

Non-Inheritance Extract Class Refactoring Suggestions		
Original Class [N _{split}]	Extract Class	
	New Class [Superclass] [Subclass]	Methods/Fields
ChartPanel [3]	ChartPanel_new_1 [ChartProgressListener] [-]	getMinimumDrawWidth() setMinimumDrawWidth(int) minimumDrawWidth getMaximumDrawWidth() setMaximumDrawWidth(int) maximumDrawWidth getMinimumDrawHeight() setMinimumDrawHeight(int) minimumDrawHeight getMaximumDrawHeight() setMaximumDrawHeight(int) maximumDrawHeight setRefreshBuffer(boolean) refreshBuffer chartChanged(ChartChangeEvent) chartProgress(ChartProgressEvent) mouseExited(MouseEvent) mouseEntered(MouseEvent) addChartMouseListener(ChartMouseListener) chartMouseListeners removeChartMouseListener(ChartMouseListener) setBackground(Color) setMouseZoomable(boolean) setMouseZoomable(boolean, boolean) ChartPanel(JFreeChart, boolean, boolean, boolean, boolean, boolean) ChartPanel(JFreeChart, boolean) ChartPanel(JFreeChart) getGraphics() zoomOutVerticalMenuItem zoomInVerticalMenuItem zoomOutBothMenuItem zoomOutHorizontalMenuItem zoomInHorizontalMenuItem autoRangeHorizontalMenuItem autoRangeVerticalMenuItem zoomInBothMenuItem autoRangeBothMenuItem setVerticalAxisTrace(boolean)

		verticalAxisTrace setHorizontalAxisTrace(boolean) horizontalAxisTrace useBuffer available chartArea setPreferredSize(Dimension) enableEvents(long) setDisplayToolTips(boolean) addMouseListener(MouseListener) addMouseMotionListener(MouseMotionListener) createPopupMenu(boolean, boolean, boolean, boolean) displayPopupMenu(int, int) setFillZoomRectangle(boolean) fillZoomRectangle ChartPanel(JFreeChart, int, int, int, int, int, int, boolean, boolean, boolean, boolean, boolean, boolean) zoomRectangle zoomPoint actionPerformed(ActionEvent) mouseReleased(MouseEvent) mouseDragged(MouseEvent) mousePressed(MouseEvent) getPopupMenu() popup setPopupMenu(JPopupMenu)
	ChartPanel_new_2 [-] [-]	repaint() createChartPrintJob() autoRangeBoth() zoomInBoth(double, double) zoomOutBoth(double, double) info getChartRenderingInfo() mouseClicked(MouseEvent) zoomOutVertical(double) zoomOutHorizontal(double) zoomInVertical(double) zoomInHorizontal(double) setHorizontalZoom(boolean) horizontalZoom setVerticalZoom(boolean) verticalZoom

		autoRangeVertical() zoom(Rectangle2D) getChart() setChart(JFreeChart) chart print(Graphics,PageFormat, int) autoRangeHorizontal() attemptEditChartProperties() doSaveAs()
	ChartPanel_new_3 [-] [-]	getHeight() getWidth() setEnforceFileExtensions(boolean) enforceFileExtensions isEnforceFileExtensions() createImage(int, int) chartBuffer chartBufferHeight chartBufferWidth drawVerticalAxisTrace(int) horizontalTraceLine getInsets() paintComponent(Graphics) scaleY mouseMoved(MouseEvent) scaleX getScaledDataArea() getEntityForPoint(int, int) translateScreenToJava2D(Point) translateJava2DToScreen(Point2D) getToolTipText(MouseEvent) verticalTraceLine drawHorizontalAxisTrace(int) getSize()
MeterPlot [3]	MeterPlot_new_1 [Plot] [-]	getDrawBorder() setDrawBorder(boolean) drawBorder getMeterDataset() drawTicks(Graphics2D,Rectangle2D, double, double) drawTick(Graphics2D,Rectangle2D, double) getDialBorderColor() setDialBorderColor(Color) dialBorderColor DEFAULT_CIRCLE_SIZE

		NO_LABELS DIALTYPE_CIRCLE VALUE_LABELS DEFAULT_BORDER_SIZE getDialBackgroundPaint() setDialBackgroundPaint(Paint) dialBackgroundPaint DEFAULT_DIAL_BACKGROUND_PAINT WARNING_TEXT DEFAULT_METER_ANGLE NORMAL_TEXT CRITICAL_TEXT DIALTYPE_CHORD getValuePaint() setValuePaint(Paint) valuePaint DEFAULT_VALUE_PAINT getNeedlePaint() setNeedlePaint(Paint) needlePaint DEFAULT_NEEDLE_PAINT DIALTYPE_PIE getValueFont() setValueFont(Font) valueFont DEFAULT_VALUE_FONT getLegendItemLabels() static {} getInsets() zoom(double) notifyListeners(PlotChangeEvent) getForegroundAlpha() drawOutline(Graphics2D,Rectangle2D) drawBackground(Graphics2D,Rectangle2D) draw(Graphics2D,Rectangle2D,ChartRenderingInfo) getDataset() getLegendItems() MeterPlot(MeterDataset) getPlotType() getMeterAngle() setMeterAngle(int) meterAngle getDialType()
--	--	---

		setDialType(int) dialType drawArc(Graphics2D,Rectangle2D, double, double,Paint, int) calculateAngle(double)
	MeterPlot_new_2 [-] [-]	updateInformation(MeterPlot,MeterDataset, int, int,LegendItem[],Paint[]) createLegendItem(Graphics,LegendItem, double, double) showNormal showWarning showCritical meterCalcAngle meterRange minMeterValue getNormalPaint() setNormalPaint(Paint) normalPaint DEFAULT_NORMAL_PAINT getCriticalPaint() setCriticalPaint(Paint) criticalPaint DEFAULT_CRITICAL_PAINT getWarningPaint() setWarningPaint(Paint) warningPaint DEFAULT_WARNING_PAINT drawArc(Graphics2D,Rectangle2D, double, double,Paint) drawTick(Graphics2D,Rectangle2D, double, boolean,Paint) DEFAULT_BACKGROUND_PAINT drawArcFor(Graphics2D,Rectangle2D,MeterDatas et, int) DEFAULT_LABEL_FONT
	MeterPlot_new_3 [-] [-]	getTickLabelType() setTickLabelType(int) tickLabelType getTickLabelFont() setTickLabelFont(Font) tickLabelFont drawTick(Graphics2D,Rectangle2D, double, boolean,Paint, boolean,String)
ThermometerPlot	ThermometerPlot_new	setValueFormat(NumberFormat)

[3]	_1 [VerticalValuePlot] [-]	valueFormat rangeIndicatorStroke subrangeIndicatorStroke getData() setData(ValueDataset) data getUseSubrangePaint() setUseSubrangePaint(boolean) useSubrangePaint getSubrangePaint(int) setSubrangePaint(int,Paint) subrangePaint getMercuryPaint() setMercuryPaint(Paint) mercuryPaint getCurrentPaint() subrangeIndicatorsVisible ThermometerPlot() getThermometerPaint() setThermometerPaint(Paint) thermometerPaint getFollowDataInSubranges() setFollowDataInSubranges(boolean) followDataInSubranges getShowValueLines() setShowValueLines(boolean) showValueLines getThermometerStroke() setThermometerStroke(Stroke) thermometerStroke getValuePaint() setValuePaint(Paint) valuePaint getValueLocation() setValueLocation(int) valueLocation getValueFont() setValueFont(Font) valueFont getPadding() setPadding(Spacer) padding WARNING BULB_RADIUS
-----	----------------------------------	--

		RANGE_LOW DEFAULT_UPPER_BOUND UNITS_FAHRENHEIT RANGE_HIGH isCompatibleVerticalAxis(Axis) AXIS_GAP DISPLAY_LOW getLegendItemLabels() RIGHT propertyChange() GAP_DIAMETER CRITICAL UNITS_NONE UNITS_KELVIN UNITS getUnits() setUnits(String) units setUnits(int) DEFAULT_LOWER_BOUND isCompatibleHorizontalAxis(Axis) BULB NORMAL DISPLAY_HIGH COLUMN_DIAMETER COLUMN_RADIUS GAP_RADIUS UNITS_CELCIUS BULB_DIAMETER NONE getRangeAxis() setRangeAxis(ValueAxis) rangeAxis draw(Graphics2D,Rectangle2D,ChartRenderingInfo) getLegendItems() static {} readObject(ObjectInputStream) drawBackground(Graphics2D,Rectangle2D) ThermometerPlot(ValueDataset) getVerticalDataRange(ValueAxis) getVerticalValueAxis() getInsets() getPlotType()
--	--	--

		setInsets(Insets) setBackgroundPaint(Paint) datasetChanged(DatasetChangeEvent) setOutlinePaint(Paint) writeObject(ObjectOutputStream) equals(Object) zoom(double) drawOutline(Graphics2D,Rectangle2D) notifyListeners(PlotChangeEvent) isCompatibleRangeAxis(ValueAxis)
	ThermometerPlot_new _2 [-] [-]	setAxisRange() getLowerBound() setLowerBound(double) lowerBound getMinimumVerticalDataValue() getUpperBound() setUpperBound(double) upperBound getMaximumVerticalDataValue() setRange(double, double)
	ThermometerPlot_new _3 [-] [-]	setSubrangeInfo(int, double, double, double, double) setSubrangeInfo(int, double, double) subrangeInfo setSubrange(int, double, double) subrange setDisplayRange(int, double, double) inSubrange(int, double) isValidNumber(double)
ContourPlot [2]	ContourPlot_new_1 [VerticalValuePlot] [-]	getURLGenerator() setURLGenerator(XYURLGenerator) urlGenerator propertyChange(PropertyChangeEvent) domainMarkers clearDomainMarkers() addDomainMarker(Marker) rangeMarkers clearRangeMarkers() addRangeMarker(Marker) isCompatibleColorBarAxis(NumberAxis) setColorBarAxis(NumberAxis) colorBar setDomainCrosshairVisible(boolean) domainCrosshairVisible

		isDomainCrosshairVisible() isCompatibleRangeAxis(ValueAxis) visibleRange(ContourDataset,Range,Range) addAnnotation(Annotation) clearAnnotations() annotations setRangeCrosshairVisible(boolean) rangeCrosshairVisible isRangeCrosshairVisible() DEFAULT_INSETS getHorizontalAxis() setDomainCrosshairValue(double) isCompatibleDomainAxis(ValueAxis) getDomainCrosshairPaint() setDomainCrosshairPaint(Paint) domainCrosshairPaint getRangeCrosshairStroke() setRangeCrosshairStroke(Stroke) rangeCrosshairStroke getRangeCrosshairPaint() setRangeCrosshairPaint(Paint) rangeCrosshairPaint getDomainCrosshairStroke() setDomainCrosshairStroke(Stroke) domainCrosshairStroke getContourDataset() getRangeCrosshairValue() setRangeCrosshairValue(double) rangeCrosshairValue setRangeCrosshairValue(double, boolean) getDomainCrosshairValue() setDomainCrosshairValue(double, boolean) domainCrosshairValue drawHorizontalLine(Graphics2D,Rectangle2D, double,Stroke,Paint) setRenderAsPoints(boolean) renderAsPoints isRenderAsPoints() getDomainAxis() setDomainAxis(ValueAxis) domainAxis getClipPath() setClipPath(ClipPath) clipPath
--	--	---

		render(Graphics2D,Rectangle2D,ChartRenderingInfo,CrosshairInfo) drawVerticalLine(Graphics2D,Rectangle2D,double,Stroke,Paint) getRangeAxis() setRangeAxis(ValueAxis) rangeAxis zoom(double) getInsets() getDataAreaRatio() drawOutline(Graphics2D,Rectangle2D) handleClick(int, int,ChartRenderingInfo) getHorizontalDataRange(ValueAxis) getHorizontalValueAxis() getPlotType() getContourDataRange() draw(Graphics2D,Rectangle2D,ChartRenderingInfo) datasetChanged(DatasetChangeEvent) getForegroundAlpha() axisChanged(AxisChangeEvent) drawBackground(Graphics2D,Rectangle2D) ContourPlot(ContourDataset,ValueAxis,ValueAxis,NumberAxis) drawRangeMarker(Graphics2D) notifyListeners(PlotChangeEvent) getColorBarValueAxis() getVerticalDataRange(ValueAxis) getDataset() drawDomainMarker(Graphics2D) getVerticalValueAxis() static {} getVerticalAxis()
	ContourPlot_new_2 [-] [-]	setDomainCrosshairLockedOnData(boolean) domainCrosshairLockedOnData isDomainCrosshairLockedOnData() setRangeCrosshairLockedOnData(boolean) rangeCrosshairLockedOnData isRangeCrosshairLockedOnData() getPtSizePct() setPtSizePct(double) ptSizePct getMissingPaint() setMissingPaint(Paint)

		missingPaint getToolTipGenerator() setToolTipGenerator(ContourToolTipGenerator) toolTipGenerator contourRenderer(Graphics2D,Rectangle2D,ChartR enderingInfo) pointRenderer(Graphics2D,Rectangle2D,ChartRen deringInfo)
Hour [2]	Hour_new_1 [RegularTimePeriod] [-]	Hour() FIRST_HOUR_IN_DAY LAST_HOUR_IN_DAY parseHour(String) DATE_FORMAT DATE_FORMAT_SHORT parseDay(String) DATE_FORMAT_LONG DATE_FORMAT_MEDIUM getHour() hour previous() getSerialIndex() getFirstMillisecond(Calendar) Hour(int,Day) compareTo(Object) next() Hour(Date) Hour(Date,TimeZone) getLastMillisecond(Calendar)
	Hour_new_2 [-] [-]	equals(Object) getDay() day getYear() getMonth() getDayOfMonth()
TimeSeries [2]	TimeSeries_new_1 [Series] [-]	addOrUpdate(RegularTimePeriod,Number) fireSeriesChanged() createCopy(int, int) TimeSeries(String,String,String,Class) removeChangeListener(SeriesChangeListener) addChangeListener(SeriesChangeListener) firePropertyChange(String,Object,Object) getName() clone()
	TimeSeries_new_2	DEFAULT_RANGE_DESCRIPTION

	<div>[-]</div> <div>[-]</div>	<div>DEFAULT_DOMAIN_DESCRIPTION</div> <div>setDomainDescription(String)</div> <div>getDomainDescription()</div> <div>domain</div> <div>setRangeDescription(String)</div> <div>range</div> <div>getRangeDescription()</div> <div>equals(Object)</div> <div>createCopy(RegularTimePeriod,RegularTimePeriod)</div> <div>getNextTimePeriod()</div> <div>getTimePeriods()</div> <div>TimeSeries(String)</div> <div>addAndOrUpdate()</div> <div>getTimePeriodsUniqueToOtherSeries()</div> <div>getTimePeriod(int)</div> <div>TimeSeries(String,Class)</div> <div>getMaximumItemCount()</div> <div>setMaximumItemCount(int)</div> <div>maximumItemCount</div> <div>getHistoryCount()</div> <div>setHistoryCount(int)</div> <div>historyCount</div> <div>getTimePeriodClass()</div> <div>timePeriodClass</div> <div>add()</div> <div>delete(int, int)</div> <div>update(RegularTimePeriod,Number)</div> <div>getDataPair(RegularTimePeriod)</div> <div>getDataPair(int)</div> <div>getItemCount()</div> <div>data</div> <div>getIndex(RegularTimePeriod)</div> <div>delete(RegularTimePeriod)</div> <div>add(RegularTimePeriod, double)</div> <div>add(RegularTimePeriod,Number)</div> <div>getValue(int)</div> <div>getValue(RegularTimePeriod)</div>
<div>JFreeChart</div> <div>[3]</div>	<div>JFreeChart_new_1</div> <div>[TitleChangeListener]</div> <div>[-]</div>	<div>getBackgroundImage()</div> <div>setBackgroundImage(Image)</div> <div>backgroundImage</div> <div>getBackgroundImageAlignment()</div> <div>setBackgroundImageAlignment(int)</div> <div>backgroundImageAlignment</div>

		getBackgroundImageAlpha() setBackgroundImageAlpha(float) backgroundImageAlpha getTitle() setTitle(TextTitle) title JFreeChart(String,Plot) legendChanged(LegendChangeEvent) JFreeChart(String,Font,Plot, boolean) static {} setTitle(String) titleChanged(TitleChangeEvent) plotChanged(PlotChangeEvent)
	JFreeChart_new_2 [-] [-]	getSuppressChartChangeEvents() setSuppressChartChangeEvents(boolean) notifyListeners(ChartChangeEvent) setNotify(boolean) notify isNotify() removeChangeListener(ChartChangeListener) addChangeListener(ChartChangeListener) changeListeners
	JFreeChart_new_3 [-] [-]	JFreeChart(Plot) notifyListeners(ChartProgressEvent) progressListeners removeProgressListener(ChartProgressListener) addProgressListener(ChartProgressListener) main(String[]) INFO createBufferedImage(int, int) createBufferedImage(int, int,ChartRenderingInfo) draw(Graphics2D,Rectangle2D) fireChartChanged() getAntiAlias() setAntiAlias(boolean) antialias setBackgroundPaint() setBackgroundPaint(Paint) backgroundPaint writeObject(ObjectOutputStream) getLegend() setLegend(Legend) legend draw(Graphics2D,Rectangle2D,ChartRenderingInf

		o) equals(Object) addSubtitle(AbstractTitle) getSubtitles() setSubtitles(List) subtitles getSubtitleCount() getSubtitle(int) readObject(ObjectInputStream) getPlot() plot getCategoryPlot() handleClick(int, int,ChartRenderingInfo) getXYPlot() drawTitle(AbstractTitle,Graphics2D,Rectangle2D)
DynamicTimeSeries Collection [2]	DynamicTimeSeriesCol lection_new_1 [RangeInfo] [-]	setSeriesName(int,String) seriesNames getPosition() setPosition(int) position getX(RegularTimePeriod) workingCalendar minValue maximumItemCount timePeriodClass pointsInTime getMaximumDomainValue() getItemCount(int) getStartXValue(int, int) DynamicTimeSeriesCollection(int, int,RegularTimePeriod,TimeZone) getMaximumRangeValue() getEndYValue(int, int) seriesChanged(SeriesChangeEvent) getYValue(int, int) getSeriesName(int) getStartYValue(int, int) getMinimumRangeValue() getEndXValue(int, int) getXValue(int, int) getMinimumDomainValue() getDomainRange() domainRange getSeriesCount()

		seriesCount getValueRange() valueRange
	DynamicTimeSeriesCollection_new_2 [-] [-]	MIDDLE START END DynamicTimeSeriesCollection(int, int, RegularTimePeriod) maxValue invalidateRangeInfo() fireSeriesChanged() domainIsPointsInTime domainStart domainEnd findDomainLimits() deltaTime getNewestTime() offsetFromNewest(int) DynamicTimeSeriesCollection(int, int) getNewestIndex() newestAt DynamicTimeSeriesCollection(int, int, TimeZone) getY(int, int) valueHistory appendData(float[]) addSeries(float[], int, String) addValue(int, int, float) findMaxValue() historyCount wrapOffset(int) advanceTime() setTimeBase(RegularTimePeriod) offsetFromOldest(int) oldestAt getOldestIndex() getOldestTime() translateGet(int)
DrawableLegendItem m [2]	DrawableLegendItem_new_1 [-] [-]	getMarker() setMarker(Shape) marker getLabelPosition() setLabelPosition(Point2D) labelPosition getItem()

		item DrawableLegendItem(LegendItem)
	DrawableLegendItem_ new_2 [-] [-]	draw(Graphics2D, double, double) getHeight() height getWidth() width getX() setX(double) x getY() setY(double) y setBounds(double, double, double, double)
DefaultIntervalCategoryDataset [2]	DefaultIntervalCategoryDataset_new_1 [IntervalCategoryDataset] [-]	endData setEndValue(int, Object, Number) generateKeys(int, String) startData getCategoryCount() setStartValue(int, Object, Number) setCategoryKeys(Comparable[]) categoryKeys getCategory(int) getItemCount() getCategoryIndex(Object) DefaultIntervalCategoryDataset(String[], Number[][], Number[][]) DefaultIntervalCategoryDataset(double[][], double[][]) DefaultIntervalCategoryDataset(Number[][], Number[][]) setSeriesKeys(Comparable[]) seriesKeys getRowKey(int) DefaultIntervalCategoryDataset(Comparable[], Comparable[], Number[][], Number[][]) getColumnCount() getColumnKeys() getSeriesCount() getSeriesName(int) getEndValue(int, int) getEndValue(Comparable, Comparable) getRowCount() getColumnIndex(Comparable)

		fireDatasetChanged() getRowIndex(Comparable) getStartValue(Comparable,Comparable) getStartValue(int, int) getRowKeys() getValue(int, int) getColumnKey(int)
	DefaultIntervalCategoryDataset_new_2 [-] [-]	getSeriesIndex(Object) getItem(Object) getCategories() getValue(Comparable,Comparable) getSeries() getSeries(int)
HorizontalBarRenderer3D [2]	HorizontalBarRenderer3D_new_1 [HorizontalBarRenderer] [-]	DEFAULT_X_OFFSET DEFAULT_Y_OFFSET HorizontalBarRenderer3D() HorizontalBarRenderer3D(double, double) valuesGap static {} DEFAULT_WALL_PAINT hiddenClip drawRangeMarker(Graphics2D,CategoryPlot,ValueAxis,Marker,Rectangle2D) getRowCount() drawItem(Graphics2D,Rectangle2D,CategoryPlot,CategoryAxis,ValueAxis,KeyedValues2DDataset,int, int, int) HorizontalBarRenderer3D(double, double,CategoryToolTipGenerator,CategoryURLGenerator) getUpperClip() getItemStroke(int, int, int) drawOutline(Graphics2D,CategoryPlot,Rectangle2D) getItemOutlinePaint(int, int, int) getLowerClip() getColumnCount() getItemMargin() getBarWidth() getItemPaint(int, int, int) drawRangeGridline(Graphics2D,CategoryPlot,ValueAxis,Rectangle2D, double) drawDomainGridline(Graphics2D,CategoryPlot,Rectangle2D, double)

		drawBackground(Graphics2D,CategoryPlot,Rectangle2D) getInfo() getURLGenerator() getToolTipGenerator() getXOffset() xOffset getYOffset() yOffset
	HorizontalBarRenderer 3D_new_2 [-] [-]	getWallPaint() setWallPaint(Paint) wallPaint writeObject(ObjectOutputStream) readObject(ObjectInputStream)
VerticalSymbolicAxis [2]	VerticalSymbolicAxis_new_1 [VerticalNumberAxis] [-]	getAnchorValue() setRange(double, double) ySymbolicZoomIsAccepted setAnchoredRange(double) getSymbolicValue() symbolicValue valueToString(double) isGridLinesVisible() setSymbolicGridLinesVisible(boolean) symbolicGridLinesVisible getTickLabelFont() calculateLowestVisibleTickValue() getTickUnit() notifyListeners(AxisChangeEvent) getNumberFormatOverride() setRangeAttribute(Range) refreshTicks(Graphics2D,Rectangle2D,Rectangle2D, int) autoRangeStickyZero() selectAutoTickUnit(Graphics2D,Rectangle2D,Rectangle2D) getTicks() VerticalSymbolicAxis(String,String[]) autoAdjustRange() setAutoRangeStickyZero(boolean) autoRangeIncludesZero() setAutoTickUnitSelection(boolean, boolean) getAutoRangeMinimumSize() static {} translateValueToJava2D(double,Rectangle2D)

		draw(Graphics2D,Rectangle2D,Rectangle2D, int) calculateVisibleTickCount() getTickLabelInsets() getPlot()
	VerticalSymbolicAxis_new_2 [-] [-]	DEFAULT_SYMBOLIC_GRID_LINE_PAINT drawSymbolicGridLines(Graphics2D,Rectangle2D, Rectangle2D) getSymbolicGridPaint() symbolicGridPaint drawSymbolicGridLines(Graphics2D,Rectangle2D, Rectangle2D, boolean) symbolicGridLineList getSymbolicGridLine(int)
HorizontalSymbolicAxis [2]	HorizontalSymbolicAxis_new_1 [HorizontalNumberAxis] [-]	getAnchorValue() setRange(double, double) xSymbolicZoomIsAccepted setAnchoredRange(double) getSymbolicValue() symbolicValue valueToString(double) isGridLinesVisible() setSymbolicGridLinesVisible(boolean) symbolicGridLinesVisible autoAdjustRange() translateValueToJava2D(double,Rectangle2D) getNumberFormatOverride() selectAutoTickUnit(Graphics2D,Rectangle2D,Rectangle2D) setAutoTickUnitSelection(boolean, boolean) getTickLabelInsets() getTickUnit() getTicks() notifyListeners(AxisChangeEvent) refreshTicks(Graphics2D,Rectangle2D,Rectangle2D, int) autoRangeStickyZero() setRangeAttribute(Range) setAutoRangeStickyZero(boolean) HorizontalSymbolicAxis(String,String[]) calculateLowestVisibleTickValue() autoRangeIncludesZero() static {} getAutoRangeMinimumSize() calculateVisibleTickCount()

		getPlot() getTickLabelFont() draw(Graphics2D,Rectangle2D,Rectangle2D, int) isVisible() isVerticalTickLabels()
	HorizontalSymbolicAxis_new_2 [-] [-]	DEFAULT_SYMBOLIC_GRID_LINE_PAINT drawSymbolicGridLines(Graphics2D,Rectangle2D,Rectangle2D) getSymbolicGridLine(int) symbolicGridLineList getSymbolicGridPaint() symbolicGridPaint drawSymbolicGridLines(Graphics2D,Rectangle2D,Rectangle2D, boolean)
Month [2]	Month_new_1 [RegularTimePeriod] [-]	toString() equals(Object) getYear() year getYearValue() Month() Month(int, int) getMonth() month Month(Date) getLastMillisecond(Calendar) Month(Date,TimeZone) previous() next() compareTo(Object) getSerialIndex() Month(int,Year) getFirstMillisecond(Calendar)
	Month_new_2 [-] [-]	evaluateAsYear(String) findSeparator(String) parseMonth(String) parseYear(String)
Week [2]	Week_new_1 [RegularTimePeriod] [-]	toString() getYear() year getYearValue() equals(Object) FIRST_WEEK_IN_YEAR LAST_WEEK_IN_YEAR Week(int, int)

		Week() getWeek() week Week(Date) next() getSerialIndex() compareTo(Object) previous() Week(int,Year) getFirstMillisecond(Calendar) getLastMillisecond(Calendar) Week(Date,TimeZone)
	Week_new_2 [-] [-]	stringToWeek(String) findSeparator(String) parseWeek(String) evaluateAsYear(String)
XYSeries [2]	XYSeries_new_1 [Series] [-]	update(int,Number) getYValue(int) getXValue(int) allowDuplicateXValues getMaximumItemCount() setMaximumItemCount(int) maximumItemCount equals(Object) getDataPair(int) delete(int, int) getItemCount() data clear() add(XYDataPair) getName() XYSeries(String, boolean) clone() addChangeListener(SeriesChangeListener) fireSeriesChanged() createCopy(int, int) removeChangeListener(SeriesChangeListener)
	XYSeries_new_2 [-] [-]	add(double, double) add(double,Number) add(Number,Number) XYSeries(String) sampleFunction2D(Function2D, double, double, int,String)
ChartUtilities	ChartUtilities_new_1	ChartUtilities()

[2]	[-] [-]	<p>DEFAULT_PNG_COMPRESSION</p> <p>DEFAULT_JPEG_QUALITY</p> <p>writeImageMap(PrintWriter,String,ChartRenderingInfo)</p> <p>writeImageMap(PrintWriter,String,ChartRenderingInfo, boolean)</p> <p>writeScaledChartAsPNG(OutputStream,JFreeChart, int, int, int, int)</p> <p>writeChartAsJPEG(OutputStream,JFreeChart, int, int)</p> <p>writeChartAsJPEG(OutputStream, float,JFreeChart, int, int)</p> <p>writeChartAsJPEG(OutputStream, float,JFreeChart, int, int,ChartRenderingInfo)</p> <p>writeBufferedImageAsJPEG(OutputStream, float,BufferedImage)</p> <p>writeBufferedImageAsJPEG(OutputStream,BufferedImage)</p> <p>writeChartAsJPEG(OutputStream,JFreeChart, int, int,ChartRenderingInfo)</p> <p>saveChartAsJPEG(File,JFreeChart, int, int,ChartRenderingInfo)</p> <p>saveChartAsJPEG(File, float,JFreeChart, int, int,ChartRenderingInfo)</p> <p>saveChartAsJPEG(File, float,JFreeChart, int, int)</p> <p>saveChartAsJPEG(File,JFreeChart, int, int)</p>
	<p>ChartUtilities_new_2</p> <p>[-]</p> <p>[-]</p>	<p>writeChartAsPNG(OutputStream,JFreeChart, int, int, boolean, int)</p> <p>writeChartAsPNG(OutputStream,JFreeChart, int, int,ChartRenderingInfo, boolean, int)</p> <p>writeBufferedImageAsPNG(OutputStream,BufferedImage)</p> <p>writeBufferedImageAsPNG(OutputStream,BufferedImage, boolean, int)</p> <p>writeChartAsPNG(OutputStream,JFreeChart, int, int,ChartRenderingInfo)</p> <p>writeChartAsPNG(OutputStream,JFreeChart, int, int)</p> <p>saveChartAsPNG(File,JFreeChart, int, int)</p> <p>saveChartAsPNG(File,JFreeChart, int, int,ChartRenderingInfo)</p> <p>saveChartAsPNG(File,JFreeChart, int, int,ChartRenderingInfo, boolean, int)</p>