

# Java Visualize Library Document

제작자 : 양성준 (egod1537@gmail.com)

Android 환경에서 쉽게 자료구조를 시각화 할 수 있도록 작성된 라이브러리입니다.

## 목차

---

### 1 사용 방법

- 1.1 라이브러리 적용
- 1.2 GraphicView 클래스

### 2 시각화 가능한 자료구조

- 2.1 1차원 배열 (CList 클래스)
- 2.2 2차원 배열 (CGrid 클래스)
- 2.3 좌표 평면 (CPlane 클래스)
- 2.4 그래프 (CGraph 클래스)

### 3 View 활용법

- 3.1 LinearView

---

## 1 사용방법

---

라이브러리 적용 및 GraphicView 클래스에 대해 설명하는 Section입니다.

### 1.1 라이브러리 적용

드라이브에 있는 프로젝트의 클래스를 직접 import할 수 있지만 .aar 파일을 사용하면 손쉽게 라이브러리를 import할 수 있습니다.

- File -> Project Structure -> All Dependencies -> + button -> JAR/AAR Dependency -> library .aar file 선택

라이브러리를 import하면 바로 사용할 수 있습니다.

1. 시각화 될 영역을 정의해주는 GraphicView 클래스의 인스턴스를 생성합니다. 생성자에는 Context를 인자로 넘겨줍니다.
2. 인스턴스를 생성한 후 Initialize 함수를 호출합니다.

3. 시각화 할 자료 구조 인스턴스의 생성자에 앞서 정의한 `GraphicView`의 인스턴스를 인자로 넘겨줍니다.
4. 자료 구조에 포함된 함수를 사용하여 시각화 합니다.

## 1.2 `GraphicView` 클래스

`GraphicView` 클래스는 시각화 될 영역을 정의하고 그려주는 역할을 하는 클래스입니다. 클래스의 메서드는 도형을 그리는 것으로 이루어져 있습니다.

`GraphicView` 클래스의 생성자는 `Context` 인자를 필요로 합니다. **주의할 점은 인스턴스 생성 후 `Initialize` 메서드를 무조건 호출하여 그려질 영역을 정의해주어야 합니다.**

`GraphicView`는 독립적으로도 사용될 수 있지만 시각화 가능한 자료구조와 연결하여 사용합니다.

분류	정의	설명
생성자	<code>GraphicView(context)</code>	클래스의 기본 생성자입니다. <code>Layout</code> 에서 그려질 <code>Context</code> 를 넣어주면 됩니다.
메서드	<code>Initialize(x, y, width, height)</code>	그려질 영역을 초기화합니다.
메서드	<code>Initialize(Rect)</code>	그려질 영역을 초기화합니다.
메서드	<code>width()</code>	현재 영역의 가로 길이를 가져옵니다.
메서드	<code>height()</code>	현재 영역의 세로 길이를 가져옵니다.
메서드	<code>draw()</code>	현재 상태를 <code>Render Queue</code> 에 넣습니다.
메서드	<code>render()</code>	<code>Render Queue</code> 의 <code>front</code> 에 있는 꺼내 그립니다.
메서드	<code>renderClear()</code>	영역을 하양색으로 초기화합니다.
메서드	<code>setNextDrawOnTouch(onoff)</code>	해당 영역을 <code>Touch</code> 시 <code>render</code> 메서드가 호출되는 기능을 <code>on/off</code> 합니다. 기본적으로 <code>on</code> 되어 있습니다.
그리기	<code>drawLine(sx, sy, ex, ey, paint)</code>	$(sx, sy) - (ex, ey)$ 선을 그립니다. 그리기 옵션은 <code>paint</code> 입니다.
그리기	<code>drawLine(sp, ep, paint)</code>	<code>Point sp - ep</code> 선을 그립니다. 그리기 옵션을 <code>paint</code> 입니다.
그리기	<code>drawWireRect(x, y, w, h, paint)</code>	$(x, y)$ 에서 $(w, h)$ 길이를 가지고 외곽선만 있는 직사각형을 그립니다. 그리기 옵션은 <code>paint</code> 입니다.
그리기	<code>drawWireRect(rect, paint)</code>	위치와 크기가 <code>rect</code> 이고 외곽선만 있는 직사각형을 그립니다. 그리기 옵션은 <code>paint</code> 입니다.
그리기	<code>drawFillRect(x, y, w, h, paint)</code>	$(x, y)$ 에서 $(w, h)$ 길이를 가지고 면이 채워져 있는 직사각형을 그립니다. 그리기 옵션은 <code>paint</code> 입니다.
그리기	<code>drawFillRect(rect, paint)</code>	위치와 크기가 <code>rect</code> 이고 면이 채워져 있는 직사각형을 그립니다. 그리기 옵션은 <code>paint</code> 입니다.
그리기	<code>drawRect(x, y, w, h, wirePaint, fillPaint)</code>	$(x, y)$ 에서 $(w, h)$ 길이를 가진 직사각형을 그립니다. 외곽선 그리기 옵션은 <code>wirePaint</code> , 안쪽 그리기 옵션

		은 fillPaint입니다.
그리기	drawWirePolygon(poly, paint)	다각형 poly의 외곽선을 그립니다. 그리기 옵션은 paint입니다.
그리기	drawFillPolygon(poly, paint)	다각형 poly의 면을 그립니다. 그리기 옵션은 paint입니다.
그리기	drawPolygon(poly, wirePaint, fillPaint)	외곽선과 면이 있는 다각형 poly를 그립니다. 외곽선 그리기 옵션은 wirePaint, 면 그리기 옵션은 fillPaint입니다.
그리기	drawText(x, y, text, paint)	(x, y) 위치에 text를 그립니다. 그리기 옵션은 paint입니다.
그리기	drawRectWithText(x, y, w, h, text, wirePaint, fillPaint, textPaint)	(x, y)에서 (w, h) 길이를 가지고 있는 직사각형과 Text를 그립니다. 옵션은 기본 메서드와 동일합니다.
그리기	drawWireCircle(point, r, paint)	중심이 point이고 반지름이 r인 원의 외곽선을 그립니다. 그리기 옵션은 paint입니다.
그리기	drawFillCircle(point, r, paint)	중심이 point이고 반지름이 r인 원의 면을 그립니다. 그리기 옵션은 paint입니다.
그리기	drawCircle(point, r, wirePaint, fillPaint)	중심이 point이고 반지름이 r인 원을 그립니다. 외곽선 그리기 옵션은 wirePaint, 면 그리기 옵션은 fillPaint입니다.

## 2 시각화 가능한 자료구조

흔히 사용되는 4가지 자료구조가 구현되어 있습니다. 각 섹션마다 메서드의 정의와 Example이 포함되어 있습니다.

각 클래스의 메서드는 크게 생성자, 메서드, 색깔, 마크 관련 함수로 이루어져 있습니다.

### 2.1 1차원 배열 (CList 클래스)

분류	정의	설명
생성자	CList(panel)	panel에 그리는 클래스를 생성합니다.
생성자	CList(panel, list)	panel에 그리고 list와 동일한 원소를 가진 클래스를 생성합니다. (deep copy)
메서드	setGraphicPanel(panel)	시각화 할 패널을 설정합니다.
메서드	add(x)	x 원소를 추가합니다.
메서드	add(idx, x)	인덱스 idx에 x 원소를 추가합니다.
메서드	addAll(collection)	여러 원소를 추가합니다.
메서드	remove(x)	x 원소를 제거합니다.
메서드	remove(idx)	인덱스 idx 원소를 제거합니다.
메서드	clear()	모든 원소를 제거합니다.
메서드	clone()	List를 복제하여 반환합니다.
색깔	setColor(idx, color)	인덱스 idx의 색깔을 color로 설정합니다.
색깔	setColors(idxs, color)	여러 인덱스 idxs의 색깔을 color로 설정합니다.

		다.
색깔	clearColors()	모든 color를 하양색으로 설정합니다.
마크	addLineMark(idx, text="", up=true)	up방향에 text가 표시되어 있는 선 마크를 인덱스 idx에 표시합니다.
마크	removeLineMark(idx)	인덱스 idx의 선 마크를 제거합니다.
마크	clearLineMark()	모든 선 마크를 제거합니다.
마크	addCheckMark(idx, up=true)	up방향에 있는 체크 마크를 인덱스 idx에 표시합니다.
마크	addCheckMarks(idxs, up=true)	up 방향인 체크 마크를 여러 인덱스 idxs에 표시합니다.
마크	removeCheckMark(idx)	인덱스 idx의 체크 마크를 제거합니다.
마크	clearCheckMark()	모든 체크 마크를 제거합니다.
마크	addRangeMark(l, r, text="", up=true)	up 방향에 text가 표시되어 있는 구간 [l, r] 마크를 추가합니다.
마크	removeRangeMark(l, r)	구간 [l, r] 마크를 제거합니다.
마크	clearRangeMark()	모든 구간 마크를 제거합니다.
마크	addRectMark(l, r, wireColor=BLACK, fillcolor=TRANSPARENT)	구간 [l, r]에 네모 마크를 추가합니다. 모서리의 색깔은 wireColor, 면의 색깔은 wireColor입니다
마크	removeRectMark(l, r)	구간 [l, r]인 네모 마크를 제거합니다.
마크	removeRectMark()	모든 네모 마크를 제거합니다.
마크	clearMarks()	모든 마크를 제거합니다.
메서드	draw()	현재 자료 구조를 시각화 합니다.

## 2.2 2차원 배열 (CGrid)

분류	정의	설명
생성자	CGrid(panel)	panel에 그리는 클래스를 생성합니다.
생성자	CGrid(panel, grid)	panel에 그리고 grid와 동일한 원소를 가진 클래스를 생성합니다. (deep copy)
메서드	setGraphicPanel(panel)	시각화 할 패널을 설정합니다.
메서드	get(x)	인덱스 x 행을 가져옵니다.
메서드	get(x, y)	인덱스 x, y의 값을 가져옵니다.
메서드	add(list)	list 행을 추가합니다.
메서드	add(idx, list)	인덱스 idx에 list 행을 추가합니다.
메서드	addAll(collection)	collection의 행을 추가합니다.
메서드	remove(x)	x와 동일한 행을 제거합니다.
메서드	remove(idx, x)	x 행의 원소 중 x와 동일한 원소를 제거합니다.
메서드	remove(idx)	인덱스 idx 행을 제거합니다.
메서드	remove(x, y)	x 행, y 열의 원소를 제거합니다.
메서드	clear(idx)	인덱스 idx 행을 제거합니다.

메서드	clear()	모든 원소를 제거합니다.
색깔	setColor(x, y, color)	x 행 y 열의 색깔을 color로 설정합니다.
색깔	setColors(x, idxs, color)	x 행의 여러 인덱스 idxs 열의 색깔을 color로 설정합니다.
색깔	setColor(x, color)	x 행의 모든 색깔을 color로 설정합니다.
색깔	setColor(color)	모든 색깔을 color로 설정합니다.
색깔	clearColor(x, y)	x 행 y 열의 색깔을 하양색으로 초기화합니다.
색깔	clearColor(x)	x 행의 모든 색깔을 하양색으로 초기화합니다.
색깔	clearColor()	모든 색깔을 하양색으로 초기화합니다.
마크	addLineMark(idx, text="", dir=up)	dir 방향에 text가 표시되어 있는 선 마크를 인덱스 idx에 추가합니다.
마크	removeLineMark(idx)	인덱스 idx에 있는 선 마크를 제거합니다.
마크	clearLineMarks()	모든 선 마크를 제거합니다.
마크	addCheckMark(idx, dir=up)	dir 방향의 체크 마크를 인덱스 idx에 추가합니다.
마크	removeCheckMark(idx)	인덱스 idx의 체크 마크를 제거합니다.
마크	clearCheckMark()	모든 체크 마크를 제거합니다.
마크	addRangeMark(l, r, text="", dir=up)	dir 방향에 text가 표시되어 있는 구간 [l, r] 마크를 추가합니다.
마크	removeRangeMark(l, r)	구간 [l, r] 마크를 제거합니다.
마크	clearRangeMarks()	모든 구간 마크를 제거합니다.
마크	addRectMark(sx, sy, ex, ey, wireColor=BLACK, fillColor=TRANSPARENT)	[sx, sy] x [ex, ey] 영역을 표시하는 네모 마크를 추가합니다. wireColor는 네모의 모서리 색깔, fillColor는 네모의 면 색깔입니다.
마크	removeRectMark(sx, sy, ex, ey)	[sx, sy] x [ex, ey] 영역의 네모 마크를 제거합니다.
마크	clearRectMarks()	모든 네모 마크를 제거합니다.
메서드	draw()	현재 자료 구조를 시각화 합니다.

## 2.3 좌표 평면 (CPlane)

좌표 평면 클래스는 PointF 클래스의 원소를 기본으로 관리하고 있습니다.

분류	정의	설명
생성자	CPlane(panel)	panel에 그리는 클래스를 생성합니다.
생성자	CPlane(panel, points)	panel에 그리고 points와 동일한 원소를 가진 클래스를 생성합니다. (deep copy)
메서드	setGraphicPanel(panel)	시각화 할 패널을 설정합니다.
메서드	add(x)	점 x를 추가합니다.
메서드	add(idx, x)	인덱스 idx에 점 x를 추가합니다.
메서드	addAll(collection)	Collection 원소를 추가합니다.
메서드	remove(x)	점 x와 동일한 원소를 제거합니다.

메서드	remove(idx)	인덱스 idx에 있는 원소를 제거합니다.
메서드	clear()	모든 원소를 제거합니다.
메서드	setAutoXYRange(flag)	자동으로 평면의 x, y의 범위를 자동으로 할 지 설정합니다. 기본값은 true입니다.
메서드	setXRange(xl, xr)	평면의 x범위를 [xl, xr]으로 설정합니다.
메서드	setYRange(yl, yr)	평면의 y범위를 [yl, yr]으로 설정합니다.
마크	addPolygonMark(poly, wireColor=WHITE, fillColor=TRANSPARENT, alpha=1)	다각형 poly 마크를 추가합니다. wireColor는 모서리 색깔, fillColor는 면의 색깔입니다.
마크	removePolygonMark(poly)	다각형 poly 마크를 제거합니다.
마크	clearPolygonMarks()	모든 다각형 마크를 제거합니다.
마크	addFuncMark(function, color=BLACK)	함수 function 마크를 추가합니다. color는 함수의 색깔입니다.
마크	addPolynomialMark(poly, color=BLACK)	다항 함수 function 마크를 추가합니다. color는 다항 함수의 색깔입니다.
마크	clearFuncMarks()	모든 함수 마크를 제거합니다.
메서드	draw()	현재 자료 구조를 시각화 합니다.

## 2.4 그래프 (CGraph)

분류	정의	설명
생성자	CGraph(panel)	panel에 그리는 클래스를 생성합니다.
생성자	CGraph(panel, verts)	panel에 그리고 points와 동일한 정점 가진 클래스를 생성합니다. (deep copy)
생성자	CGraph(panel, verts, edges)	panel에 그리고 points와 동일한 정점, edges와 동일한 간선을 가진 클래스를 생성합니다. (deep copy)
메서드	setGraphicPanel(panel)	시각화 할 패널을 설정합니다.
메서드	setDirectedGraph(directed)	그래프의 방향성 여부를 설정합니다.
메서드	setWeightGraph(weight)	그래프의 가중치 여부를 설정합니다.
메서드	setVertSize(size=auto)	그려질 정점의 크기를 설정합니다.
메서드	setTextSize(size=auto)	그려질 글자의 크기를 설정합니다.
메서드	add_vert(point, color=WHITE)	point 위치에 정점을 추가합니다. color는 정점의 색깔입니다.
메서드	addAll_vert(points, colors=WHITE)	points 위치에 정점들을 추가합니다. colors는 정점들의 색깔입니다.
메서드	remove_vert(idx)	인덱스 idx의 정점을 제거합니다.
메서드	remove_vert(point)	point와 동일한 위치의 정점을 제거합니다.
색깔	setVertColor(idx, color)	인덱스 idx의 정점 색깔을 설정합니다.

색깔	setVertColor(color=WHITE)	모든 정점 색깔을 color로 변경합니다.
색깔	setVertText(idx, text="")	인덱스 idx의 정점 내용을 text로 설정합니다.
메서드	add_edge(edge, color=BLACK)	edge 간선을 추가합니다. color는 간선의 색깔입니다.
메서드	addAll_edge(edges, colors=BLACK)	edges 간선들을 추가합니다. colors는 간선들의 색깔입니다.
메서드	remove_edge(edge)	edge 간선과 동일한 원소를 제거합니다.
색깔	setEdgeColor(idx, color)	인덱스 idx의 간선 색깔을 color로 설정합니다.
색깔	setEdgeColor(color=BLACK)	모든 간선의 색깔을 color로 설정합니다.
메서드	setVertActive(idx, active)	인덱스 idx의 정점 활성화 여부를 설정합니다.
메서드	setEdgeActive(idx, active)	인덱스 idx의 간선 활성화 여부를 설정합니다.
메서드	draw()	현재 자료 구조를 시각화 합니다.

### 3. View 활용법

GraphicPanel 클래스는 View를 상속 받아 LinearView 같은 곳에 넣을 수 있습니다.

자세한 내용은 예제 프로젝트를 확인해주세요.