a) double field upper → NO_COVERAGE 5. Decremented (--a) double fieldupper → NO_COVERAGE
193		1. changed conditional boundary → NO_COVERAGE 2. negated conditional → NO_COVERAGE 3. removed conditional - replaced comparison check with false → NO_COVERAGE 4. removed conditional - replaced comparison check with true → NO_COVERAGE 5. Negated double local variable number 1 → NO_COVERAGE 6. Negated double field lower → NO_COVERAGE 7. greater or equal to less than → NO_COVERAGE 8. greater or equal to less or equal → NO_COVERAGE 9. greater or equal to greater than → NO_COVERAGE 10. greater or equal to equal → NO_COVERAGE 11. greater or equal to not equal → NO_COVERAGE 12. Incremented (a++) double local variable number 1 → NO_COVERAGE 13. Incremented (a++) double field lower → NO_COVERAGE 14. Decremented (a--) double local variable number 1 → NO_COVERAGE 15. Decremented (a--) double field lower → NO_COVERAGE 16. Incremented (++a) double local variable number 1 → NO_COVERAGE 17. Incremented (++a) double field lower → NO_COVERAGE 18. Decremented (--a) double local variable number 1 → NO_COVERAGE 19. Decremented (--a) double fieldlower → NO_COVERAGE
194		1. Negated double field lower → NO_COVERAGE 2. Incremented (a++) double field lower → NO_COVERAGE 3. Decremented (a--) double field lower → NO_COVERAGE 4. Incremented (++a) double field lower → NO_COVERAGE 5. Decremented (--a) double fieldlower → NO_COVERAGE
197		1. replaced double return with 0.0d for org/jfree/data/Range::constrain → NO_COVERAGE 2. replaced return of double value with -(x + 1) for org/jfree/data/Range::constrain → NO_COVERAGE 3. Negated double local variable number 3 → NO_COVERAGE 4. Incremented (a++) double local variable number 3 → NO_COVERAGE 5. Decremented (a--) double local variable number 3 → NO_COVERAGE 6. Incremented (++a) double local variable number 3 → NO_COVERAGE 7. Decremented (--a) double local variable number 3 → NO_COVERAGE
217		1. negated conditional → NO_COVERAGE 2. removed conditional - replaced equality check with false → NO_COVERAGE 3. removed conditional - replaced equality check with true → NO_COVERAGE 4. not equal to equal → NO_COVERAGE
218		1. replaced return value with null for

		org/jfree/data/Range::combine → NO_COVERAGE 2. mutated return of Object value for org/jfree/data/Range::combine to (if (x != null) null else throw new RuntimeException) → NO_COVERAGE
220		1. negated conditional → NO_COVERAGE 2. removed conditional - replaced equality check with false → NO_COVERAGE 3. removed conditional - replaced equality check with true → NO_COVERAGE 4. not equal to equal → NO_COVERAGE
221		1. replaced return value with null for org/jfree/data/Range::combine → NO_COVERAGE 2. mutated return of Object value for org/jfree/data/Range::combine to (if (x != null) null else throw new RuntimeException) → NO_COVERAGE
223		1. replaced call to java/lang/Math::min with argument → NO_COVERAGE 2. removed call to org/jfree/data/Range::getLowerBound → NO_COVERAGE 3. removed call to org/jfree/data/Range::getLowerBound → NO_COVERAGE 4. removed call to java/lang/Math::min → NO_COVERAGE
224		1. replaced call to java/lang/Math::max with argument → NO_COVERAGE 2. removed call to org/jfree/data/Range::getUpperBound → NO_COVERAGE 3. removed call to org/jfree/data/Range::getUpperBound → NO_COVERAGE 4. removed call to java/lang/Math::max → NO_COVERAGE
225		1. removed call to org/jfree/data/Range::<init> → NO_COVERAGE 2. replaced return value with null for org/jfree/data/Range::combine → NO_COVERAGE 3. mutated return of Object value for org/jfree/data/Range::combine to (if (x != null) null else throw new RuntimeException) → NO_COVERAGE 4. Negated double local variable number 2 → NO_COVERAGE 5. Negated double local variable number 4 → NO_COVERAGE 6. Incremented (a++) double local variable number 2 → NO_COVERAGE 7. Incremented (a++) double local variable number 4 → NO_COVERAGE 8. Decremented (a--) double local variable number 2 → NO_COVERAGE 9. Decremented (a--) double local variable number 4 → NO_COVERAGE 10. Incremented (++a) double local variable number 2 → NO_COVERAGE 11. Incremented (++a) double local variable number 4 → NO_COVERAGE 12. Decremented (--a) double local variable number 2 → NO_COVERAGE 13. Decremented (--a) double local variable number 4 → NO_COVERAGE
241		1. negated conditional → KILLED 2. removed conditional - replaced equality check with false

		→ KILLED 3. removed conditional - replaced equality check with true → KILLED 4. not equal to equal → KILLED
242		1. negated conditional → KILLED 2. negated conditional → KILLED 3. removed call to org/jfree/data/Range::isNaNRange → KILLED 4. removed conditional - replaced equality check with false → KILLED 5. removed conditional - replaced equality check with false → KILLED 6. removed conditional - replaced equality check with true → KILLED 7. removed conditional - replaced equality check with true → KILLED 8. equal to less than → KILLED 9. equal to less or equal → SURVIVED 10. equal to greater than → KILLED 11. equal to greater or equal → KILLED 12. equal to not equal → KILLED 13. equal to not equal → KILLED
243		1. mutated return of Object value for org/jfree/data/Range::combineIgnoringNaN to (if (x != null) null else throw new RuntimeException) → KILLED
245		1. replaced return value with null for org/jfree/data/Range::combineIgnoringNaN → KILLED 2. mutated return of Object value for org/jfree/data/Range::combineIgnoringNaN to (if (x != null) null else throw new RuntimeException) → KILLED
247		1. negated conditional → KILLED 2. removed conditional - replaced equality check with false → KILLED 3. removed conditional - replaced equality check with true → KILLED 4. not equal to equal → KILLED
248		1. negated conditional → KILLED 2. removed call to org/jfree/data/Range::isNaNRange → KILLED 3. removed conditional - replaced equality check with false → KILLED 4. removed conditional - replaced equality check with true → KILLED 5. equal to less than → KILLED 6. equal to less or equal → SURVIVED 7. equal to greater than → KILLED 8. equal to greater or equal → KILLED 9. equal to not equal → KILLED
249		1. mutated return of Object value for org/jfree/data/Range::combineIgnoringNaN to (if (x != null) null else throw new RuntimeException) → KILLED
251		1. replaced return value with null for org/jfree/data/Range::combineIgnoringNaN → KILLED 2. mutated return of Object value for org/jfree/data/Range::combineIgnoringNaN to (if (x != null) null else throw new RuntimeException) → KILLED

253		1. replaced call to org/jfree/data/Range::min with argument → KILLED 2. removed call to org/jfree/data/Range::getLowerBound → KILLED 3. removed call to org/jfree/data/Range::getLowerBound → KILLED 4. removed call to org/jfree/data/Range::min → KILLED
254		1. replaced call to org/jfree/data/Range::max with argument → KILLED 2. removed call to org/jfree/data/Range::getUpperBound → KILLED 3. removed call to org/jfree/data/Range::getUpperBound → KILLED 4. removed call to org/jfree/data/Range::max → KILLED
255		1. negated conditional → KILLED 2. negated conditional → KILLED 3. removed call to java/lang/Double::isNaN → KILLED 4. removed call to java/lang/Double::isNaN → KILLED 5. removed conditional - replaced equality check with false → KILLED 6. removed conditional - replaced equality check with false → KILLED 7. removed conditional - replaced equality check with true → SURVIVED 8. removed conditional - replaced equality check with true → SURVIVED 9. Negated double local variable number 2 → SURVIVED 10. Negated double local variable number 4 → SURVIVED 11. equal to less than → SURVIVED 12. equal to less than → SURVIVED 13. equal to less or equal → SURVIVED 14. equal to less or equal → SURVIVED 15. equal to greater than → KILLED 16. equal to greater than → KILLED 17. equal to greater or equal → KILLED 18. equal to greater or equal → KILLED 19. equal to not equal → KILLED 20. equal to not equal → KILLED 21. Incremented (a++) double local variable number 2 → KILLED 22. Incremented (a++) double local variable number 4 → SURVIVED 23. Decremented (a--) double local variable number 2 → KILLED 24. Decremented (a--) double local variable number 4 → SURVIVED 25. Incremented (++a) double local variable number 2 → KILLED 26. Incremented (++a) double local variable number 4 → SURVIVED 27. Decremented (--a) double local variable number 2 → KILLED 28. Decremented (--a) double local variable number 4 → SURVIVED
256		1. mutated return of Object value for org/jfree/data/Range::combineIgnoringNaN to (if (x != null) null else throw new RuntimeException) → KILLED

258		1. removed call to org/jfree/data/Range::<init> → KILLED 2. replaced return value with null for org/jfree/data/Range::combineIgnoringNaN → KILLED 3. mutated return of Object value for org/jfree/data/Range::combineIgnoringNaN to (if (x != null) null else throw new RuntimeException) → KILLED 4. Negated double local variable number 2 → KILLED 5. Negated double local variable number 4 → KILLED 6. Incremented (a++) double local variable number 2 → SURVIVED 7. Incremented (a++) double local variable number 4 → SURVIVED 8. Decremented (a--) double local variable number 2 → SURVIVED 9. Decremented (a--) double local variable number 4 → SURVIVED 10. Incremented (++a) double local variable number 2 → KILLED 11. Incremented (++a) double local variable number 4 → KILLED 12. Decremented (--a) double local variable number 2 → KILLED 13. Decremented (--a) double local variable number 4 → KILLED
271		1. negated conditional → KILLED 2. removed call to java/lang/Double::isNaN → SURVIVED 3. removed conditional - replaced equality check with false → SURVIVED 4. removed conditional - replaced equality check with true → KILLED 5. Negated double local variable number 0 → SURVIVED 6. equal to less than → KILLED 7. equal to less or equal → SURVIVED 8. equal to greater than → KILLED 9. equal to greater or equal → SURVIVED 10. equal to not equal → KILLED 11. Incremented (a++) double local variable number 0 → KILLED 12. Decremented (a--) double local variable number 0 → KILLED 13. Incremented (++a) double local variable number 0 → KILLED 14. Decremented (--a) double local variable number 0 → KILLED
272		1. replaced double return with 0.0d for org/jfree/data/Range::min → KILLED 2. replaced return of double value with -(x + 1) for org/jfree/data/Range::min → KILLED 3. Negated double local variable number 2 → SURVIVED 4. Incremented (a++) double local variable number 2 → SURVIVED 5. Decremented (a--) double local variable number 2 → SURVIVED 6. Incremented (++a) double local variable number 2 → SURVIVED 7. Decremented (--a) double local variable number 2 → SURVIVED

274		1. negated conditional → KILLED 2. removed call to java/lang/Double::isNaN → KILLED 3. removed conditional - replaced equality check with false → KILLED 4. removed conditional - replaced equality check with true → KILLED 5. Negated double local variable number 2 → SURVIVED 6. equal to less than → KILLED 7. equal to less or equal → SURVIVED 8. equal to greater than → KILLED 9. equal to greater or equal → KILLED 10. equal to not equal → KILLED 11. Incremented (a++) double local variable number 2 → KILLED 12. Decremented (a--) double local variable number 2 → KILLED 13. Incremented (++a) double local variable number 2 → KILLED 14. Decremented (--a) double local variable number 2 → KILLED
275		1. replaced double return with 0.0d for org/jfree/data/Range::min → KILLED 2. replaced return of double value with -(x + 1) for org/jfree/data/Range::min → KILLED 3. Negated double local variable number 0 → KILLED 4. Incremented (a++) double local variable number 0 → SURVIVED 5. Decremented (a--) double local variable number 0 → SURVIVED 6. Incremented (++a) double local variable number 0 → KILLED 7. Decremented (--a) double local variable number 0 → KILLED
277		1. replaced call to java/lang/Math::min with argument → KILLED 2. removed call to java/lang/Math::min → KILLED 3. replaced double return with 0.0d for org/jfree/data/Range::min → KILLED 4. replaced return of double value with -(x + 1) for org/jfree/data/Range::min → KILLED 5. Negated double local variable number 0 → KILLED 6. Negated double local variable number 2 → KILLED 7. Incremented (a++) double local variable number 0 → SURVIVED 8. Incremented (a++) double local variable number 2 → SURVIVED 9. Decremented (a--) double local variable number 0 → SURVIVED 10. Decremented (a--) double local variable number 2 → SURVIVED 11. Incremented (++a) double local variable number 0 → KILLED 12. Incremented (++a) double local variable number 2 → KILLED 13. Decremented (--a) double local variable number 0 → KILLED 14. Decremented (--a) double local variable number 2 → KILLED

281		1. negated conditional → KILLED 2. removed call to java/lang/Double::isNaN → SURVIVED 3. removed conditional - replaced equality check with false → SURVIVED 4. removed conditional - replaced equality check with true → KILLED 5. Negated double local variable number 0 → SURVIVED 6. equal to less than → KILLED 7. equal to less or equal → SURVIVED 8. equal to greater than → KILLED 9. equal to greater or equal → SURVIVED 10. equal to not equal → KILLED 11. Incremented (a++) double local variable number 0 → KILLED 12. Decremented (a--) double local variable number 0 → KILLED 13. Incremented (++a) double local variable number 0 → KILLED 14. Decremented (--a) double local variable number 0 → KILLED
282		1. replaced double return with 0.0d for org/jfree/data/Range::max → KILLED 2. replaced return of double value with -(x + 1) for org/jfree/data/Range::max → KILLED 3. Negated double local variable number 2 → SURVIVED 4. Incremented (a++) double local variable number 2 → SURVIVED 5. Decremented (a--) double local variable number 2 → SURVIVED 6. Incremented (++a) double local variable number 2 → SURVIVED 7. Decremented (--a) double local variable number 2 → SURVIVED
284		1. negated conditional → KILLED 2. removed call to java/lang/Double::isNaN → KILLED 3. removed conditional - replaced equality check with false → KILLED 4. removed conditional - replaced equality check with true → KILLED 5. Negated double local variable number 2 → SURVIVED 6. equal to less than → KILLED 7. equal to less or equal → SURVIVED 8. equal to greater than → KILLED 9. equal to greater or equal → KILLED 10. equal to not equal → KILLED 11. Incremented (a++) double local variable number 2 → KILLED 12. Decremented (a--) double local variable number 2 → KILLED 13. Incremented (++a) double local variable number 2 → KILLED 14. Decremented (--a) double local variable number 2 → KILLED
285		1. replaced double return with 0.0d for org/jfree/data/Range::max → KILLED 2. replaced return of double value with -(x + 1) for org/jfree/data/Range::max → KILLED

		3. Negated double local variable number 0 → KILLED 4. Incremented (a++) double local variable number 0 → SURVIVED 5. Decremented (a--) double local variable number 0 → SURVIVED 6. Incremented (++a) double local variable number 0 → KILLED 7. Decremented (--a) double local variable number 0 → KILLED
287		1. replaced call to java/lang/Math::max with argument → KILLED 2. removed call to java/lang/Math::max → KILLED 3. replaced double return with 0.0d for org/jfree/data/Range::max → KILLED 4. replaced return of double value with -(x + 1) for org/jfree/data/Range::max → KILLED 5. Negated double local variable number 0 → KILLED 6. Negated double local variable number 2 → KILLED 7. Incremented (a++) double local variable number 0 → SURVIVED 8. Incremented (a++) double local variable number 2 → SURVIVED 9. Decremented (a--) double local variable number 0 → SURVIVED 10. Decremented (a--) double local variable number 2 → SURVIVED 11. Incremented (++a) double local variable number 0 → KILLED 12. Incremented (++a) double local variable number 2 → KILLED 13. Decremented (--a) double local variable number 0 → KILLED 14. Decremented (--a) double local variable number 2 → KILLED
302		1. negated conditional → KILLED 2. removed conditional - replaced equality check with false → KILLED 3. removed conditional - replaced equality check with true → KILLED 4. not equal to equal → KILLED
303		1. removed call to org/jfree/data/Range::<init> → KILLED 2. replaced return value with null for org/jfree/data/Range::expandToInclude → KILLED 3. mutated return of Object value for org/jfree/data/Range::expandToInclude to (if (x != null) null else throw new RuntimeException) → KILLED 4. Negated double local variable number 1 → KILLED 5. Negated double local variable number 1 → KILLED 6. Incremented (a++) double local variable number 1 → KILLED 7. Incremented (a++) double local variable number 1 → SURVIVED 8. Decremented (a--) double local variable number 1 → KILLED 9. Decremented (a--) double local variable number 1 → SURVIVED 10. Incremented (++a) double local variable number 1 →

		KILLED 11. Incremented (++a) double local variable number 1 → KILLED 12. Decremented (--a) double local variable number 1 → KILLED 13. Decremented (--a) double local variable number 1 → KILLED
305		1. changed conditional boundary → SURVIVED 2. negated conditional → KILLED 3. removed call to org/jfree/data/Range::getLowerBound → KILLED 4. removed conditional - replaced comparison check with false → KILLED 5. removed conditional - replaced comparison check with true → KILLED 6. Negated double local variable number 1 → KILLED 7. greater or equal to less than → KILLED 8. greater or equal to less or equal → KILLED 9. greater or equal to greater than → SURVIVED 10. greater or equal to equal → KILLED 11. greater or equal to not equal → KILLED 12. Incremented (a++) double local variable number 1 → KILLED 13. Decremented (a--) double local variable number 1 → KILLED 14. Incremented (++a) double local variable number 1 → KILLED 15. Decremented (--a) double local variable number 1 → KILLED
306		1. removed call to org/jfree/data/Range::<init> → KILLED 2. removed call to org/jfree/data/Range::getUpperBound → KILLED 3. replaced return value with null for org/jfree/data/Range::expandToInclude → KILLED 4. mutated return of Object value for org/jfree/data/Range::expandToInclude to (if (x != null) null else throw new RuntimeException) → KILLED 5. Negated double local variable number 1 → KILLED 6. Incremented (a++) double local variable number 1 → SURVIVED 7. Decremented (a--) double local variable number 1 → SURVIVED 8. Incremented (++a) double local variable number 1 → KILLED 9. Decremented (--a) double local variable number 1 → KILLED
308		1. changed conditional boundary → SURVIVED 2. negated conditional → KILLED 3. removed call to org/jfree/data/Range::getUpperBound → KILLED 4. removed conditional - replaced comparison check with false → KILLED 5. removed conditional - replaced comparison check with true → KILLED 6. Negated double local variable number 1 → KILLED 7. Less or equal to less than → SURVIVED 8. Less or equal to greater than → KILLED

		9. Less or equal to greater or equal → KILLED 10. Less or equal to equal → KILLED 11. Less or equal to not equal → KILLED 12. Incremented (a++) double local variable number 1 → KILLED 13. Decremented (a--) double local variable number 1 → KILLED 14. Incremented (++a) double local variable number 1 → KILLED 15. Decremented (--a) double local variable number 1 → KILLED
309		1. removed call to org/jfree/data/Range::<init> → KILLED 2. removed call to org/jfree/data/Range::getLowerBound → KILLED 3. replaced return value with null for org/jfree/data/Range::expandToInclude → KILLED 4. mutated return of Object value for org/jfree/data/Range::expandToInclude to (if (x != null) null else throw new RuntimeException) → KILLED 5. Negated double local variable number 1 → KILLED 6. Incremented (a++) double local variable number 1 → SURVIVED 7. Decremented (a--) double local variable number 1 → SURVIVED 8. Incremented (++a) double local variable number 1 → KILLED 9. Decremented (--a) double local variable number 1 → KILLED
312		1. replaced return value with null for org/jfree/data/Range::expandToInclude → KILLED 2. mutated return of Object value for org/jfree/data/Range::expandToInclude to (if (x != null) null else throw new RuntimeException) → KILLED
329		1. removed call to org/jfree/chart/util/ParamChecks::nullNotPermitted → NO_COVERAGE
330		1. removed call to org/jfree/data/Range::getLength → NO_COVERAGE
331		1. Replaced double multiplication with division → NO_COVERAGE 2. Replaced double subtraction with addition → NO_COVERAGE 3. removed call to org/jfree/data/Range::getLowerBound → NO_COVERAGE 4. Negated double local variable number 5 → NO_COVERAGE 5. Negated double local variable number 1 → NO_COVERAGE 6. Replaced double operation with first member → NO_COVERAGE 7. Replaced double operation with first member → NO_COVERAGE 8. Replaced double operation by second member → NO_COVERAGE 9. Replaced double operation by second member → NO_COVERAGE 10. Replaced double multiplication with division → NO_COVERAGE 11. Replaced double subtraction with addition → NO_COVERAGE 12. Replaced double multiplication with modulus → NO_COVERAGE 13. Replaced double subtraction with multiplication →

		NO_COVERAGE 14. Replaced double multiplication with addition → NO_COVERAGE 15. Replaced double subtraction with division → NO_COVERAGE 16. Replaced double multiplication with subtraction → NO_COVERAGE 17. Replaced double subtraction with modulus → NO_COVERAGE 18. Incremented (a++) double local variable number 5 → NO_COVERAGE 19. Incremented (a++) double local variable number 1 → NO_COVERAGE 20. Decremented (a--) double local variable number 5 → NO_COVERAGE 21. Decremented (a--) double local variable number 1 → NO_COVERAGE 22. Incremented (++a) double local variable number 5 → NO_COVERAGE 23. Incremented (++a) double local variable number 1 → NO_COVERAGE 24. Decremented (--a) double local variable number 5 → NO_COVERAGE 25. Decremented (--a) double local variable number 1 → NO_COVERAGE
332		1. Replaced double multiplication with division → NO_COVERAGE 2. Replaced double addition with subtraction → NO_COVERAGE 3. removed call to org/jfree/data/Range::getUpperBound → NO_COVERAGE 4. Negated double local variable number 5 → NO_COVERAGE 5. Negated double local variable number 3 → NO_COVERAGE 6. Replaced double operation with first member → NO_COVERAGE 7. Replaced double operation with first member → NO_COVERAGE 8. Replaced double operation by second member → NO_COVERAGE 9. Replaced double operation by second member → NO_COVERAGE 10. Replaced double multiplication with division → NO_COVERAGE 11. Replaced double addition with subtraction → NO_COVERAGE 12. Replaced double multiplication with modulus → NO_COVERAGE 13. Replaced double addition with multiplication → NO_COVERAGE 14. Replaced double multiplication with addition → NO_COVERAGE 15. Replaced double addition with division → NO_COVERAGE 16. Replaced double multiplication with subtraction → NO_COVERAGE 17. Replaced double addition with modulus → NO_COVERAGE 18. Incremented (a++) double local variable number 5 → NO_COVERAGE 19. Incremented (a++) double local variable number 3 → NO_COVERAGE 20. Decremented (a--) double local variable number 5 → NO_COVERAGE 21. Decremented (a--) double local variable number 3 → NO_COVERAGE 22. Incremented (++a) double local variable number 5 →

		NO_COVERAGE 23. Incremented (++a) double local variable number 3 → NO_COVERAGE 24. Decremented (--a) double local variable number 5 → NO_COVERAGE 25. Decremented (--a) double local variable number 3 → NO_COVERAGE
333		1. changed conditional boundary → NO_COVERAGE 2. negated conditional → NO_COVERAGE 3. removed conditional - replaced comparison check with false → NO_COVERAGE 4. removed conditional - replaced comparison check with true → NO_COVERAGE 5. Negated double local variable number 7 → NO_COVERAGE 6. Negated double local variable number 9 → NO_COVERAGE 7. Less or equal to less than → NO_COVERAGE 8. Less or equal to greater than → NO_COVERAGE 9. Less or equal to greater or equal → NO_COVERAGE 10. Less or equal to equal → NO_COVERAGE 11. Less or equal to not equal → NO_COVERAGE 12. Incremented (a++) double local variable number 7 → NO_COVERAGE 13. Incremented (a++) double local variable number 9 → NO_COVERAGE 14. Decremented (a--) double local variable number 7 → NO_COVERAGE 15. Decremented (a--) double local variable number 9 → NO_COVERAGE 16. Incremented (++a) double local variable number 7 → NO_COVERAGE 17. Incremented (++a) double local variable number 9 → NO_COVERAGE 18. Decremented (--a) double local variable number 7 → NO_COVERAGE 19. Decremented (--a) double local variable number 9 → NO_COVERAGE
334		1. Substituted 2.0 with 1.0 → NO_COVERAGE 2. Substituted 2.0 with 1.0 → NO_COVERAGE 3. Replaced double division with multiplication → NO_COVERAGE 4. Replaced double division with multiplication → NO_COVERAGE 5. Replaced double addition with subtraction → NO_COVERAGE 6. Negated double local variable number 7 → NO_COVERAGE 7. Negated double local variable number 9 → NO_COVERAGE 8. Replaced double operation with first member → NO_COVERAGE 9. Replaced double operation with first member → NO_COVERAGE 10. Replaced double operation with first member → NO_COVERAGE 11. Replaced double operation by second member → NO_COVERAGE 12. Replaced double operation by second member → NO_COVERAGE 13. Replaced double operation by second member → NO_COVERAGE 14. Replaced double division with multiplication →

		NO_COVERAGE 15. Replaced double division with multiplication → NO_COVERAGE 16. Replaced double addition with subtraction → NO_COVERAGE 17. Replaced double division with modulus → NO_COVERAGE 18. Replaced double division with modulus → NO_COVERAGE 19. Replaced double addition with multiplication → NO_COVERAGE 20. Replaced double division with addition → NO_COVERAGE 21. Replaced double division with addition → NO_COVERAGE 22. Replaced double addition with division → NO_COVERAGE 23. Replaced double division with subtraction → NO_COVERAGE 24. Replaced double division with subtraction → NO_COVERAGE 25. Replaced double addition with modulus → NO_COVERAGE 26. Substituted 2.0 with 1.0 → NO_COVERAGE 27. Substituted 2.0 with 1.0 → NO_COVERAGE 28. Substituted 2.0 with 0.0 → NO_COVERAGE 29. Substituted 2.0 with 0.0 → NO_COVERAGE 30. Substituted 2.0 with -1.0 → NO_COVERAGE 31. Substituted 2.0 with -1.0 → NO_COVERAGE 32. Substituted 2.0 with -2.0 → NO_COVERAGE 33. Substituted 2.0 with -2.0 → NO_COVERAGE 34. Substituted 2.0 with 3.0 → NO_COVERAGE 35. Substituted 2.0 with 3.0 → NO_COVERAGE 36. Substituted 2.0 with 1.0 → NO_COVERAGE 37. Substituted 2.0 with 1.0 → NO_COVERAGE 38. Incremented (a++) double local variable number 7 → NO_COVERAGE 39. Incremented (a++) double local variable number 9 → NO_COVERAGE 40. Decremented (a--) double local variable number 7 → NO_COVERAGE 41. Decremented (a--) double local variable number 9 → NO_COVERAGE 42. Incremented (++a) double local variable number 7 → NO_COVERAGE 43. Incremented (++a) double local variable number 9 → NO_COVERAGE 44. Decremented (--a) double local variable number 7 → NO_COVERAGE 45. Decremented (--a) double local variable number 9 → NO_COVERAGE
335		1. Negated double local variable number 7 → NO_COVERAGE 2. Incremented (a++) double local variable number 7 → NO_COVERAGE 3. Decremented (a--) double local variable number 7 → NO_COVERAGE 4. Incremented (++a) double local variable number 7 → NO_COVERAGE 5. Decremented (--a) double local variable number 7 → NO_COVERAGE
337		1. removed call to org/jfree/data/Range::<init> → NO_COVERAGE 2. replaced return value with null for org/jfree/data/Range::expand → NO_COVERAGE 3. mutated return of Object value for org/jfree/data/Range::expand to (if (x != null) null else throw new RuntimeException) → NO_COVERAGE

		4. Negated double local variable number 7 → NO_COVERAGE 5. Negated double local variable number 9 → NO_COVERAGE 6. Incremented (a++) double local variable number 7 → NO_COVERAGE 7. Incremented (a++) double local variable number 9 → NO_COVERAGE 8. Decremented (a--) double local variable number 7 → NO_COVERAGE 9. Decremented (a--) double local variable number 9 → NO_COVERAGE 10. Incremented (++a) double local variable number 7 → NO_COVERAGE 11. Incremented (++a) double local variable number 9 → NO_COVERAGE 12. Decremented (--a) double local variable number 7 → NO_COVERAGE 13. Decremented (--a) double local variable number 9 → NO_COVERAGE
349		1. replaced call to org/jfree/data/Range::shift with argument → NO_COVERAGE 2. Substituted 0 with 1 → NO_COVERAGE 3. removed call to org/jfree/data/Range::shift → NO_COVERAGE 4. replaced return value with null for org/jfree/data/Range::shift → NO_COVERAGE 5. mutated return of Object value for org/jfree/data/Range::shift to (if (x != null) null else throw new RuntimeException) → NO_COVERAGE 6. Negated double local variable number 1 → NO_COVERAGE 7. Substituted 0 with 1 → NO_COVERAGE 8. Substituted 0 with -1 → NO_COVERAGE 9. Substituted 0 with 1 → NO_COVERAGE 10. Substituted 0 with -1 → NO_COVERAGE 11. Incremented (a++) double local variable number 1 → NO_COVERAGE 12. Decremented (a--) double local variable number 1 → NO_COVERAGE 13. Incremented (++a) double local variable number 1 → NO_COVERAGE 14. Decremented (--a) double local variable number 1 → NO_COVERAGE
365		1. removed call to org/jfree/chart/util/ParamChecks::nullNotPermitted → SURVIVED
366		1. negated conditional → KILLED 2. removed conditional - replaced equality check with false → KILLED 3. removed conditional - replaced equality check with true → KILLED 4. Negated integer local variable number 3 → SURVIVED 5. equal to less than → KILLED 6. equal to less or equal → SURVIVED 7. equal to greater than → KILLED 8. equal to greater or equal → KILLED 9. equal to not equal → KILLED 10. Incremented (a++) integer local variable number 3 → SURVIVED

		11. Decremented (a--) integer local variable number 3 → SURVIVED 12. Incremented (++a) integer local variable number 3 → KILLED 13. Decremented (--a) integer local variable number 3 → KILLED
367		1. removed call to org/jfree/data/Range::<init> → KILLED 2. Replaced double addition with subtraction → KILLED 3. removed call to org/jfree/data/Range::getLowerBound → KILLED 4. replaced return value with null for org/jfree/data/Range::shift → KILLED 5. mutated return of Object value for org/jfree/data/Range::shift to (if (x != null) null else throw new RuntimeException) → KILLED 6. Negated double local variable number 1 → KILLED 7. Replaced double operation with first member → KILLED 8. Replaced double operation by second member → KILLED 9. Replaced double addition with subtraction → KILLED 10. Replaced double addition with multiplication → KILLED 11. Replaced double addition with division → KILLED 12. Replaced double addition with modulus → KILLED 13. Incremented (a++) double local variable number 1 → KILLED 14. Decremented (a--) double local variable number 1 → KILLED 15. Incremented (++a) double local variable number 1 → KILLED 16. Decremented (--a) double local variable number 1 → KILLED
368		1. Replaced double addition with subtraction → KILLED 2. removed call to org/jfree/data/Range::getUpperBound → KILLED 3. Negated double local variable number 1 → KILLED 4. Replaced double operation with first member → KILLED 5. Replaced double operation by second member → KILLED 6. Replaced double addition with subtraction → KILLED 7. Replaced double addition with multiplication → KILLED 8. Replaced double addition with division → KILLED 9. Replaced double addition with modulus → KILLED 10. Incremented (a++) double local variable number 1 → SURVIVED 11. Decremented (a--) double local variable number 1 → SURVIVED 12. Incremented (++a) double local variable number 1 → KILLED 13. Decremented (--a) double local variable number 1 → KILLED
371		1. replaced call to org/jfree/data/Range::shiftWithNoZeroCrossing with argument → KILLED 2. removed call to org/jfree/data/Range::<init> → KILLED 3. removed call to org/jfree/data/Range::getLowerBound → KILLED 4. removed call to org/jfree/data/Range::shiftWithNoZeroCrossing → KILLED 5. replaced return value with null for

		org/jfree/data/Range::shift → KILLED 6. mutated return of Object value for org/jfree/data/Range::shift to (if (x != null) null else throw new RuntimeException) → KILLED
372		1. replaced call to org/jfree/data/Range::shiftWithNoZeroCrossing with argument → KILLED 2. removed call to org/jfree/data/Range::getUpperBound → KILLED 3. removed call to org/jfree/data/Range::shiftWithNoZeroCrossing → KILLED 4. Negated double local variable number 1 → KILLED 5. Incremented (a++) double local variable number 1 → KILLED 6. Decremented (a--) double local variable number 1 → KILLED 7. Incremented (++a) double local variable number 1 → KILLED 8. Decremented (--a) double local variable number 1 → KILLED
373		1. Negated double local variable number 1 → KILLED 2. Incremented (a++) double local variable number 1 → SURVIVED 3. Decremented (a--) double local variable number 1 → SURVIVED 4. Incremented (++a) double local variable number 1 → KILLED 5. Decremented (--a) double local variable number 1 → KILLED
387		1. changed conditional boundary → KILLED 2. Substituted 0.0 with 1.0 → SURVIVED 3. negated conditional → KILLED 4. removed conditional - replaced comparison check with false → KILLED 5. removed conditional - replaced comparison check with true → KILLED 6. Negated double local variable number 0 → KILLED 7. Substituted 0.0 with 1.0 → SURVIVED 8. Substituted 0.0 with -1.0 → KILLED 9. Substituted 0.0 with 1.0 → SURVIVED 10. Substituted 0.0 with -1.0 → KILLED 11. Less or equal to less than → KILLED 12. Less or equal to greater than → KILLED 13. Less or equal to greater or equal → KILLED 14. Less or equal to equal → KILLED 15. Less or equal to not equal → KILLED 16. Incremented (a++) double local variable number 0 → KILLED 17. Decremented (a--) double local variable number 0 → KILLED 18. Incremented (++a) double local variable number 0 → KILLED 19. Decremented (--a) double local variable number 0 → KILLED
388		1. replaced call to java/lang/Math::max with argument → KILLED 2. Substituted 0.0 with 1.0 → KILLED

3. Replaced double addition with subtraction → KILLED
4. removed call to java/lang/Math::max → KILLED
5. replaced double return with 0.0d for
org/jfree/data/Range::shiftWithNoZeroCrossing → KILLED
6. replaced return of double value with -(x + 1) for
org/jfree/data/Range::shiftWithNoZeroCrossing → KILLED
7. Negated double local variable number 0 → KILLED
8. Negated double local variable number 2 → KILLED
9. Replaced double operation with first member → KILLED
10. Replaced double operation by second member → KILLED
11. Replaced double addition with subtraction → KILLED
12. Replaced double addition with multiplication → KILLED
13. Replaced double addition with division → KILLED
14. Replaced double addition with modulus → KILLED
15. Substituted 0.0 with 1.0 → KILLED
16. Substituted 0.0 with -1.0 → KILLED
17. Substituted 0.0 with 1.0 → KILLED
18. Substituted 0.0 with -1.0 → KILLED
19. Incremented (a++) double local variable number 0 →
SURVIVED
20. Incremented (a++) double local variable number 2 →
SURVIVED
21. Decremented (a--) double local variable number 0 →
SURVIVED
22. Decremented (a--) double local variable number 2 →
SURVIVED
23. Incremented (++a) double local variable number 0 →
KILLED
24. Incremented (++a) double local variable number 2 →
KILLED
25. Decremented (--a) double local variable number 0 →
KILLED
26. Decremented (--a) double local variable number 2 →
KILLED

390

1. changed conditional boundary → KILLED
2. Substituted 0.0 with 1.0 → KILLED
3. negated conditional → KILLED
4. removed conditional - replaced comparison check with
false → KILLED
5. removed conditional - replaced comparison check with
true → KILLED
6. Negated double local variable number 0 → KILLED
7. Substituted 0.0 with 1.0 → KILLED
8. Substituted 0.0 with -1.0 → SURVIVED
9. Substituted 0.0 with 1.0 → KILLED
10. Substituted 0.0 with -1.0 → SURVIVED
11. greater or equal to less than → KILLED
12. greater or equal to less or equal → KILLED
13. greater or equal to greater than → KILLED
14. greater or equal to equal → SURVIVED
15. greater or equal to not equal → KILLED
16. Incremented (a++) double local variable number 0 →
KILLED
17. Decremented (a--) double local variable number 0 →
KILLED
18. Incremented (++a) double local variable number 0 →
KILLED
19. Decremented (--a) double local variable number 0 →

		KILLED
391		1. replaced call to java/lang/Math::min with argument → KILLED 2. Substituted 0.0 with 1.0 → KILLED 3. Replaced double addition with subtraction → KILLED 4. removed call to java/lang/Math::min → KILLED 5. replaced double return with 0.0d for org/jfree/data/Range::shiftWithNoZeroCrossing → KILLED 6. replaced return of double value with -(x + 1) for org/jfree/data/Range::shiftWithNoZeroCrossing → KILLED 7. Negated double local variable number 0 → KILLED 8. Negated double local variable number 2 → KILLED 9. Replaced double operation with first member → KILLED 10. Replaced double operation by second member → KILLED 11. Replaced double addition with subtraction → KILLED 12. Replaced double addition with multiplication → KILLED 13. Replaced double addition with division → KILLED 14. Replaced double addition with modulus → KILLED 15. Substituted 0.0 with 1.0 → KILLED 16. Substituted 0.0 with -1.0 → KILLED 17. Substituted 0.0 with 1.0 → KILLED 18. Substituted 0.0 with -1.0 → KILLED 19. Incremented (a++) double local variable number 0 → SURVIVED 20. Incremented (a++) double local variable number 2 → SURVIVED 21. Decremented (a--) double local variable number 0 → SURVIVED 22. Decremented (a--) double local variable number 2 → SURVIVED 23. Incremented (++a) double local variable number 0 → KILLED 24. Incremented (++a) double local variable number 2 → KILLED 25. Decremented (--a) double local variable number 0 → KILLED 26. Decremented (--a) double local variable number 2 → KILLED
394		1. Replaced double addition with subtraction → KILLED 2. replaced double return with 0.0d for org/jfree/data/Range::shiftWithNoZeroCrossing → KILLED 3. replaced return of double value with -(x + 1) for org/jfree/data/Range::shiftWithNoZeroCrossing → KILLED 4. Negated double local variable number 0 → SURVIVED 5. Negated double local variable number 2 → KILLED 6. Replaced double operation with first member → KILLED 7. Replaced double operation by second member → SURVIVED 8. Replaced double addition with subtraction → KILLED 9. Replaced double addition with multiplication → KILLED 10. Replaced double addition with division → KILLED 11. Replaced double addition with modulus → KILLED 12. Incremented (a++) double local variable number 0 → SURVIVED 13. Incremented (a++) double local variable number 2 → SURVIVED 14. Decremented (a--) double local variable number 0 → SURVIVED 15. Decremented (a--) double local variable number 2 →

		SURVIVED 16. Incremented (++a) double local variable number 0 → KILLED 17. Incremented (++a) double local variable number 2 → KILLED 18. Decremented (--a) double local variable number 0 → KILLED 19. Decremented (--a) double local variable number 2 → KILLED
409		1. removed call to org/jfree/chart/util/ParamChecks::nullNotPermitted → NO_COVERAGE
410		1. changed conditional boundary → NO_COVERAGE 2. Substituted 0.0 with 1.0 → NO_COVERAGE 3. negated conditional → NO_COVERAGE 4. removed conditional - replaced comparison check with false → NO_COVERAGE 5. removed conditional - replaced comparison check with true → NO_COVERAGE 6. Negated double local variable number 1 → NO_COVERAGE 7. Substituted 0.0 with 1.0 → NO_COVERAGE 8. Substituted 0.0 with -1.0 → NO_COVERAGE 9. Substituted 0.0 with 1.0 → NO_COVERAGE 10. Substituted 0.0 with -1.0 → NO_COVERAGE 11. greater or equal to less than → NO_COVERAGE 12. greater or equal to less or equal → NO_COVERAGE 13. greater or equal to greater than → NO_COVERAGE 14. greater or equal to equal → NO_COVERAGE 15. greater or equal to not equal → NO_COVERAGE 16. Incremented (a++) double local variable number 1 → NO_COVERAGE 17. Decremented (a--) double local variable number 1 → NO_COVERAGE 18. Incremented (++a) double local variable number 1 → NO_COVERAGE 19. Decremented (--a) double local variable number 1 → NO_COVERAGE
411		1. removed call to java/lang/IllegalArgumentException::<init> → NO_COVERAGE
413		1. removed call to org/jfree/data/Range::<init> → NO_COVERAGE 2. Replaced double multiplication with division → NO_COVERAGE 3. removed call to org/jfree/data/Range::getLowerBound → NO_COVERAGE 4. replaced return value with null for org/jfree/data/Range::scale → NO_COVERAGE 5. mutated return of Object value for org/jfree/data/Range::scale to (if (x != null) null else throw new RuntimeException) → NO_COVERAGE 6. Negated double local variable number 1 → NO_COVERAGE 7. Replaced double operation with first member → NO_COVERAGE 8. Replaced double operation by second member → NO_COVERAGE 9. Replaced double multiplication with division → NO_COVERAGE 10. Replaced double multiplication with modulus →

		NO_COVERAGE 11. Replaced double multiplication with addition → NO_COVERAGE 12. Replaced double multiplication with subtraction → NO_COVERAGE 13. Incremented (a++) double local variable number 1 → NO_COVERAGE 14. Decremented (a--) double local variable number 1 → NO_COVERAGE 15. Incremented (++a) double local variable number 1 → NO_COVERAGE 16. Decremented (--a) double local variable number 1 → NO_COVERAGE
414		1. Replaced double multiplication with division → NO_COVERAGE 2. removed call to org/jfree/data/Range::getUpperBound → NO_COVERAGE 3. Negated double local variable number 1 → NO_COVERAGE 4. Replaced double operation with first member → NO_COVERAGE 5. Replaced double operation by second member → NO_COVERAGE 6. Replaced double multiplication with division → NO_COVERAGE 7. Replaced double multiplication with modulus → NO_COVERAGE 8. Replaced double multiplication with addition → NO_COVERAGE 9. Replaced double multiplication with subtraction → NO_COVERAGE 10. Incremented (a++) double local variable number 1 → NO_COVERAGE 11. Decremented (a--) double local variable number 1 → NO_COVERAGE 12. Incremented (++a) double local variable number 1 → NO_COVERAGE 13. Decremented (--a) double local variable number 1 → NO_COVERAGE
426		1. negated conditional → KILLED 2. removed conditional - replaced equality check with false → SURVIVED 3. removed conditional - replaced equality check with true → KILLED 4. not equal to less than → KILLED 5. not equal to less or equal → KILLED 6. not equal to greater than → SURVIVED 7. not equal to greater or equal → SURVIVED 8. not equal to equal → KILLED
427		1. replaced boolean return with true for org/jfree/data/Range::equals → NO_COVERAGE 2. Substituted 0 with 1 → NO_COVERAGE 3. replaced return of integer sized value with (x == 0 ? 1 : 0) → NO_COVERAGE 4. Substituted 0 with 1 → NO_COVERAGE 5. Substituted 0 with -1 → NO_COVERAGE 6. Substituted 0 with 1 → NO_COVERAGE 7. Substituted 0 with -1 → NO_COVERAGE
430		1. negated conditional → KILLED

		2. removed conditional - replaced equality check with false → SURVIVED
		3. removed conditional - replaced equality check with true → KILLED
		4. Negated double field lower → KILLED
		5. Negated double field lower → KILLED
		6. equal to less than → KILLED
		7. equal to less or equal → SURVIVED
		8. equal to greater than → KILLED
		9. equal to greater or equal → SURVIVED
		10. equal to not equal → KILLED
		11. Incremented (a++) double field lower → KILLED
		12. Incremented (a++) double field lower → SURVIVED
		13. Decremented (a--) double field lower → KILLED
		14. Decremented (a--) double field lower → SURVIVED
		15. Incremented (++a) double field lower → KILLED
		16. Incremented (++a) double field lower → KILLED
		17. Decremented (--a) double fieldlower → KILLED
		18. Decremented (--a) double fieldlower → KILLED
431		1. replaced boolean return with true for org/jfree/data/Range::equals → NO_COVERAGE
		2. Substituted 0 with 1 → NO_COVERAGE
		3. replaced return of integer sized value with (x == 0 ? 1 : 0) → NO_COVERAGE
		4. Substituted 0 with 1 → NO_COVERAGE
		5. Substituted 0 with -1 → NO_COVERAGE
		6. Substituted 0 with 1 → NO_COVERAGE
		7. Substituted 0 with -1 → NO_COVERAGE
433		1. negated conditional → KILLED
		2. removed conditional - replaced equality check with false → SURVIVED
		3. removed conditional - replaced equality check with true → KILLED
		4. Negated double field upper → KILLED
		5. Negated double field upper → KILLED
		6. equal to less than → KILLED
		7. equal to less or equal → SURVIVED
		8. equal to greater than → KILLED
		9. equal to greater or equal → SURVIVED
		10. equal to not equal → KILLED
		11. Incremented (a++) double field upper → KILLED
		12. Incremented (a++) double field upper → SURVIVED
		13. Decremented (a--) double field upper → KILLED
		14. Decremented (a--) double field upper → SURVIVED
		15. Incremented (++a) double field upper → KILLED
		16. Incremented (++a) double field upper → KILLED
		17. Decremented (--a) double fieldupper → KILLED
		18. Decremented (--a) double fieldupper → KILLED
434		1. replaced boolean return with true for org/jfree/data/Range::equals → NO_COVERAGE
		2. Substituted 0 with 1 → NO_COVERAGE
		3. replaced return of integer sized value with (x == 0 ? 1 : 0) → NO_COVERAGE
		4. Substituted 0 with 1 → NO_COVERAGE
		5. Substituted 0 with -1 → NO_COVERAGE
		6. Substituted 0 with 1 → NO_COVERAGE
		7. Substituted 0 with -1 → NO_COVERAGE

436		1. replaced boolean return with false for org/jfree/data/Range::equals → KILLED 2. Substituted 1 with 0 → KILLED 3. replaced return of integer sized value with (x == 0 ? 1 : 0) → KILLED 4. Substituted 1 with 0 → KILLED 5. Substituted 1 with -1 → SURVIVED 6. Substituted 1 with -1 → SURVIVED 7. Substituted 1 with 2 → KILLED 8. Substituted 1 with 0 → KILLED
448		1. replaced boolean return with false for org/jfree/data/Range::isNaNRange → KILLED 2. replaced boolean return with true for org/jfree/data/Range::isNaNRange → KILLED 3. Substituted 1 with 0 → KILLED 4. Substituted 0 with 1 → KILLED 5. negated conditional → KILLED 6. negated conditional → KILLED 7. removed call to java/lang/Double::isNaN → KILLED 8. removed call to java/lang/Double::isNaN → KILLED 9. removed conditional - replaced equality check with false → KILLED 10. removed conditional - replaced equality check with false → KILLED 11. removed conditional - replaced equality check with true → KILLED 12. removed conditional - replaced equality check with true → KILLED 13. replaced return of integer sized value with (x == 0 ? 1 : 0) → KILLED 14. replaced return of integer sized value with (x == 0 ? 1 : 0) → KILLED 15. Negated double field lower → SURVIVED 16. Negated double field upper → SURVIVED 17. Substituted 0 with 1 → KILLED 18. Substituted 1 with 0 → KILLED 19. Substituted 1 with -1 → SURVIVED 20. Substituted 0 with -1 → KILLED 21. Substituted 1 with -1 → SURVIVED 22. Substituted 1 with 2 → KILLED 23. Substituted 0 with 1 → KILLED 24. Substituted 1 with 0 → KILLED 25. Substituted 0 with -1 → KILLED 26. equal to less than → KILLED 27. equal to less than → KILLED 28. equal to less or equal → SURVIVED 29. equal to less or equal → SURVIVED 30. equal to greater than → KILLED 31. equal to greater than → KILLED 32. equal to greater or equal → KILLED 33. equal to greater or equal → KILLED 34. equal to not equal → KILLED 35. equal to not equal → KILLED 36. Incremented (a++) double field lower → KILLED 37. Incremented (a++) double field upper → SURVIVED 38. Decremented (a--) double field lower → KILLED 39. Decremented (a--) double field upper → SURVIVED 40. Incremented (++a) double field lower → KILLED

		41. Incremented (++a) double field upper → SURVIVED 42. Decremented (--a) double fieldlower → KILLED 43. Decremented (--a) double fieldupper → SURVIVED
460		1. removed call to java/lang/Double::doubleToLongBits → NO_COVERAGE 2. Negated double field lower → NO_COVERAGE 3. Incremented (a++) double field lower → NO_COVERAGE 4. Decremented (a--) double field lower → NO_COVERAGE 5. Incremented (++a) double field lower → NO_COVERAGE 6. Decremented (--a) double fieldlower → NO_COVERAGE
461		1. Substituted 32 with 33 → NO_COVERAGE 2. Replaced Unsigned Shift Right with Shift Left → NO_COVERAGE 3. Replaced XOR with AND → NO_COVERAGE 4. Negated long local variable number 1 → NO_COVERAGE 5. Negated long local variable number 1 → NO_COVERAGE 6. Substituted 32 with 1 → NO_COVERAGE 7. Substituted 32 with 0 → NO_COVERAGE 8. Substituted 32 with -1 → NO_COVERAGE 9. Substituted 32 with -32 → NO_COVERAGE 10. Substituted 32 with 33 → NO_COVERAGE 11. Substituted 32 with 31 → NO_COVERAGE 12. Incremented (a++) long local variable number 1 → NO_COVERAGE 13. Incremented (a++) long local variable number 1 → NO_COVERAGE 14. Decremented (a--) long local variable number 1 → NO_COVERAGE 15. Decremented (a--) long local variable number 1 → NO_COVERAGE 16. Incremented (++a) long local variable number 1 → NO_COVERAGE 17. Incremented (++a) long local variable number 1 → NO_COVERAGE 18. Decremented (--a) long local variable number 1 → NO_COVERAGE 19. Decremented (--a) long local variable number 1 → NO_COVERAGE
462		1. removed call to java/lang/Double::doubleToLongBits → NO_COVERAGE 2. Negated double field upper → NO_COVERAGE 3. Incremented (a++) double field upper → NO_COVERAGE 4. Decremented (a--) double field upper → NO_COVERAGE 5. Incremented (++a) double field upper → NO_COVERAGE 6. Decremented (--a) double fieldupper → NO_COVERAGE
463		1. Substituted 29 with 30 → NO_COVERAGE 2. Substituted 32 with 33 → NO_COVERAGE 3. Replaced integer multiplication with division → NO_COVERAGE 4. Replaced Unsigned Shift Right with Shift Left → NO_COVERAGE 5. Replaced XOR with AND → NO_COVERAGE 6. Replaced integer addition with subtraction → NO_COVERAGE 7. Negated integer local variable number 3 → NO_COVERAGE 8. Negated long local variable number 1 → NO_COVERAGE 9. Negated long local variable number 1 → NO_COVERAGE 10. Replaced integer operation with first member →

		NO_COVERAGE 11. Replaced integer operation with first member → NO_COVERAGE 12. Replaced integer operation by second member → NO_COVERAGE 13. Replaced integer operation by second member → NO_COVERAGE 14. Replaced integer multiplication with division → NO_COVERAGE 15. Replaced integer addition with subtraction → NO_COVERAGE 16. Replaced integer multiplication with modulus → NO_COVERAGE 17. Replaced integer addition with multiplication → NO_COVERAGE 18. Replaced integer multiplication with addition → NO_COVERAGE 19. Replaced integer addition with division → NO_COVERAGE 20. Replaced integer multiplication with subtraction → NO_COVERAGE 21. Replaced integer addition with modulus → NO_COVERAGE 22. Substituted 29 with 1 → NO_COVERAGE 23. Substituted 32 with 1 → NO_COVERAGE 24. Substituted 29 with 0 → NO_COVERAGE 25. Substituted 32 with 0 → NO_COVERAGE 26. Substituted 29 with -1 → NO_COVERAGE 27. Substituted 32 with -1 → NO_COVERAGE 28. Substituted 29 with -29 → NO_COVERAGE 29. Substituted 32 with -32 → NO_COVERAGE 30. Substituted 29 with 30 → NO_COVERAGE 31. Substituted 32 with 33 → NO_COVERAGE 32. Substituted 29 with 28 → NO_COVERAGE 33. Substituted 32 with 31 → NO_COVERAGE 34. Incremented (a++) integer local variable number 3 → NO_COVERAGE 35. Incremented (a++) long local variable number 1 → NO_COVERAGE 36. Incremented (a++) long local variable number 1 → NO_COVERAGE 37. Decremented (a--) integer local variable number 3 → NO_COVERAGE 38. Decremented (a--) long local variable number 1 → NO_COVERAGE 39. Decremented (a--) long local variable number 1 → NO_COVERAGE 40. Incremented (++a) integer local variable number 3 → NO_COVERAGE 41. Incremented (++a) long local variable number 1 → NO_COVERAGE 42. Incremented (++a) long local variable number 1 → NO_COVERAGE 43. Decremented (--a) integer local variable number 3 → NO_COVERAGE 44. Decremented (--a) long local variable number 1 → NO_COVERAGE 45. Decremented (--a) long local variable number 1 → NO_COVERAGE
464		1. replaced int return with 0 for

		org/jfree/data/Range::hashCode → NO_COVERAGE 2. replaced return of integer sized value with (x == 0 ? 1 : 0) → NO_COVERAGE 3. Negated integer local variable number 3 → NO_COVERAGE 4. Incremented (a++) integer local variable number 3 → NO_COVERAGE 5. Decremented (a--) integer local variable number 3 → NO_COVERAGE 6. Incremented (++a) integer local variable number 3 → NO_COVERAGE 7. Decremented (--a) integer local variable number 3 → NO_COVERAGE
475		1. removed call to java/lang/StringBuilder::<init> → NO_COVERAGE 2. replaced return value with "" for org/jfree/data/Range::toString → NO_COVERAGE 3. removed call to java/lang/StringBuilder::append → NO_COVERAGE 4. removed call to java/lang/StringBuilder::append → NO_COVERAGE 5. removed call to java/lang/StringBuilder::append → NO_COVERAGE 6. removed call to java/lang/StringBuilder::append → NO_COVERAGE 7. removed call to java/lang/StringBuilder::toString → NO_COVERAGE 8. mutated return of Object value for org/jfree/data/Range::toString to (if (x != null) null else throw new RuntimeException) → NO_COVERAGE 9. replaced call to java/lang/StringBuilder::append with receiver → NO_COVERAGE 10. replaced call to java/lang/StringBuilder::append with receiver → NO_COVERAGE 11. replaced call to java/lang/StringBuilder::append with receiver → NO_COVERAGE 12. replaced call to java/lang/StringBuilder::append with receiver → NO_COVERAGE 13. Negated double field lower → NO_COVERAGE 14. Negated double field upper → NO_COVERAGE 15. Incremented (a++) double field lower → NO_COVERAGE 16. Incremented (a++) double field upper → NO_COVERAGE 17. Decremented (a--) double field lower → NO_COVERAGE 18. Decremented (a--) double field upper → NO_COVERAGE 19. Incremented (++a) double field lower → NO_COVERAGE 20. Incremented (++a) double field upper → NO_COVERAGE 21. Decremented (--a) double fieldlower → NO_COVERAGE 22. Decremented (--a) double fieldupper → NO_COVERAGE

Active mutators

- ABS_MUTATOR
- AOD_1_MUTATOR
- AOD_2_MUTATOR
- AOR_1_MUTATOR
- AOR_2_MUTATOR
- AOR_3_MUTATOR
- AOR_4_MUTATOR
- ARGUMENT_PROPAGATION_MUTATOR
- BOOLEAN_FALSE_RETURN

- BOOLEAN_TRUE_RETURN
- CONDITIONALS_BOUNDARY_MUTATOR
- CONSTRUCTOR_CALL_MUTATOR
- CRCR_1_MUTATOR
- CRCR_2_MUTATOR
- CRCR_3_MUTATOR
- CRCR_4_MUTATOR
- CRCR_5_MUTATOR
- CRCR_6_MUTATOR
- EMPTY_RETURN_VALUES
- EXPERIMENTAL_BIGINTEGER_MUTATOR
- EXPERIMENTAL_MEMBER_VARIABLE_MUTATOR
- EXPERIMENTAL_REMOVE_SWITCH_MUTATOR_[0-99]
- EXPERIMENTAL_SWITCH_MUTATOR
- INCREMENTS_MUTATOR
- INLINE_CONSTANT_MUTATOR
- INVERT_NEGS_MUTATOR
- MATH_MUTATOR
- NAKED_RECEIVER
- NEGATE_CONDITIONALS_MUTATOR
- NON_VOID_METHOD_CALL_MUTATOR
- NULL_RETURN_VALUES
- OBBN_1_MUTATOR
- OBBN_2_MUTATOR
- OBBN_3_MUTATOR
- PRIMITIVE_RETURN_VALS_MUTATOR
- REMOVE_CONDITIONALS_EQUAL_ELSE_MUTATOR
- REMOVE_CONDITIONALS_EQUAL_IF_MUTATOR
- REMOVE_CONDITIONALS_ORDER_ELSE_MUTATOR
- REMOVE_CONDITIONALS_ORDER_IF_MUTATOR
- REMOVE_INCREMENTS_MUTATOR
- RETURN_VALS_MUTATOR
- ROR_1_MUTATOR
- ROR_2_MUTATOR
- ROR_3_MUTATOR
- ROR_4_MUTATOR
- ROR_5_MUTATOR
- UOI_1_MUTATOR
- UOI_2_MUTATOR
- UOI_3_MUTATOR
- UOI_4_MUTATOR
- VOID_METHOD_CALL_MUTATOR

Tests examined

- org.jfree.data.RangeTest (49 ms)

Report generated by PIT 1.4.11

Pit Test Coverage Report
Project Summary

Number of Classes	Line Coverage	Mutation Coverage
490	0% 79/57191	0% 466/358334

Breakdown by Package

Name	Number of Classes	Line Coverage	Mutation Coverage
org.jfree.chart	21	0% 0/4177	0% 0/18941
org.jfree.chart.annotations	16	0% 0/1477	0% 0/9780
org.jfree.chart.axis	46	0% 0/6879	0% 0/45896
org.jfree.chart.block	16	0% 0/1356	0% 0/8570
org.jfree.chart.demo	3	0% 0/185	0% 0/1546
org.jfree.chart.editor	13	0% 0/1067	0% 0/3695
org.jfree.chart.encoders	5	0% 0/117	0% 0/302
org.jfree.chart.entity	13	0% 0/490	0% 0/2318
org.jfree.chart.event	9	0% 0/83	0% 0/174
org.jfree.chart.imagemap	5	0% 0/105	0% 0/488
org.jfree.chart.labels	26	0% 0/1098	0% 0/4957
org.jfree.chart.needle	10	0% 0/400	0% 0/2794
org.jfree.chart.panel	2	0% 0/301	0% 0/1346
org.jfree.chart.plot	46	0% 0/11548	0% 0/67655
org.jfree.chart.plot.dial	12	0% 0/1415	0% 0/8064
org.jfree.chart.renderer	11	0% 0/1755	0% 0/10288
org.jfree.chart.renderer.category	24	0% 0/5180	0% 0/36433

org.jfree.chart.renderer.xy	32	0% 0/6045	0% 0/45263	
org.jfree.chart.resources	1	0% 0/9	0% 0/2	
org.jfree.chart.servlet	3	0% 0/152	0% 0/582	
org.jfree.chart.title	10	0% 0/1311	0% 0/5283	
org.jfree.chart.urls	9	0% 0/333	0% 0/1697	
org.jfree.chart.util	15	0% 2/904	0% 3/7781	
org.jfree.data	15	7% 77/1116	7% 463/6839	
org.jfree.data.category	4	0% 0/431	0% 0/2502	
org.jfree.data.contour	2	0% 0/190	0% 0/1834	
org.jfree.data.function	4	0% 0/77	0% 0/733	
org.jfree.data.gantt	5	0% 0/458	0% 0/2662	
org.jfree.data.general	14	0% 0/1515	0% 0/12016	
org.jfree.data.io	1	0% 0/59	0% 0/394	
org.jfree.data.jdbc	3	0% 0/265	0% 0/1189	
org.jfree.data.resources	6	0% 0/36	0% 0/12	
org.jfree.data.statistics	14	0% 0/1480	0% 0/12371	
org.jfree.data.time	24	0% 0/2665	0% 0/18663	
org.jfree.data.time.ohlc	4	0% 0/158	0% 0/885	
org.jfree.data.xml	8	0% 0/208	0% 0/418	
org.jfree.data.xy	38	0% 0/2146	0% 0/13961	

Report generated by PIT 1.4.11

March 14, 2022   8:39 AM

Pit Test Coverage Report
Package Summary
org.jfree.data

Number of Classes	Line Coverage	Mutation Coverage
15	7% 77/1116	7% 463/6839

Breakdown by Class

Name	Line Coverage	Mutation Coverage
ComparableObjectItem.java	0% 0/28	0% 0/146
ComparableObjectSeries.java	0% 0/117	0% 0/886
DataUtilities.java	63% 50/80	58% 400/687
DefaultKeyedValue.java	0% 0/25	0% 0/153
DefaultKeyedValues.java	22% 27/123	10% 63/661
DefaultKeyedValues2D.java	0% 0/152	0% 0/862
DomainOrder.java	0% 0/25	0% 0/91
KeyToGroupMap.java	0% 0/94	0% 0/322
KeyedObject.java	0% 0/23	0% 0/87
KeyedObjects.java	0% 0/95	0% 0/437
KeyedObjects2D.java	0% 0/157	0% 0/871
KeyedValueComparator.java	0% 0/52	0% 0/230
KeyedValueComparatorType.java	0% 0/17	0% 0/56
Range.java	0% 0/103	0% 0/1259
RangeType.java	0% 0/25	0% 0/91

Report generated by PIT 1.4.11

DataUtilities.java

1		/* ===
2		* JFreeChart : a free chart library for the Java(tm) platform
3		* ===
4		*
5		* (C) Copyright 2000-2013, by Object Refinery Limited and Contributors.
6		*
7		* Project Info: http://www.jfree.org/jfreechart/index.html
8		*
9		* This library is free software; you can redistribute it and/or modify it
10		* under the terms of the GNU Lesser General Public License as published by
11		* the Free Software Foundation; either version 2.1 of the License, or
12		* (at your option) any later version.
13		*
14		* This library is distributed in the hope that it will be useful, but
15		* WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY
16		* or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public
17		* License for more details.
18		*
19		* You should have received a copy of the GNU Lesser General Public
20		* License along with this library; if not, write to the Free Software
21		* Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301,
22		* USA.
23		*
24		* [Oracle and Java are registered trademarks of Oracle and/or its affiliates.
25		* Other names may be trademarks of their respective owners.]
26		*
27		* -----------------
28		* DataUtilities.java
29		* -----------------
30		* (C) Copyright 2003-2013, by Object Refinery Limited and contributors.
31		*
32		* Original Author: David Gilbert (for Object Refinery Limited);
33		* Contributor(s): Peter Kolb (patch 2511330);
34		*
35		* Changes

36		* -------
37		* 05-Mar-2003 : Version 1 (DG);
38		* 03-Mar-2005 : Moved createNumberArray() and createNumberArray2D() methods
39		* from the DatasetUtilities class (DG);
40		* 17-May-2005 : Added calculateColumnTotal() and calculateRowTotal()
41		* methods (DG);
42		* 28-Jan-2009 : Added equal(double[][], double[][]) method (DG);
43		* 28-Jan-2009 : Added clone(double[][]) method (DG);
44		* 04-Feb-2009 : Added calculateColumnTotal/RowTotal variants (PK);
45		* 03-Jul-2013 : Use ParamChecks (DG);
46		*
47		*/
48		
49		package org.jfree.data;
50		
51		import java.util.Arrays;
52		import org.jfree.chart.util.ParamChecks;
53		import org.jfree.data.general.DatasetUtilities;
54		
55		/**
56		* Utility methods for use with some of the data classes (but not the datasets,
57		* see {@link DatasetUtilities}).
58		*/
59		public abstract class DataUtilities {
60		
61		/**
62		* Tests two arrays for equality. To be considered equal, the arrays must
63		* have exactly the same dimensions, and the values in each array must also
64		* match (two values that qre both NaN or both INF are considered equal
65		* in this test).
66		*
67		* @param a the first array (<code>null</code> permitted).
68		* @param b the second array (<code>null</code> permitted).
69		*
70		* @return A boolean.
71		*
72		* @since 1.0.13
73		* */
74		public static boolean equal(double[][] a, double[][] b) {
75	4	`        if (a == null) {`
76	19	`            return (b == null);`

77		}
78	4	`        if (b == null) {`
79	7	`            return false;  // already know 'a' isn't null`
80		`        }`
81	8	`        if (a.length != b.length) {`
82	7	`            return false;`
83		`        }`
84	19	`        for (int i = 0; i < a.length; i++) {`
85	19	`            if (!Arrays.equals(a[i], b[i])) {`
86	7	`                return false;`
87		`            }`
88		`        }`
89	8	`        return true;`
90		`    }`
91		
92		/**
93		* Returns a clone of the specified array.
94		*
95		* @param source the source array (<code>null</code> not permitted).
96		*
97		* @return A clone of the array.
98		*
99		* @since 1.0.13
100		*/
101		public static double[][] clone(double[][] source) {
102	1	`        ParamChecks.nullNotPermitted(source, "source");`
103		double[][] clone = new double[source.length][];
104	19	`        for (int i = 0; i < source.length; i++) {`
105	9	`            if (source[i] != null) {`
106	5	`                double[] row = new double[source[i].length];`
107	21	`                System.arraycopy(source[i], 0, row, 0, source[i].length);`
108	5	`                clone[i] = row;`
109		`            }`
110		`        }`
111	2	`        return clone;`
112		`    }`
113		
114		/**
115		* Returns the total of the values in one column of the supplied data
116		* table.
117		*

118		* @param data the table of values (<code>null</code> not permitted).
119		* @param column the column index (zero-based).
120		*
121		* @return The total of the values in the specified column.
122		*/
123		public static double calculateColumnTotal(Values2D data, int column) {
124	1	`ParamChecks.nullNotPermitted(data, "data");`
125	5	`double total = 0.0;`
126	1	`int rowCount = data.getRowCount();`
127	25	`for (int r = 0; r < rowCount; r++) {`
128	11	`Number n = data.getValue(r, column);`
129	4	`if (n != null) {`
130	13	`total += n.doubleValue();`
131		`}`
132		`}`
133	7	`return total;`
134		`}`
135		
136		/**
137		* Returns the total of the values in one column of the supplied data
138		* table by taking only the row numbers in the array into account.
139		*
140		* @param data the table of values (<code>null</code> not permitted).
141		* @param column the column index (zero-based).
142		* @param validRows the array with valid rows (zero-based).
143		*
144		* @return The total of the valid values in the specified column.
145		*
146		* @since 1.0.13
147		*/
148		public static double calculateColumnTotal(Values2D data, int column,
149		int[] validRows) {
150	1	`ParamChecks.nullNotPermitted(data, "data");`
151	5	`double total = 0.0;`
152	1	`int rowCount = data.getRowCount();`
153	20	`for (int v = 0; v < validRows.length; v++) {`
154	10	`int row = validRows[v];`
155	19	`if (row < rowCount) {`
156	11	`Number n = data.getValue(row, column);`
157	4	`if (n != null) {`
158	13	`total += n.doubleValue();`
159		`}`

160		}
161		}
162	7	`        return total;`
163		}
164		
165		/**
166		* Returns the total of the values in one row of the supplied data
167		* table.
168		*
169		* @param data the table of values (<code>null</code> not permitted).
170		* @param row the row index (zero-based).
171		*
172		* @return The total of the values in the specified row.
173		*/
174		public static double calculateRowTotal(Values2D data, int row) {
175	1	`        ParamChecks.nullNotPermitted(data, "data");`
176	5	`        double total = 0.0;`
177	1	`        int columnCount = data.getColumnCount();`
178	25	`        for (int c = 0; c < columnCount; c++) {`
179	11	`            Number n = data.getValue(row, c);`
180	4	`            if (n != null) {`
181	13	`                total += n.doubleValue();`
182		}
183		}
184	7	`        return total;`
185		}
186		
187		/**
188		* Returns the total of the values in one row of the supplied data
189		* table by taking only the column numbers in the array into account.
190		*
191		* @param data the table of values (<code>null</code> not permitted).
192		* @param row the row index (zero-based).
193		* @param validCols the array with valid cols (zero-based).
194		*
195		* @return The total of the valid values in the specified row.
196		*
197		* @since 1.0.13
198		*/
199		public static double calculateRowTotal(Values2D data, int row,
200		int[] validCols) {
201	1	`        ParamChecks.nullNotPermitted(data, "data");`

202	5	`            double total = 0.0;`
203	1	`            int colCount = data.getColumnCount();`
204	20	`            for (int v = 0; v < validCols.length; v++) {`
205	10	`                int col = validCols[v];`
206	19	`                if (col < colCount) {`
207	11	`                    Number n = data.getValue(row, col);`
208	4	`                    if (n != null) {`
209	13	`                        total += n.doubleValue();`
210		`                    }`
211		`                }`
212		`            }`
213	7	`        return total;`
214		`    }`
215		
216		`    /**`
217		`     * Constructs an array of <code>Number</code> objects from an array of`
218		`     * <code>double</code> primitives.`
219		`     *`
220		`     * @param data  the data (<code>null</code> not permitted).`
221		`     *`
222		`     * @return An array of <code>Double</code>.`
223		`     */`
224		`    public static Number[] createNumberArray(double[] data) {`
225	1	`        ParamChecks.nullNotPermitted(data, "data");`
226		`        Number[] result = new Number[data.length];`
227	19	`        for (int i = 0; i < data.length; i++) {`
228	16	`            result[i] = new Double(data[i]);`
229		`        }`
230	2	`        return result;`
231		`    }`
232		
233		`    /**`
234		`     * Constructs an array of arrays of <code>Number</code> objects from a`
235		`     * corresponding structure containing <code>double</code> primitives.`
236		`     *`
237		`     * @param data  the data (<code>null</code> not permitted).`
238		`     *`
239		`     * @return An array of <code>Double</code>.`
240	5	`     */`
241	1	`    public static Number[][] createNumberArray2D(double[][] data) {`
242	1	`        ParamChecks.nullNotPermitted(data, "data");`
243		`        int l1 = data.length;`

244	5	`            Number[][] result = new Number[l1][];`
245	25	`            for (int i = 0; i < l1; i++) {`
246	11	`                result[i] = createNumberArray(data[i]);`
247		`            }`
248	2	`            return result;`
249		`        }`
250		
251		`        /**`
252		`         * Returns a {@link KeyedValues} instance that contains the cumulative`
253		`         * percentage values for the data in another {@link KeyedValues} instance.`
254		`         * <p>`
255		`         * The percentages are values between 0.0 and 1.0 (where 1.0 = 100%).`
256		`         *`
257		`         * @param data  the data (<code>null</code> not permitted).`
258		`         *`
259		`         * @return The cumulative percentages.`
260		`         */`
261		`        public static KeyedValues getCumulativePercentages(KeyedValues data) {`
262	1	`            ParamChecks.nullNotPermitted(data, "data");`
263	1	`            DefaultKeyedValues result = new DefaultKeyedValues();`
264	5	`            double total = 0.0;`
265	20	`            for (int i = 0; i < data.getItemCount(); i++) {`
266	6	`                Number v = data.getValue(i);`
267	4	`                if (v != null) {`
268	13	`                    total = total + v.doubleValue();`
269		`                }`
270		`            }`
271	5	`            double runningTotal = 0.0;`
272	20	`            for (int i = 0; i < data.getItemCount(); i++) {`
273	6	`                Number v = data.getValue(i);`
274	4	`                if (v != null) {`
275	13	`                    runningTotal = runningTotal + v.doubleValue();`
276		`                }`
277	25	`                result.addValue(data.getKey(i), new Double(runningTotal / total));`
278		`            }`
279	2	`            return result;`
280		`        }`
281		
282		`}`
		Mutations

75		1. negated conditional → NO_COVERAGE 2. removed conditional - replaced equality check with false → NO_COVERAGE 3. removed conditional - replaced equality check with true → NO_COVERAGE 4. not equal to equal → NO_COVERAGE
76		1. replaced boolean return with false for org/jfree/data/DataUtilities::equal → NO_COVERAGE 2. replaced boolean return with true for org/jfree/data/DataUtilities::equal → NO_COVERAGE 3. Substituted 1 with 0 → NO_COVERAGE 4. Substituted 0 with 1 → NO_COVERAGE 5. negated conditional → NO_COVERAGE 6. removed conditional - replaced equality check with false → NO_COVERAGE 7. removed conditional - replaced equality check with true → NO_COVERAGE 8. replaced return of integer sized value with (x == 0 ? 1 : 0) → NO_COVERAGE 9. replaced return of integer sized value with (x == 0 ? 1 : 0) → NO_COVERAGE 10. Substituted 0 with 1 → NO_COVERAGE 11. Substituted 1 with 0 → NO_COVERAGE 12. Substituted 1 with -1 → NO_COVERAGE 13. Substituted 0 with -1 → NO_COVERAGE 14. Substituted 1 with -1 → NO_COVERAGE 15. Substituted 1 with 2 → NO_COVERAGE 16. Substituted 0 with 1 → NO_COVERAGE 17. Substituted 1 with 0 → NO_COVERAGE 18. Substituted 0 with -1 → NO_COVERAGE 19. not equal to equal → NO_COVERAGE
78		1. negated conditional → NO_COVERAGE 2. removed conditional - replaced equality check with false → NO_COVERAGE 3. removed conditional - replaced equality check with true → NO_COVERAGE 4. not equal to equal → NO_COVERAGE
79		1. replaced boolean return with true for org/jfree/data/DataUtilities::equal → NO_COVERAGE 2. Substituted 0 with 1 → NO_COVERAGE 3. replaced return of integer sized value with (x == 0 ? 1 : 0) → NO_COVERAGE 4. Substituted 0 with 1 → NO_COVERAGE 5. Substituted 0 with -1 → NO_COVERAGE 6. Substituted 0 with 1 → NO_COVERAGE 7. Substituted 0 with -1 → NO_COVERAGE
81		1. negated conditional → NO_COVERAGE 2. removed conditional - replaced equality check with false → NO_COVERAGE 3. removed conditional - replaced equality check with true → NO_COVERAGE 4. equal to less than → NO_COVERAGE 5. equal to less or equal → NO_COVERAGE 6. equal to greater than → NO_COVERAGE 7. equal to greater or equal → NO_COVERAGE 8. equal to not equal → NO_COVERAGE

82		1. replaced boolean return with true for org/jfree/data/DataUtilities::equal → NO_COVERAGE 2. Substituted 0 with 1 → NO_COVERAGE 3. replaced return of integer sized value with (x == 0 ? 1 : 0) → NO_COVERAGE 4. Substituted 0 with 1 → NO_COVERAGE 5. Substituted 0 with -1 → NO_COVERAGE 6. Substituted 0 with 1 → NO_COVERAGE 7. Substituted 0 with -1 → NO_COVERAGE
84		1. changed conditional boundary → NO_COVERAGE 2. Substituted 0 with 1 → NO_COVERAGE 3. negated conditional → NO_COVERAGE 4. removed conditional - replaced comparison check with false → NO_COVERAGE 5. removed conditional - replaced comparison check with true → NO_COVERAGE 6. Negated integer local variable number 2 → NO_COVERAGE 7. Substituted 0 with 1 → NO_COVERAGE 8. Substituted 0 with -1 → NO_COVERAGE 9. Substituted 0 with 1 → NO_COVERAGE 10. Substituted 0 with -1 → NO_COVERAGE 11. Less than to less or equal → NO_COVERAGE 12. Less than to greater than → NO_COVERAGE 13. Less than to greater or equal → NO_COVERAGE 14. Less than to equal → NO_COVERAGE 15. Less than to not equal → NO_COVERAGE 16. Incremented (a++) integer local variable number 2 → NO_COVERAGE 17. Decremented (a--) integer local variable number 2 → NO_COVERAGE 18. Incremented (++a) integer local variable number 2 → NO_COVERAGE 19. Decremented (--a) integer local variable number 2 → NO_COVERAGE
85		1. negated conditional → NO_COVERAGE 2. removed call to java/util/Arrays::equals → NO_COVERAGE 3. removed conditional - replaced equality check with false → NO_COVERAGE 4. removed conditional - replaced equality check with true → NO_COVERAGE 5. Negated integer local variable number 2 → NO_COVERAGE 6. Negated integer local variable number 2 → NO_COVERAGE 7. not equal to less than → NO_COVERAGE 8. not equal to less or equal → NO_COVERAGE 9. not equal to greater than → NO_COVERAGE 10. not equal to greater or equal → NO_COVERAGE 11. not equal to equal → NO_COVERAGE 12. Incremented (a++) integer local variable number 2 → NO_COVERAGE 13. Incremented (a++) integer local variable number 2 → NO_COVERAGE 14. Decremented (a--) integer local variable number 2 → NO_COVERAGE 15. Decremented (a--) integer local variable number 2 → NO_COVERAGE 16. Incremented (++a) integer local variable number 2 → NO_COVERAGE

		17. Incremented (++a) integer local variable number 2 → NO_COVERAGE 18. Decremented (--a) integer local variable number 2 → NO_COVERAGE 19. Decremented (--a) integer local variable number 2 → NO_COVERAGE
86		1. replaced boolean return with true for org/jfree/data/DataUtilities::equal → NO_COVERAGE 2. Substituted 0 with 1 → NO_COVERAGE 3. replaced return of integer sized value with (x == 0 ? 1 : 0) → NO_COVERAGE 4. Substituted 0 with 1 → NO_COVERAGE 5. Substituted 0 with -1 → NO_COVERAGE 6. Substituted 0 with 1 → NO_COVERAGE 7. Substituted 0 with -1 → NO_COVERAGE
89		1. replaced boolean return with false for org/jfree/data/DataUtilities::equal → NO_COVERAGE 2. Substituted 1 with 0 → NO_COVERAGE 3. replaced return of integer sized value with (x == 0 ? 1 : 0) → NO_COVERAGE 4. Substituted 1 with 0 → NO_COVERAGE 5. Substituted 1 with -1 → NO_COVERAGE 6. Substituted 1 with -1 → NO_COVERAGE 7. Substituted 1 with 2 → NO_COVERAGE 8. Substituted 1 with 0 → NO_COVERAGE
102		1. removed call to org/jfree/chart/util/ParamChecks::nullNotPermitted → NO_COVERAGE
104		1. changed conditional boundary → NO_COVERAGE 2. Substituted 0 with 1 → NO_COVERAGE 3. negated conditional → NO_COVERAGE 4. removed conditional - replaced comparison check with false → NO_COVERAGE 5. removed conditional - replaced comparison check with true → NO_COVERAGE 6. Negated integer local variable number 2 → NO_COVERAGE 7. Substituted 0 with 1 → NO_COVERAGE 8. Substituted 0 with -1 → NO_COVERAGE 9. Substituted 0 with 1 → NO_COVERAGE 10. Substituted 0 with -1 → NO_COVERAGE 11. Less than to less or equal → NO_COVERAGE 12. Less than to greater than → NO_COVERAGE 13. Less than to greater or equal → NO_COVERAGE 14. Less than to equal → NO_COVERAGE 15. Less than to not equal → NO_COVERAGE 16. Incremented (a++) integer local variable number 2 → NO_COVERAGE 17. Decremented (a--) integer local variable number 2 → NO_COVERAGE 18. Incremented (++a) integer local variable number 2 → NO_COVERAGE 19. Decremented (--a) integer local variable number 2 → NO_COVERAGE
105		1. negated conditional → NO_COVERAGE 2. removed conditional - replaced equality check with false → NO_COVERAGE

		3. removed conditional - replaced equality check with true → NO_COVERAGE 4. Negated integer local variable number 2 → NO_COVERAGE 5. equal to not equal → NO_COVERAGE 6. Incremented (a++) integer local variable number 2 → NO_COVERAGE 7. Decremented (a--) integer local variable number 2 → NO_COVERAGE 8. Incremented (++a) integer local variable number 2 → NO_COVERAGE 9. Decremented (--a) integer local variable number 2 → NO_COVERAGE
106		1. Negated integer local variable number 2 → NO_COVERAGE 2. Incremented (a++) integer local variable number 2 → NO_COVERAGE 3. Decremented (a--) integer local variable number 2 → NO_COVERAGE 4. Incremented (++a) integer local variable number 2 → NO_COVERAGE 5. Decremented (--a) integer local variable number 2 → NO_COVERAGE
107		1. Substituted 0 with 1 → NO_COVERAGE 2. Substituted 0 with 1 → NO_COVERAGE 3. removed call to java/lang/System::arraycopy → NO_COVERAGE 4. Negated integer local variable number 2 → NO_COVERAGE 5. Negated integer local variable number 2 → NO_COVERAGE 6. Substituted 0 with 1 → NO_COVERAGE 7. Substituted 0 with 1 → NO_COVERAGE 8. Substituted 0 with -1 → NO_COVERAGE 9. Substituted 0 with -1 → NO_COVERAGE 10. Substituted 0 with 1 → NO_COVERAGE 11. Substituted 0 with 1 → NO_COVERAGE 12. Substituted 0 with -1 → NO_COVERAGE 13. Substituted 0 with -1 → NO_COVERAGE 14. Incremented (a++) integer local variable number 2 → NO_COVERAGE 15. Incremented (a++) integer local variable number 2 → NO_COVERAGE 16. Decremented (a--) integer local variable number 2 → NO_COVERAGE 17. Decremented (a--) integer local variable number 2 → NO_COVERAGE 18. Incremented (++a) integer local variable number 2 → NO_COVERAGE 19. Incremented (++a) integer local variable number 2 → NO_COVERAGE 20. Decremented (--a) integer local variable number 2 → NO_COVERAGE 21. Decremented (--a) integer local variable number 2 → NO_COVERAGE
108		1. Negated integer local variable number 2 → NO_COVERAGE 2. Incremented (a++) integer local variable number 2 → NO_COVERAGE 3. Decremented (a--) integer local variable number 2 → NO_COVERAGE 4. Incremented (++a) integer local variable number 2 →

		NO_COVERAGE 5. Decremented (--a) integer local variable number 2 → NO_COVERAGE
111		1. replaced return value with null for org/jfree/data/DataUtilities::clone → NO_COVERAGE 2. mutated return of Object value for org/jfree/data/DataUtilities::clone to (if (x != null) null else throw new RuntimeException) → NO_COVERAGE
124		1. removed call to org/jfree/chart/util/ParamChecks::nullNotPermitted → SURVIVED
125		1. Substituted 0.0 with 1.0 → KILLED 2. Substituted 0.0 with 1.0 → KILLED 3. Substituted 0.0 with -1.0 → KILLED 4. Substituted 0.0 with 1.0 → KILLED 5. Substituted 0.0 with -1.0 → KILLED
126		1. removed call to org/jfree/data/Values2D::getRowCount → KILLED
127		1. changed conditional boundary → KILLED 2. Changed increment from 1 to -1 → KILLED 3. Substituted 0 with 1 → KILLED 4. negated conditional → KILLED 5. removed conditional - replaced comparison check with false → KILLED 6. removed conditional - replaced comparison check with true → KILLED 7. Negated integer local variable number 5 → KILLED 8. Negated integer local variable number 4 → KILLED 9. Substituted 0 with 1 → KILLED 10. Substituted 0 with -1 → KILLED 11. Substituted 0 with 1 → KILLED 12. Substituted 0 with -1 → KILLED 13. Less than to less or equal → KILLED 14. Less than to greater than → KILLED 15. Less than to greater or equal → KILLED 16. Less than to equal → KILLED 17. Less than to not equal → SURVIVED 18. Incremented (a++) integer local variable number 5 → KILLED 19. Incremented (a++) integer local variable number 4 → KILLED 20. Decremented (a--) integer local variable number 5 → KILLED 21. Decremented (a--) integer local variable number 4 → KILLED 22. Incremented (++a) integer local variable number 5 → KILLED 23. Incremented (++a) integer local variable number 4 → KILLED 24. Decremented (--a) integer local variable number 5 → KILLED 25. Decremented (--a) integer local variable number 4 → KILLED
128		1. removed call to org/jfree/data/Values2D::getValue → KILLED 2. Negated integer local variable number 5 → KILLED

		3. Negated integer local variable number 1 → KILLED 4. Incremented (a++) integer local variable number 5 → KILLED 5. Incremented (a++) integer local variable number 1 → KILLED 6. Decremented (a--) integer local variable number 5 → KILLED 7. Decremented (a--) integer local variable number 1 → KILLED 8. Incremented (++a) integer local variable number 5 → KILLED 9. Incremented (++a) integer local variable number 1 → KILLED 10. Decremented (--a) integer local variable number 5 → KILLED 11. Decremented (--a) integer local variable number 1 → KILLED
129		1. negated conditional → KILLED 2. removed conditional - replaced equality check with false → KILLED 3. removed conditional - replaced equality check with true → KILLED 4. equal to not equal → KILLED
130		1. Replaced double addition with subtraction → KILLED 2. removed call to java/lang/Number::doubleValue → KILLED 3. Negated double local variable number 2 → KILLED 4. Replaced double operation with first member → KILLED 5. Replaced double operation by second member → KILLED 6. Replaced double addition with subtraction → KILLED 7. Replaced double addition with multiplication → KILLED 8. Replaced double addition with division → KILLED 9. Replaced double addition with modulus → KILLED 10. Incremented (a++) double local variable number 2 → SURVIVED 11. Decremented (a--) double local variable number 2 → SURVIVED 12. Incremented (++a) double local variable number 2 → KILLED 13. Decremented (--a) double local variable number 2 → KILLED
133		1. replaced double return with 0.0d for org/jfree/data/DataUtilities::calculateColumnTotal → KILLED 2. replaced return of double value with -(x + 1) for org/jfree/data/DataUtilities::calculateColumnTotal → KILLED 3. Negated double local variable number 2 → KILLED 4. Incremented (a++) double local variable number 2 → SURVIVED 5. Decremented (a--) double local variable number 2 → SURVIVED 6. Incremented (++a) double local variable number 2 → KILLED 7. Decremented (--a) double local variable number 2 → KILLED
150		1. removed call to org/jfree/chart/util/ParamChecks::nullNotPermitted → SURVIVED

151		1. Substituted 0.0 with 1.0 → KILLED 2. Substituted 0.0 with 1.0 → KILLED 3. Substituted 0.0 with -1.0 → KILLED 4. Substituted 0.0 with 1.0 → KILLED 5. Substituted 0.0 with -1.0 → KILLED
152		1. removed call to org/jfree/data/Values2D::getRowCount → KILLED
153		1. changed conditional boundary → KILLED 2. Changed increment from 1 to -1 → KILLED 3. Substituted 0 with 1 → KILLED 4. negated conditional → KILLED 5. removed conditional - replaced comparison check with false → KILLED 6. removed conditional - replaced comparison check with true → KILLED 7. Negated integer local variable number 6 → KILLED 8. Substituted 0 with 1 → KILLED 9. Substituted 0 with -1 → KILLED 10. Substituted 0 with 1 → KILLED 11. Substituted 0 with -1 → KILLED 12. Less than to less or equal → KILLED 13. Less than to greater than → KILLED 14. Less than to greater or equal → KILLED 15. Less than to equal → KILLED 16. Less than to not equal → SURVIVED 17. Incremented (a++) integer local variable number 6 → KILLED 18. Decremented (a--) integer local variable number 6 → KILLED 19. Incremented (++a) integer local variable number 6 → KILLED 20. Decremented (--a) integer local variable number 6 → KILLED
154		1. Negated integer local variable number 6 → KILLED 2. Negated integer array field → KILLED 3. Incremented (a++) integer local variable number 6 → KILLED 4. Incremented (a++) integer array field → SURVIVED 5. Decremented (a--) integer local variable number 6 → TIMED_OUT 6. Decremented (a--) integer array field → SURVIVED 7. Incremented (++a) integer local variable number 6 → KILLED 8. Incremented (++a) integer array field → KILLED 9. Decremented (--a) integer local variable number 6 → KILLED 10. Decremented (--a) integer array field → KILLED
155		1. changed conditional boundary → KILLED 2. negated conditional → KILLED 3. removed conditional - replaced comparison check with false → KILLED 4. removed conditional - replaced comparison check with true → KILLED 5. Negated integer local variable number 7 → KILLED 6. Negated integer local variable number 5 → KILLED 7. greater or equal to less than → KILLED 8. greater or equal to less or equal → KILLED

9. greater or equal to greater than → KILLED
10. greater or equal to equal → SURVIVED
11. greater or equal to not equal → KILLED
12. Incremented (a++) integer local variable number 7 → KILLED
13. Incremented (a++) integer local variable number 5 → KILLED
14. Decremented (a--) integer local variable number 7 → KILLED
15. Decremented (a--) integer local variable number 5 → KILLED
16. Incremented (++a) integer local variable number 7 → KILLED
17. Incremented (++a) integer local variable number 5 → KILLED
18. Decremented (--a) integer local variable number 7 → KILLED
19. Decremented (--a) integer local variable number 5 → KILLED

156		1. removed call to org/jfree/data/Values2D::getValue → KILLED 2. Negated integer local variable number 7 → KILLED 3. Negated integer local variable number 1 → KILLED 4. Incremented (a++) integer local variable number 7 → SURVIVED 5. Incremented (a++) integer local variable number 1 → KILLED 6. Decremented (a--) integer local variable number 7 → SURVIVED 7. Decremented (a--) integer local variable number 1 → KILLED 8. Incremented (++a) integer local variable number 7 → KILLED 9. Incremented (++a) integer local variable number 1 → KILLED 10. Decremented (--a) integer local variable number 7 → KILLED 11. Decremented (--a) integer local variable number 1 → KILLED
157		1. negated conditional → KILLED 2. removed conditional - replaced equality check with false → KILLED 3. removed conditional - replaced equality check with true → KILLED 4. equal to not equal → KILLED
158		1. Replaced double addition with subtraction → KILLED 2. removed call to java/lang/Number::doubleValue → KILLED 3. Negated double local variable number 3 → KILLED 4. Replaced double operation with first member → KILLED 5. Replaced double operation by second member → KILLED 6. Replaced double addition with subtraction → KILLED 7. Replaced double addition with multiplication → KILLED 8. Replaced double addition with division → KILLED 9. Replaced double addition with modulus → KILLED 10. Incremented (a++) double local variable number 3 → SURVIVED 11. Decremented (a--) double local variable number 3 →

		SURVIVED 12. Incremented (++a) double local variable number 3 → KILLED 13. Decremented (--a) double local variable number 3 → KILLED
162		1. replaced double return with 0.0d for org/jfree/data/DataUtilities::calculateColumnTotal → KILLED 2. replaced return of double value with -(x + 1) for org/jfree/data/DataUtilities::calculateColumnTotal → KILLED 3. Negated double local variable number 3 → KILLED 4. Incremented (a++) double local variable number 3 → SURVIVED 5. Decremented (a--) double local variable number 3 → SURVIVED 6. Incremented (++a) double local variable number 3 → KILLED 7. Decremented (--a) double local variable number 3 → KILLED
175		1. removed call to org/jfree/chart/util/ParamChecks::nullNotPermitted → SURVIVED
176		1. Substituted 0.0 with 1.0 → KILLED 2. Substituted 0.0 with 1.0 → KILLED 3. Substituted 0.0 with -1.0 → KILLED 4. Substituted 0.0 with 1.0 → KILLED 5. Substituted 0.0 with -1.0 → KILLED
177		1. removed call to org/jfree/data/Values2D::getColumnCount → KILLED
178		1. changed conditional boundary → KILLED 2. Changed increment from 1 to -1 → KILLED 3. Substituted 0 with 1 → KILLED 4. negated conditional → KILLED 5. removed conditional - replaced comparison check with false → KILLED 6. removed conditional - replaced comparison check with true → KILLED 7. Negated integer local variable number 5 → KILLED 8. Negated integer local variable number 4 → KILLED 9. Substituted 0 with 1 → KILLED 10. Substituted 0 with -1 → KILLED 11. Substituted 0 with 1 → KILLED 12. Substituted 0 with -1 → KILLED 13. Less than to less or equal → KILLED 14. Less than to greater than → KILLED 15. Less than to greater or equal → KILLED 16. Less than to equal → KILLED 17. Less than to not equal → SURVIVED 18. Incremented (a++) integer local variable number 5 → KILLED 19. Incremented (a++) integer local variable number 4 → KILLED 20. Decremented (a--) integer local variable number 5 → KILLED 21. Decremented (a--) integer local variable number 4 → KILLED 22. Incremented (++a) integer local variable number 5 →

		KILLED 23. Incremented (++a) integer local variable number 4 → KILLED 24. Decremented (--a) integer local variable number 5 → KILLED 25. Decremented (--a) integer local variable number 4 → KILLED
179		1. removed call to org/jfree/data/Values2D::getValue → KILLED 2. Negated integer local variable number 1 → KILLED 3. Negated integer local variable number 5 → KILLED 4. Incremented (a++) integer local variable number 1 → KILLED 5. Incremented (a++) integer local variable number 5 → KILLED 6. Decremented (a--) integer local variable number 1 → KILLED 7. Decremented (a--) integer local variable number 5 → KILLED 8. Incremented (++a) integer local variable number 1 → KILLED 9. Incremented (++a) integer local variable number 5 → KILLED 10. Decremented (--a) integer local variable number 1 → KILLED 11. Decremented (--a) integer local variable number 5 → KILLED
180		1. negated conditional → KILLED 2. removed conditional - replaced equality check with false → KILLED 3. removed conditional - replaced equality check with true → KILLED 4. equal to not equal → KILLED
181		1. Replaced double addition with subtraction → KILLED 2. removed call to java/lang/Number::doubleValue → KILLED 3. Negated double local variable number 2 → KILLED 4. Replaced double operation with first member → KILLED 5. Replaced double operation by second member → KILLED 6. Replaced double addition with subtraction → KILLED 7. Replaced double addition with multiplication → KILLED 8. Replaced double addition with division → KILLED 9. Replaced double addition with modulus → KILLED 10. Incremented (a++) double local variable number 2 → SURVIVED 11. Decremented (a--) double local variable number 2 → SURVIVED 12. Incremented (++a) double local variable number 2 → KILLED 13. Decremented (--a) double local variable number 2 → KILLED
184		1. replaced double return with 0.0d for org/jfree/data/DataUtilities::calculateRowTotal → KILLED 2. replaced return of double value with -(x + 1) for org/jfree/data/DataUtilities::calculateRowTotal → KILLED 3. Negated double local variable number 2 → KILLED 4. Incremented (a++) double local variable number 2 → SURVIVED

		5. Decremented (a--) double local variable number 2 → SURVIVED 6. Incremented (++a) double local variable number 2 → KILLED 7. Decremented (--a) double local variable number 2 → KILLED
201		1. removed call to org/jfree/chart/util/ParamChecks::nullNotPermitted → SURVIVED
202		1. Substituted 0.0 with 1.0 → KILLED 2. Substituted 0.0 with 1.0 → KILLED 3. Substituted 0.0 with -1.0 → KILLED 4. Substituted 0.0 with 1.0 → KILLED 5. Substituted 0.0 with -1.0 → KILLED
203		1. removed call to org/jfree/data/Values2D::getColumnCount → KILLED
204		1. changed conditional boundary → KILLED 2. Changed increment from 1 to -1 → KILLED 3. Substituted 0 with 1 → KILLED 4. negated conditional → KILLED 5. removed conditional - replaced comparison check with false → KILLED 6. removed conditional - replaced comparison check with true → KILLED 7. Negated integer local variable number 6 → KILLED 8. Substituted 0 with 1 → KILLED 9. Substituted 0 with -1 → KILLED 10. Substituted 0 with 1 → KILLED 11. Substituted 0 with -1 → KILLED 12. Less than to less or equal → KILLED 13. Less than to greater than → KILLED 14. Less than to greater or equal → KILLED 15. Less than to equal → KILLED 16. Less than to not equal → SURVIVED 17. Incremented (a++) integer local variable number 6 → KILLED 18. Decremented (a--) integer local variable number 6 → KILLED 19. Incremented (++a) integer local variable number 6 → KILLED 20. Decremented (--a) integer local variable number 6 → KILLED
205		1. Negated integer local variable number 6 → KILLED 2. Negated integer array field → KILLED 3. Incremented (a++) integer local variable number 6 → KILLED 4. Incremented (a++) integer array field → SURVIVED 5. Decremented (a--) integer local variable number 6 → TIMED_OUT 6. Decremented (a--) integer array field → SURVIVED 7. Incremented (++a) integer local variable number 6 → KILLED 8. Incremented (++a) integer array field → KILLED 9. Decremented (--a) integer local variable number 6 → KILLED 10. Decremented (--a) integer array field → KILLED

206		1. changed conditional boundary → KILLED 2. negated conditional → KILLED 3. removed conditional - replaced comparison check with false → KILLED 4. removed conditional - replaced comparison check with true → KILLED 5. Negated integer local variable number 7 → KILLED 6. Negated integer local variable number 5 → KILLED 7. greater or equal to less than → KILLED 8. greater or equal to less or equal → KILLED 9. greater or equal to greater than → KILLED 10. greater or equal to equal → SURVIVED 11. greater or equal to not equal → KILLED 12. Incremented (a++) integer local variable number 7 → KILLED 13. Incremented (a++) integer local variable number 5 → KILLED 14. Decremented (a--) integer local variable number 7 → KILLED 15. Decremented (a--) integer local variable number 5 → KILLED 16. Incremented (++a) integer local variable number 7 → KILLED 17. Incremented (++a) integer local variable number 5 → KILLED 18. Decremented (--a) integer local variable number 7 → KILLED 19. Decremented (--a) integer local variable number 5 → KILLED
207		1. removed call to org/jfree/data/Values2D::getValue → KILLED 2. Negated integer local variable number 1 → KILLED 3. Negated integer local variable number 7 → KILLED 4. Incremented (a++) integer local variable number 1 → KILLED 5. Incremented (a++) integer local variable number 7 → SURVIVED 6. Decremented (a--) integer local variable number 1 → KILLED 7. Decremented (a--) integer local variable number 7 → SURVIVED 8. Incremented (++a) integer local variable number 1 → KILLED 9. Incremented (++a) integer local variable number 7 → KILLED 10. Decremented (--a) integer local variable number 1 → KILLED 11. Decremented (--a) integer local variable number 7 → KILLED
208		1. negated conditional → KILLED 2. removed conditional - replaced equality check with false → KILLED 3. removed conditional - replaced equality check with true → KILLED 4. equal to not equal → KILLED
209		1. Replaced double addition with subtraction → KILLED 2. removed call to java/lang/Number::doubleValue → KILLED

		3. Negated double local variable number 3 → KILLED 4. Replaced double operation with first member → KILLED 5. Replaced double operation by second member → KILLED 6. Replaced double addition with subtraction → KILLED 7. Replaced double addition with multiplication → KILLED 8. Replaced double addition with division → KILLED 9. Replaced double addition with modulus → KILLED 10. Incremented (a++) double local variable number 3 → SURVIVED 11. Decremented (a--) double local variable number 3 → SURVIVED 12. Incremented (++a) double local variable number 3 → KILLED 13. Decremented (--a) double local variable number 3 → KILLED
213		1. replaced double return with 0.0d for org/jfree/data/DataUtilities::calculateRowTotal → KILLED 2. replaced return of double value with -(x + 1) for org/jfree/data/DataUtilities::calculateRowTotal → KILLED 3. Negated double local variable number 3 → KILLED 4. Incremented (a++) double local variable number 3 → SURVIVED 5. Decremented (a--) double local variable number 3 → SURVIVED 6. Incremented (++a) double local variable number 3 → KILLED 7. Decremented (--a) double local variable number 3 → KILLED
225		1. removed call to org/jfree/chart/util/ParamChecks::nullNotPermitted → NO_COVERAGE
227		1. changed conditional boundary → NO_COVERAGE 2. Substituted 0 with 1 → NO_COVERAGE 3. negated conditional → NO_COVERAGE 4. removed conditional - replaced comparison check with false → NO_COVERAGE 5. removed conditional - replaced comparison check with true → NO_COVERAGE 6. Negated integer local variable number 2 → NO_COVERAGE 7. Substituted 0 with 1 → NO_COVERAGE 8. Substituted 0 with -1 → NO_COVERAGE 9. Substituted 0 with 1 → NO_COVERAGE 10. Substituted 0 with -1 → NO_COVERAGE 11. Less than to less or equal → NO_COVERAGE 12. Less than to greater than → NO_COVERAGE 13. Less than to greater or equal → NO_COVERAGE 14. Less than to equal → NO_COVERAGE 15. Less than to not equal → NO_COVERAGE 16. Incremented (a++) integer local variable number 2 → NO_COVERAGE 17. Decremented (a--) integer local variable number 2 → NO_COVERAGE 18. Incremented (++a) integer local variable number 2 → NO_COVERAGE 19. Decremented (--a) integer local variable number 2 → NO_COVERAGE
228		1. removed call to java/lang/Double::<init> → NO_COVERAGE

		2. Negated integer local variable number 2 → NO_COVERAGE 3. Negated integer local variable number 2 → NO_COVERAGE 4. Negated double array field → NO_COVERAGE 5. Incremented (a++) integer local variable number 2 → NO_COVERAGE 6. Incremented (a++) integer local variable number 2 → NO_COVERAGE 7. Incremented (a++) double array field → NO_COVERAGE 8. Decremented (a--) integer local variable number 2 → NO_COVERAGE 9. Decremented (a--) integer local variable number 2 → NO_COVERAGE 10. Decremented (a--) double array field → NO_COVERAGE 11. Incremented (++a) integer local variable number 2 → NO_COVERAGE 12. Incremented (++a) integer local variable number 2 → NO_COVERAGE 13. Incremented (++a) double array field → NO_COVERAGE 14. Decremented (--a) integer local variable number 2 → NO_COVERAGE 15. Decremented (--a) integer local variable number 2 → NO_COVERAGE 16. Decremented (--a) double array field → NO_COVERAGE
230		1. replaced return value with null for org/jfree/data/DataUtilities::createNumberArray → NO_COVERAGE 2. mutated return of Object value for org/jfree/data/DataUtilities::createNumberArray to (if (x != null) null else throw new RuntimeException) → NO_COVERAGE
242		1. removed call to org/jfree/chart/util/ParamChecks::nullNotPermitted → NO_COVERAGE
244		1. Negated integer local variable number 1 → NO_COVERAGE 2. Incremented (a++) integer local variable number 1 → NO_COVERAGE 3. Decremented (a--) integer local variable number 1 → NO_COVERAGE 4. Incremented (++a) integer local variable number 1 → NO_COVERAGE 5. Decremented (--a) integer local variable number 1 → NO_COVERAGE
245		1. changed conditional boundary → NO_COVERAGE 2. Changed increment from 1 to -1 → NO_COVERAGE 3. Substituted 0 with 1 → NO_COVERAGE 4. negated conditional → NO_COVERAGE 5. removed conditional - replaced comparison check with false → NO_COVERAGE 6. removed conditional - replaced comparison check with true → NO_COVERAGE 7. Negated integer local variable number 3 → NO_COVERAGE 8. Negated integer local variable number 1 → NO_COVERAGE 9. Substituted 0 with 1 → NO_COVERAGE 10. Substituted 0 with -1 → NO_COVERAGE 11. Substituted 0 with 1 → NO_COVERAGE 12. Substituted 0 with -1 → NO_COVERAGE 13. Less than to less or equal → NO_COVERAGE

		14. Less than to greater than → NO_COVERAGE 15. Less than to greater or equal → NO_COVERAGE 16. Less than to equal → NO_COVERAGE 17. Less than to not equal → NO_COVERAGE 18. Incremented (a++) integer local variable number 3 → NO_COVERAGE 19. Incremented (a++) integer local variable number 1 → NO_COVERAGE 20. Decremented (a--) integer local variable number 3 → NO_COVERAGE 21. Decremented (a--) integer local variable number 1 → NO_COVERAGE 22. Incremented (++a) integer local variable number 3 → NO_COVERAGE 23. Incremented (++a) integer local variable number 1 → NO_COVERAGE 24. Decremented (--a) integer local variable number 3 → NO_COVERAGE 25. Decremented (--a) integer local variable number 1 → NO_COVERAGE
246		1. removed call to org/jfree/data/DataUtilities::createNumberArray → NO_COVERAGE 2. Negated integer local variable number 3 → NO_COVERAGE 3. Negated integer local variable number 3 → NO_COVERAGE 4. Incremented (a++) integer local variable number 3 → NO_COVERAGE 5. Incremented (a++) integer local variable number 3 → NO_COVERAGE 6. Decremented (a--) integer local variable number 3 → NO_COVERAGE 7. Decremented (a--) integer local variable number 3 → NO_COVERAGE 8. Incremented (++a) integer local variable number 3 → NO_COVERAGE 9. Incremented (++a) integer local variable number 3 → NO_COVERAGE 10. Decremented (--a) integer local variable number 3 → NO_COVERAGE 11. Decremented (--a) integer local variable number 3 → NO_COVERAGE
248		1. replaced return value with null for org/jfree/data/DataUtilities::createNumberArray2D → NO_COVERAGE 2. mutated return of Object value for org/jfree/data/DataUtilities::createNumberArray2D to (if (x != null) null else throw new RuntimeException) → NO_COVERAGE
262		1. removed call to org/jfree/chart/util/ParamChecks::nullNotPermitted → SURVIVED
263		1. removed call to org/jfree/data/DefaultKeyedValues::<init> → KILLED
264		1. Substituted 0.0 with 1.0 → KILLED 2. Substituted 0.0 with 1.0 → KILLED 3. Substituted 0.0 with -1.0 → KILLED 4. Substituted 0.0 with 1.0 → KILLED

		5. Substituted 0.0 with -1.0 → KILLED
265		1. changed conditional boundary → KILLED 2. Substituted 0 with 1 → KILLED 3. negated conditional → KILLED 4. removed call to org/jfree/data/KeyedValues::getItemCount → KILLED 5. removed conditional - replaced comparison check with false → KILLED 6. removed conditional - replaced comparison check with true → KILLED 7. Negated integer local variable number 4 → KILLED 8. Substituted 0 with 1 → KILLED 9. Substituted 0 with -1 → KILLED 10. Substituted 0 with 1 → KILLED 11. Substituted 0 with -1 → KILLED 12. Less than to less or equal → KILLED 13. Less than to greater than → KILLED 14. Less than to greater or equal → KILLED 15. Less than to equal → KILLED 16. Less than to not equal → SURVIVED 17. Incremented (a++) integer local variable number 4 → KILLED 18. Decremented (a--) integer local variable number 4 → KILLED 19. Incremented (++a) integer local variable number 4 → KILLED 20. Decremented (--a) integer local variable number 4 → KILLED
266		1. removed call to org/jfree/data/KeyedValues::getValue → KILLED 2. Negated integer local variable number 4 → KILLED 3. Incremented (a++) integer local variable number 4 → KILLED 4. Decremented (a--) integer local variable number 4 → TIMED_OUT 5. Incremented (++a) integer local variable number 4 → KILLED 6. Decremented (--a) integer local variable number 4 → KILLED
267		1. negated conditional → KILLED 2. removed conditional - replaced equality check with false → KILLED 3. removed conditional - replaced equality check with true → KILLED 4. equal to not equal → KILLED
268		1. Replaced double addition with subtraction → KILLED 2. removed call to java/lang/Number::doubleValue → KILLED 3. Negated double local variable number 2 → KILLED 4. Replaced double operation with first member → KILLED 5. Replaced double operation by second member → KILLED 6. Replaced double addition with subtraction → KILLED 7. Replaced double addition with multiplication → KILLED 8. Replaced double addition with division → KILLED 9. Replaced double addition with modulus → KILLED 10. Incremented (a++) double local variable number 2 → SURVIVED 11. Decremented (a--) double local variable number 2 →

		SURVIVED 12. Incremented (++a) double local variable number 2 → KILLED 13. Decremented (--a) double local variable number 2 → KILLED
271		1. Substituted 0.0 with 1.0 → KILLED 2. Substituted 0.0 with 1.0 → KILLED 3. Substituted 0.0 with -1.0 → KILLED 4. Substituted 0.0 with 1.0 → KILLED 5. Substituted 0.0 with -1.0 → KILLED
272		1. changed conditional boundary → KILLED 2. Substituted 0 with 1 → KILLED 3. negated conditional → KILLED 4. removed call to org/jfree/data/KeyedValues::getItemCount → KILLED 5. removed conditional - replaced comparison check with false → KILLED 6. removed conditional - replaced comparison check with true → KILLED 7. Negated integer local variable number 8 → KILLED 8. Substituted 0 with 1 → KILLED 9. Substituted 0 with -1 → KILLED 10. Substituted 0 with 1 → KILLED 11. Substituted 0 with -1 → KILLED 12. Less than to less or equal → KILLED 13. Less than to greater than → KILLED 14. Less than to greater or equal → KILLED 15. Less than to equal → KILLED 16. Less than to not equal → SURVIVED 17. Incremented (a++) integer local variable number 8 → KILLED 18. Decremented (a--) integer local variable number 8 → KILLED 19. Incremented (++a) integer local variable number 8 → KILLED 20. Decremented (--a) integer local variable number 8 → KILLED
273		1. removed call to org/jfree/data/KeyedValues::getValue → KILLED 2. Negated integer local variable number 8 → KILLED 3. Incremented (a++) integer local variable number 8 → KILLED 4. Decremented (a--) integer local variable number 8 → KILLED 5. Incremented (++a) integer local variable number 8 → KILLED 6. Decremented (--a) integer local variable number 8 → KILLED
274		1. negated conditional → KILLED 2. removed conditional - replaced equality check with false → KILLED 3. removed conditional - replaced equality check with true → KILLED 4. equal to not equal → KILLED
275		1. Replaced double addition with subtraction → KILLED 2. removed call to java/lang/Number::doubleValue → KILLED

		3. Negated double local variable number 6 → KILLED 4. Replaced double operation with first member → KILLED 5. Replaced double operation by second member → KILLED 6. Replaced double addition with subtraction → KILLED 7. Replaced double addition with multiplication → KILLED 8. Replaced double addition with division → KILLED 9. Replaced double addition with modulus → KILLED 10. Incremented (a++) double local variable number 6 → SURVIVED 11. Decremented (a--) double local variable number 6 → SURVIVED 12. Incremented (++a) double local variable number 6 → KILLED 13. Decremented (--a) double local variable number 6 → KILLED
277		1. removed call to java/lang/Double::<init> → KILLED 2. Replaced double division with multiplication → KILLED 3. removed call to org/jfree/data/KeyedValues::getKey → KILLED 4. removed call to org/jfree/data/DefaultKeyedValues::addValue → KILLED 5. Negated integer local variable number 8 → KILLED 6. Negated double local variable number 6 → KILLED 7. Negated double local variable number 2 → KILLED 8. Replaced double operation with first member → KILLED 9. Replaced double operation by second member → KILLED 10. Replaced double division with multiplication → KILLED 11. Replaced double division with modulus → KILLED 12. Replaced double division with addition → KILLED 13. Replaced double division with subtraction → KILLED 14. Incremented (a++) integer local variable number 8 → KILLED 15. Incremented (a++) double local variable number 6 → KILLED 16. Incremented (a++) double local variable number 2 → KILLED 17. Decremented (a--) integer local variable number 8 → TIMED_OUT 18. Decremented (a--) double local variable number 6 → KILLED 19. Decremented (a--) double local variable number 2 → KILLED 20. Incremented (++a) integer local variable number 8 → KILLED 21. Incremented (++a) double local variable number 6 → KILLED 22. Incremented (++a) double local variable number 2 → KILLED 23. Decremented (--a) integer local variable number 8 → KILLED 24. Decremented (--a) double local variable number 6 → KILLED 25. Decremented (--a) double local variable number 2 → KILLED
279		1. replaced return value with null for org/jfree/data/DataUtilities::getCumulativePercentages → KILLED 2. mutated return of Object value for

| | | org/jfree/data/DataUtilities::getCumulativePercentages to (if (x != null) null else throw new RuntimeException) → KILLED |

Active mutators

- ABS_MUTATOR
- AOD_1_MUTATOR
- AOD_2_MUTATOR
- AOR_1_MUTATOR
- AOR_2_MUTATOR
- AOR_3_MUTATOR
- AOR_4_MUTATOR
- ARGUMENT_PROPAGATION_MUTATOR
- BOOLEAN_FALSE_RETURN
- BOOLEAN_TRUE_RETURN
- CONDITIONALS_BOUNDARY_MUTATOR
- CONSTRUCTOR_CALL_MUTATOR
- CRCR_1_MUTATOR
- CRCR_2_MUTATOR
- CRCR_3_MUTATOR
- CRCR_4_MUTATOR
- CRCR_5_MUTATOR
- CRCR_6_MUTATOR
- EMPTY_RETURN_VALUES
- EXPERIMENTAL_BIGINTEGER_MUTATOR
- EXPERIMENTAL_MEMBER_VARIABLE_MUTATOR
- EXPERIMENTAL_REMOVE_SWITCH_MUTATOR_[0-99]
- EXPERIMENTAL_SWITCH_MUTATOR
- INCREMENTS_MUTATOR
- INLINE_CONSTANT_MUTATOR
- INVERT_NEGS_MUTATOR
- MATH_MUTATOR
- NAKED_RECEIVER
- NEGATE_CONDITIONALS_MUTATOR
- NON_VOID_METHOD_CALL_MUTATOR
- NULL_RETURN_VALUES
- OBBN_1_MUTATOR
- OBBN_2_MUTATOR
- OBBN_3_MUTATOR
- PRIMITIVE_RETURN_VALS_MUTATOR
- REMOVE_CONDITIONALS_EQUAL_ELSE_MUTATOR
- REMOVE_CONDITIONALS_EQUAL_IF_MUTATOR
- REMOVE_CONDITIONALS_ORDER_ELSE_MUTATOR
- REMOVE_CONDITIONALS_ORDER_IF_MUTATOR
- REMOVE_INCREMENTS_MUTATOR
- RETURN_VALS_MUTATOR
- ROR_1_MUTATOR
- ROR_2_MUTATOR
- ROR_3_MUTATOR
- ROR_4_MUTATOR
- ROR_5_MUTATOR
- UOI_1_MUTATOR
- UOI_2_MUTATOR
- UOI_3_MUTATOR
- UOI_4_MUTATOR
- VOID_METHOD_CALL_MUTATOR

Tests examined

- org.jfree.data.DataUtilitiesTest (551 ms)

Report generated by [PIT](#) 1.4.11