

## 09. 그래픽과 이미지

## 학습목표

---

- 캔버스에 도형을 그리는 방법을 익힌다.
- 이미지 파일을 처리하는 방식을 알아본다.
- 영상처리 앱을 작성한다.

## 차례

---

01 그래픽

02 이미지

## 1. 그래픽 ▶ 캔버스와 페인트 기본[1/3]

### ❖ 캔버스와 페인트

- ✓ 화면에 도형을 그릴 때 사용되는 Canvas와 Paint 클래스



그림 9-1 Canvas와 Paint 클래스

## 1. 그래픽 ▶ 캔버스와 페인트 기본[2/3]

---

### ❖ 캔버스와 페인트

- ✓ **android.graphics.Canvas** 클래스의 점을 찍는 메소드의 원형

```
public void drawPoint (float x, float y, Paint paint)
```

- ✓ **android.graphics.Paint** 클래스에서 색상을 지정하는 메소드의 원형

```
public void setColor (int color)
```

## 1. 그래픽 ▶ 캔버스와 페인트 기본[3/3]

### ❖ 캔버스와 페인트

- ✓ 그래픽을 표현할 때는 View 클래스를 재정의하는 형태를 많이 사용

```
public void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    setContentView(new 재정의한 클래스 이름(this));  
}  
  
private static class 재정의한 클래스 이름 extends View {  
    public 재정의한 클래스 이름(Context context) {  
        super(context);  
    }  
    @Override  
    protected void onDraw(Canvas canvas) {  
        super.onDraw(canvas);  
        // 여기에 화면에 그려질 내용을 코딩  
    }  
}
```


## 1. 그래픽 ▶ 그래픽 처리 기본[1/2]

### ❖ 그래픽 기본 Java 코드

- ✓ 그래픽을 출력할 때는 주로 `View.onDraw()` 메소드를 오버라이딩해서 사용

예제 9-1 그래픽 기본의 Java 코드

```
1 public void onCreate(Bundle savedInstanceState) {
2     super.onCreate(savedInstanceState);
3     setContentView(new MyGraphicView(this));
4 }
5
6 private static class MyGraphicView extends View {
7     public MyGraphicView(Context context) {
8         super(context);
9     }
10
11     @Override
12     protected void onDraw(Canvas canvas) {
13         super.onDraw(canvas);
14         Paint paint = new Paint();
15         paint.setAntiAlias(true);
16         paint.setColor(Color.GREEN);
17         canvas.drawLine(10, 10, 300, 10, paint);
18
19         paint.setColor(Color.BLUE);
20         paint.setStrokeWidth(5);
21         canvas.drawLine(10, 30, 300, 30, paint);
22
23         paint.setColor(Color.RED);
24         paint.setStrokeWidth(0);
25
26         paint.setStyle(Paint.Style.FILL);
27         Rect rect1 = new Rect(10, 50, 100+100, 50+100);
28         canvas.drawRect(rect1, paint);
29
30         paint.setStyle(Paint.Style.STROKE);
31         Rect rect2 = new Rect(130, 50, 130+100, 50+100);
32         canvas.drawRect(rect2, paint);
33
34         RectF rect3 = new RectF(250, 50, 250+100, 50+100);
35         canvas.drawRoundRect(rect3, 20, 20, paint);
36
37         canvas.drawCircle(60, 220, 50, paint);
38
39         paint.setStrokeWidth(5);
40         Path path1 = new Path();
41         path1.moveTo(10, 290);
42         path1.lineTo(10+50, 290+50);
43         path1.lineTo(10+100, 290);
44         path1.lineTo(10+150, 290+50);
45         path1.lineTo(10+200, 290);
46         canvas.drawPath(path1, paint);
47
48         paint.setStrokeWidth(0);
49         paint.setTextSize(30);
50         canvas.drawText("안드로이드", 10, 390, paint);
51     }
52 }
```



## Coding Exercise

---

```
package com.cookandroid.project9;

import android.support.v7.app.AppCompatActivity;
import android.content.Context;
import android.graphics.Canvas;
import android.graphics.Color;
import android.graphics.Paint;
import android.graphics.Path;
import android.graphics.Rect;
import android.graphics.RectF;
import android.os.Bundle;
import android.view.View;
```



## Coding Exercise

---

```
public class MainActivity extends AppCompatActivity {

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(new MyGraphicView(this));
    }

    private static class MyGraphicView extends View {
        public MyGraphicView(Context context) {
            super(context);
        }
    }
}
```

## Coding Exercise

---

```
@Override
protected void onDraw(Canvas canvas) {
    super.onDraw(canvas);
    Paint paint = new Paint();
    paint.setAntiAlias(true);
    paint.setColor(Color.GREEN);
    canvas.drawLine(10, 10, 300, 10, paint);

    paint.setColor(Color.BLUE);
    paint.setStrokeWidth(5);
    canvas.drawLine(10, 30, 300, 30, paint);

    paint.setColor(Color.RED);
    paint.setStrokeWidth(0);

    paint.setStyle(Paint.Style.FILL);
    Rect rect1 = new Rect(10, 50, 10 + 100, 50 + 100);
    canvas.drawRect(rect1, paint);

    paint.setStyle(Paint.Style.STROKE);
    Rect rect2 = new Rect(130, 50, 130 + 100, 50 + 100);
    canvas.drawRect(rect2, paint);
}
```

## Coding Exercise

```
RectF rect3 = new RectF(250, 50, 250 + 100, 50 + 100);
canvas.drawRoundRect(rect3, 20, 20, paint);

canvas.drawCircle(60, 220, 50, paint);

paint.setStrokeWidth(5);
Path path1 = new Path();
path1.moveTo(10, 290);
path1.lineTo(10 + 50, 290 + 50);
path1.lineTo(10 + 100, 290);
path1.lineTo(10 + 150, 290 + 50);
path1.lineTo(10 + 200, 290);
canvas.drawPath(path1, paint);

paint.setStrokeWidth(0);
paint.setTextSize(30);
canvas.drawText("안드로이드", 10, 390, paint);
    }
}
}
```

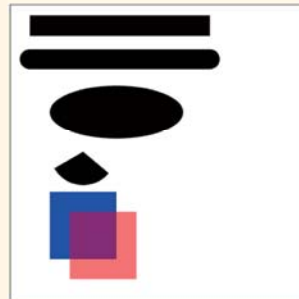
## 1. 그래픽 ▶ 그래픽 처리 기본[2/2]

---

### ▶ 직접 풀어보기 9-1

그림과 같은 화면을 출력하도록 다음 메소드를 사용하여 Java를 코딩하라.

- `Paint.setStrokeCap()`
- `Canvas.drawOval()`
- `Paint.setColor(Color.argb())`



## 1. 그래픽 ▶ 터치 이벤트[1/7]

---

### ❖ 터치 이벤트

- ✓ 화면에 생성한 뷰를 터치하면 Touch 이벤트가 발생
- ✓ View 클래스의 `onTouchEvent()` 메소드를 오버라이드해서 코딩

```
public boolean onTouchEvent(MotionEvent event) {  
    switch (event.getAction()) {  
        case MotionEvent.ACTION_DOWN:  
            // 손가락으로 화면을 누르기 시작했을 때 할 일  
            break;  
        case MotionEvent.ACTION_MOVE:  
            // 터치 후 손가락을 움직일 때 할 일  
            break;  
        case MotionEvent.ACTION_UP:  
            // 손가락을 화면에서 떼 때 할 일  
            break;  
        case MotionEvent.ACTION_CANCEL:  
            // 터치가 취소될 때 할 일  
            break;  
        default:  
            break;  
    }  
    return true;  
}
```

## 1. 그래픽 ▶ 터치 이벤트[2/7]

### 실습 9-1 간단 그림판 앱 만들기

- ❖ 안드로이드 프로젝트 생성
  - ✓ 프로젝트 이름 : Project9\_1
  - ✓ 패키지 이름 : com.cookandroid.project9\_1
- ❖ 화면 디자인 및 편집
  - ✓ Java 코드로만 작성



그림 9-2 간단 그림판 앱 결과 화면

## 1. 그래픽 ▶ 터치 이벤트[3/7]

### 실습 9-1 간단 그림판 앱 만들기

#### ❖ Java 코드 작성 및 수정

#### ✓ View 클래스의 상속을 받는 MyGraphicView 클래스를 만들

예제 9-2 간단 그림판의 Java 코드 1

```
1  ~~~~ 중간 생략(import문) ~~~~
2  public class MainActivity extends AppCompatActivity {
3      final static int LINE = 1, CIRCLE = 2;
4      static int curShape = LINE;
5
6      @Override
7      public void onCreate(Bundle savedInstanceState) {
8          super.onCreate(savedInstanceState);
9          setContentView(new MyGraphicView(this));
10         setTitle("간단 그림판");
11     }
12
13     private static class MyGraphicView extends View {
14         public MyGraphicView(Context context) {
15             super(context);
16         }
17     }
18 }
19 }
```

## 1. 그래픽 ▶ 터치 이벤트[4/7]

### 실습 9-1 간단 그림판 앱 만들기

#### ❖ 옵션 메뉴 작성

- ✓ 선 그리기, 원 그리기 옵션 메뉴 만들
- ✓ 항목을 클릭하면 `curShape` 변수에 선택한 전역상수를 대입
- ✓ `onCreateOptionsMenu()`와 `onOptionsItemSelected()` 메소드 자동 완성

예제 9-3 간단 그림판의 Java 코드 2

```
1 public boolean onCreateOptionsMenu(Menu menu) {
2     super.onCreateOptionsMenu(menu);
3     menu.add(0, 1, 0, "선 그리기");
4     menu.add(0, 2, 0, "원 그리기");
5     return true;
6 }
7 public boolean onOptionsItemSelected(MenuItem item) {
8     switch (item.getItemId()) {
9         case 1:
10            curShape = LINE; // 선
11            return true;
12         case 2:
13            curShape = CIRCLE; // 원
14            return true;
15     }
16     return super.onOptionsItemSelected(item);
17 }
```





## 1. 그래픽 ▶ 터치 이벤트[5/7]

### 실습 9-1 간단 그림판 앱 만들기

- ❖ MyGraphicView 클래스에 터치와 관련된 메소드를 완성
  - ✓ MyGraphicView의 전역변수 시작x, 시작y, 끝x, 끝y 및 반지름 변수를 선언
  - ✓ onTouchEvent() 메소드 자동 완성 후 자동 완성 외의 코드 완성

예제 9-4 간단 그림판의 Java 코드 3

```
1 private static class MyGraphicView extends View {
2     int startX = -1, startY = -1, stopX = -1, stopY = -1;
3     public MyGraphicView(Context context) {
4         super(context);
5     }
6     @Override
7     public boolean onTouchEvent(MotionEvent event) {
8         switch (event.getAction()) {
9             case MotionEvent.ACTION_DOWN:
10                startX = (int) event.getX();
11                startY = (int) event.getY();
12                break;
13             case MotionEvent.ACTION_MOVE:
14             case MotionEvent.ACTION_UP:
15                stopX = (int) event.getX();
16                stopY = (int) event.getY();
17                this.invalidate();
18                break;
19            }
20            return true;
21        }
22    }
23 }
```

## 1. 그래픽 ▶ 터치 이벤트[6/7]

### 실습 9-1 간단 그림판 앱 만들기

- ❖ 실제로 화면에 도형이 그려질 `onDraw()` 메소드 완성
  - ✓ `MyGraphicView`의 내부에 `onDraw()`를 자동 완성하고 나머지를 코딩
  - ✓ 페인트에 선의 두께, 채우기 여부, 선의 색상을 지정
  - ✓ `switch()`~`case`문으로 메뉴에서 선택한 내용에 따라 선 또는 원을 그림

예제 9-5 간단 그림판의 Java 코드 4

```
1  protected void onDraw(Canvas canvas) {
2      super.onDraw(canvas);
3      Paint paint = new Paint();
4      paint.setAntiAlias(true);
5      paint.setStrokeWidth(5);
6      paint.setStyle(Paint.Style.STROKE);
7      paint.setColor(Color.RED);
8
9      switch (curShape) {
10         case LINE:
11             canvas.drawLine(startX, startY, stopX, stopY, paint);
12             break;
13         case CIRCLE:
14             int radius = (int) Math.sqrt(Math.pow(stopX - startX, 2)
15                                     + Math.pow(stopY - startY, 2));
16             canvas.drawCircle(startX, startY, radius, paint);
17             break;
18         }
19     }
```

## Coding Exercise 실습 9-1 : activity\_main.xml

---

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Hello World!"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

</android.support.constraint.ConstraintLayout>
```

## Coding Exercise 실습 9-1 : MainActivity.java

---

```
package com.cookandroid.project9_1;

import android.graphics.Canvas;
import android.graphics.Color;
import android.graphics.Paint;
import android.support.v7.app.AppCompatActivity;
import android.content.Context;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;
import android.view.MotionEvent;
import android.view.View;

public class MainActivity extends AppCompatActivity {

    final static int LINE = 1, CIRCLE = 2;
    static int curShape = LINE;
```

## Coding Exercise

---

```
@Override
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(new MyGraphicView(this));
    setTitle("간단 그림판");
}

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    super.onCreateOptionsMenu(menu);
    menu.add(0, 1, 0, "선 그리기");
    menu.add(0, 2, 0, "원 그리기");
    return true;
}
```

## Coding Exercise

---

```
@Override
public boolean onOptionsItemSelected(MenuItem item) {
    switch (item.getItemId()) {
        case 1:
            curShape = LINE; // 선
            return true;
        case 2:
            curShape = CIRCLE; // 원
            return true;
    }
    return super.onOptionsItemSelected(item);
}
```

## Coding Exercise

---

```
private static class MyGraphicView extends View {
    int startX = -1, startY = -1, stopX = -1, stopY = -1;
    public MyGraphicView(Context context) {
        super(context);
    }

    @Override
    public boolean onTouchEvent(MotionEvent event) {
        switch (event.getAction()) {
            case MotionEvent.ACTION_DOWN:
                startX = (int) event.getX();
                startY = (int) event.getY();
                break;
            case MotionEvent.ACTION_MOVE:
            case MotionEvent.ACTION_UP:
                stopX = (int) event.getX();
                stopY = (int) event.getY();
                this.invalidate();
                break;
        }
        return true;
    }
}
```

## Coding Exercise

---

```
protected void onDraw(Canvas canvas) {  
    super.onDraw(canvas);  
    Paint paint = new Paint();  
    paint.setAntiAlias(true);  
    paint.setStrokeWidth(5);  
    paint.setStyle(Paint.Style.STROKE);  
    paint.setColor(Color.RED);  
  
    switch (curShape) {  
        case LINE:  
            canvas.drawLine(startX, startY, stopX, stopY, paint);  
            break;  
        case CIRCLE:  
            int radius = (int) Math.sqrt(Math.pow(stopX - startX, 2)  
                + Math.pow(stopY - startY, 2));  
            canvas.drawCircle(startX, startY, radius, paint);  
            break;  
    }  
}  
}
```

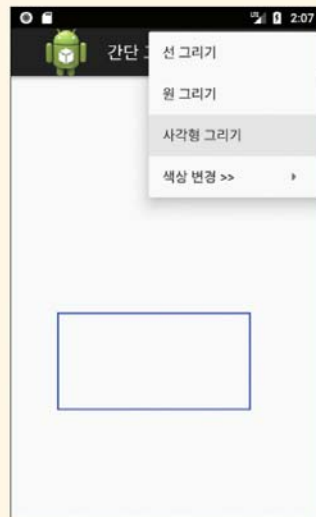


## 1. 그래픽 ▶ 터치 이벤트[7/7]

### ▶ 직접 풀어보기 9-2

[실습 9-1]을 다음과 같이 수정하라.

- 클릭한 두 점을 끝점으로 하는 사각형이 추가로 그려지게 한다.
- 색상이 옵션 메뉴에서 선택되게 한다. 색상은 서브 메뉴로 나오게 하고 빨강, 초록, 파랑만 사용한다.



## 2. 이미지 ▶ 비트맵 기본[1/3]

---

### ❖ 비트맵(Bitmap)

- ✓ 비트맵(Bitmap) 클래스는 캔버스에 이미지 파일을 보여주기 위해서 사용
  - /res/drawable 폴더에 있는 이미지 파일을 보여주는 onDraw() 메소드

```
protected void onDraw(Canvas canvas) {  
    super.onDraw(canvas);  
  
    Bitmap picture = BitmapFactory.decodeResource(getResources(),  
R.drawable.그림id);  
    canvas.drawBitmap(picture, 시작x, 시작y, null);  
    picture.recycle();  
}
```

## 2. 이미지 ▶ 비트맵 기본[2/3]

---

### ❖ 비트맵(Bitmap)

- SD 카드의 이미지 파일을 보여주는 onDraw( ) 메소드

```
protected void onDraw(Canvas canvas) {  
    super.onDraw(canvas);  
  
    Bitmap picture = BitmapFactory.decodeFile("파일경로 및 파일");  
    canvas.drawBitmap(picture, 시작x, 시작y, null);  
    picture.recycle();  
}
```

- 이미지를 화면 중앙에 출력하기 위한 방법

```
시작x = (View 폭 - 이미지 폭) / 2  
시작y = (View 높이 - 이미지 높이) / 2
```

## 2. 이미지 ▶ 비트맵 기본[3/3]

### ❖ 화면 중앙에 이미지 파일을 출력하는 Java 코드

- ✓ jeju14.jpg를 /res/drawable에 미리 복사

예제 9-6 화면 중앙에 이미지 파일을 출력하는 Java 코드

```
1 public void onCreate(Bundle savedInstanceState) {  
2     super.onCreate(savedInstanceState);  
3     setContentView(new MyGraphicView(this));  
4 }  
5  
6 private static class MyGraphicView extends View {  
7     public MyGraphicView(Context context) {  
8         super(context);  
9     }  
10    @Override  
11    protected void onDraw(Canvas canvas) {  
12        super.onDraw(canvas);  
13        Bitmap picture = BitmapFactory.decodeResource(getResources(),  
14            R.drawable.jeju14);  
14        int picX = (this.getWidth() - picture.getWidth()) / 2;  
15        int picY = (this.getHeight() - picture.getHeight()) / 2;  
16        canvas.drawBitmap(picture, picX, picY, null);  
17        picture.recycle();  
18    }  
19 }
```



## Coding Exercise

---

```
package com.cookandroid.project9;

import android.support.v7.app.AppCompatActivity;
import android.content.Context;
import android.graphics.Bitmap;
import android.graphics.BitmapFactory;
import android.graphics.Canvas;
import android.os.Bundle;
import android.view.View;

public class MainActivity extends AppCompatActivity {

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(new MyGraphicView(this));
    }
}
```

## Coding Exercise

---

```
private static class MyGraphicView extends View {
    public MyGraphicView(Context context) {
        super(context);
    }

    @Override
    protected void onDraw(Canvas canvas) {
        super.onDraw(canvas);
        Bitmap picture = BitmapFactory.decodeResource(getResources(),
            R.drawable.jejul4);
        int picX = (this.getWidth() - picture.getWidth()) / 2;
        int picY = (this.getHeight() - picture.getHeight()) / 2;
        canvas.drawBitmap(picture, picX, picY, null);
        picture.recycle();
    }
}
```

## 2. 이미지 ▶ 이미지의 기하학적 변환[1/2]

### ❖ 화면 중앙에 이미지 파일을 출력하는 Java 코드

✓ 많이 사용되는 Canvas 클래스의 기하학적 메소드



(a) 회전



(b) 이동



(c) 확대



(d) 기울이기

그림 9-3 기하학적 변환 결과

## 2. 이미지 ▶ 이미지의 기하학적 변환[2/2]

### ❖ 기하학적 변환 Java 코드

예제 9-7 기하학적 변환의 Java 코드

```
1  protected void onDraw(Canvas canvas) {  
2      super.onDraw(canvas);  
3      Bitmap picture = BitmapFactory.decodeResource(getResources(), R.drawable.  
        small);  
4  
5      int cenX = this.getWidth() / 2;  
6      int cenY = this.getHeight() / 2;  
7      int picX = (this.getWidth() - picture.getWidth()) / 2;  
8      int picY = (this.getHeight() - picture.getHeight()) / 2;  
9  
10     canvas.rotate(45, cenX, cenY);  
11     canvas.drawBitmap(picture, picX, picY, null);  
12  
13     canvas.translate(-150, 200);  
14     canvas.drawBitmap(picture, picX, picY, null);  
15  
16     canvas.scale(2, 2, cenX, cenY);  
17     canvas.drawBitmap(picture, picX, picY, null);  
18  
19     canvas.skew(0.3f, 0.3f);  
20     canvas.drawBitmap(picture, picX, picY, null);  
21  
22     picture.recycle();  
23 }
```



## Coding Exercise

---

```
package com.cookandroid.project9;

import android.support.v7.app.AppCompatActivity;
import android.content.Context;
import android.graphics.Bitmap;
import android.graphics.BitmapFactory;
import android.graphics.Canvas;
import android.os.Bundle;
import android.view.View;

public class MainActivity extends AppCompatActivity {

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(new MyGraphicView(this));
    }
}
```

## Coding Exercise

---

```
private static class MyGraphicView extends View {
    public MyGraphicView(Context context) {
        super(context);
    }

    @Override
    protected void onDraw(Canvas canvas) {
        super.onDraw(canvas);
        Bitmap picture = BitmapFactory.decodeResource(getResources(),
            R.drawable.small);

        int cenX = this.getWidth() / 2;
        int cenY = this.getHeight() / 2;
        int picX = (this.getWidth() - picture.getWidth()) / 2;
        int picY = (this.getHeight() - picture.getHeight()) / 2;
```

## Coding Exercise

---

```
        canvas.rotate(45, cenX, cenY);
        canvas.drawBitmap(picture, picX, picY, null);

        // canvas.translate(-150, 200);
        // canvas.drawBitmap(picture, picX, picY, null);

        // canvas.scale(2, 2, cenX, cenY);
        // canvas.drawBitmap(picture, picX, picY, null);

        // canvas.skew (0.3f, 0.3f);
        // canvas.drawBitmap(picture, picX, picY, null);

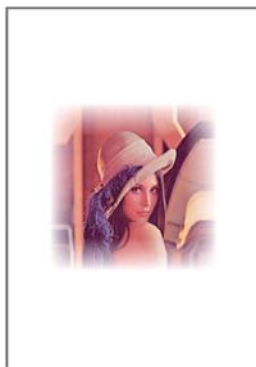
        picture.recycle();
    }
}
```

## 2. 이미지 ▶ 이미지 활용[1/16]

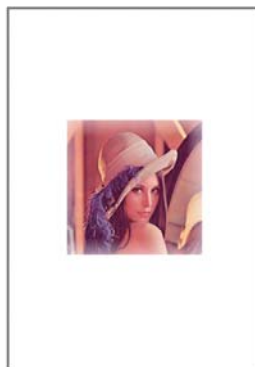
### ❖ 블러링(Blurring)

- ✓ 이미지를 뿌옇게 만드는 것으로 BlurMaskFilter 클래스 사용

```
BlurMaskFilter(반지름, 스타일);
```



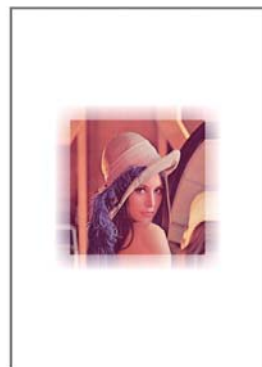
(a) NORMAL



(b) INNER



(c) OUTER



(d) SOLID

그림 9-4 블러링 효과

## 2. 이미지 ▶ 이미지 활용[2/16]

### ❖ 블러링 효과 Java 코드

예제 9-8 블러링 효과의 Java 코드

```
1  protected void onDraw(Canvas canvas) {
2      super.onDraw(canvas);
3      Bitmap picture = BitmapFactory.decodeResource(getResources(),
4                                          R.drawable.lena256);
5
6      int picX = (this.getWidth() - picture.getWidth()) / 2;
7      int picY = (this.getHeight() - picture.getHeight()) / 2;
8
9      Paint paint = new Paint();
10     BlurMaskFilter bMask;
11
12     bMask = new BlurMaskFilter(30, BlurMaskFilter.Blur.NORMAL);
13     paint.setMaskFilter(bMask);
14     canvas.drawBitmap(picture, picX, picY, paint);
15     picture.recycle();
16     ~~~~ 중간 생략(INNER, OUTER, SOLID 스타일) ~~~~
17 }
```

## Coding Exercise

---

```
package com.cookandroid.project9;

import android.support.v7.app.AppCompatActivity;
import android.content.Context;
import android.graphics.Bitmap;
import android.graphics.BitmapFactory;
import android.graphics.BlurMaskFilter;
import android.graphics.Canvas;
import android.graphics.Paint;
import android.os.Bundle;
import android.view.View;
```

## Coding Exercise

---

```
public class MainActivity extends AppCompatActivity {

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(new MyGraphicView(this));
    }

    private static class MyGraphicView extends View {
        public MyGraphicView(Context context) {
            super(context);
        }
    }
}
```

## Coding Exercise

---

```
@Override
protected void onDraw(Canvas canvas) {
    super.onDraw(canvas);
    Bitmap picture = BitmapFactory.decodeResource(getResources(),
        R.drawable.lena256);

    int picX = (this.getWidth() - picture.getWidth()) / 2;
    int picY = (this.getHeight() - picture.getHeight()) / 2;

    Paint paint = new Paint();
    BlurMaskFilter bMask;

    bMask = new BlurMaskFilter(30, BlurMaskFilter.Blur.NORMAL);
    paint.setMaskFilter(bMask);
    canvas.drawBitmap(picture, picX, picY, paint);
}
```



## Coding Exercise

---

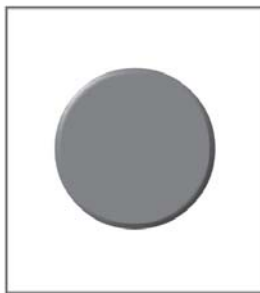
```
// bMask = new BlurMaskFilter(30, BlurMaskFilter.Blur.INNER);
// paint.setMaskFilter(bMask);
// canvas.drawBitmap(picture, picX, picY, paint);
// picture.recycle();
//
// bMask = new BlurMaskFilter(30, BlurMaskFilter.Blur.OUTER);
// paint.setMaskFilter(bMask);
// canvas.drawBitmap(picture, picX, picY, paint);
// picture.recycle();
//
// bMask = new BlurMaskFilter (30, BlurMaskFilter.Blur.SOLID);
// paint.setMaskFilter(bMask);
// canvas.drawBitmap(picture, picX, picY, paint);
// picture.recycle();
    }
}
}
```

## 2. 이미지 ▶ 이미지 활용[3/16]

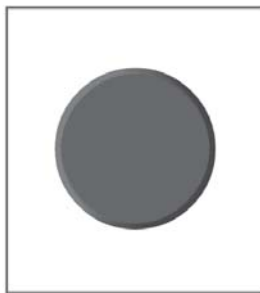
### ❖ 엠보싱(Embossing)

- ✓ 이미지가 볼록하게 튀어나와 보이는 효과로, `EmbossMaskFilter` 클래스 제공

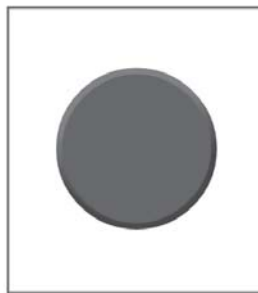
`EmbossMaskFilter(빛의 xyz 방향 1차 배열, 빛의 밝기, 반사 계수, 블러링 크기);`



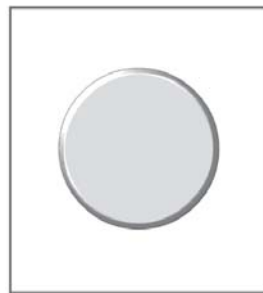
(a) 빛 방향 {3, 3, 3}



(b) 빛 방향 {10, 3, 3}



(c) 빛 방향 {3, 10, 3}



(d) 빛 방향 {3, 3, 10}

그림 9-5 엠보싱 효과

## 2. 이미지 ▶ 이미지 활용[4/16]

### ❖ 엠보싱 효과 Java 코드

예제 9-9 엠보싱 효과의 Java 코드

```
1  protected void onDraw(Canvas canvas) {  
2      super.onDraw(canvas);  
3  
4      int cenX = this.getWidth() / 2 ;  
5      int cenY = this.getHeight() / 2;  
6  
7      Paint paint = new Paint();  
8      paint.setColor(Color.GRAY);  
9      EmbossMaskFilter eMask;  
10  
11     eMask = new EmbossMaskFilter(new float[] {3,3,3}, 0.5f, 5, 10);  
12     paint.setMaskFilter(eMask);  
13     canvas.drawCircle(cenX, cenY, 150, paint);  
14     ~~~~ 중간 생략(빛의 방향을 바꾼 세 가지) ~~~~  
15 }
```

## Coding Exercise

---

```
package com.cookandroid.project9;

import android.support.v7.app.AppCompatActivity;
import android.content.Context;
import android.graphics.Canvas;
import android.graphics.Color;
import android.graphics.EmbossMaskFilter;
import android.graphics.Paint;
import android.os.Bundle;
import android.view.View;
```

## Coding Exercise

---

```
public class MainActivity extends AppCompatActivity {

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(new MyGraphicView(this));
    }

    private static class MyGraphicView extends View {
        public MyGraphicView(Context context) {
            super(context);
        }
    }
}
```

## Coding Exercise

---

```
@Override
protected void onDraw(Canvas canvas) {
    super.onDraw(canvas);

    int cenX = this.getWidth() / 2;
    int cenY = this.getHeight() / 2;

    Paint paint = new Paint();
    paint.setColor(Color.GRAY);
    EmbossMaskFilter eMask;

    eMask = new EmbossMaskFilter(new float[] { 3, 3, 3 }, 0.5f, 5, 10);
    paint.setMaskFilter(eMask);
    canvas.drawCircle(cenX, cenY, 150, paint);
}
```

## Coding Exercise

---

```
// eMask = new EmbossMaskFilter(new float[] {10,3,3}, 0.5f, 5, 10);
// paint.setMaskFilter(eMask);
// canvas.drawCircle(cenX, cenY, 150, paint);

// eMask = new EmbossMaskFilter(new float[] {3,10,3}, 0.5f, 5, 10);
// paint.setMaskFilter(eMask);
// canvas.drawCircle(cenX, cenY, 150, paint);

// eMask = new EmbossMaskFilter(new float[] {3,3,10}, 0.5f, 5, 10);
// paint.setMaskFilter(eMask);
// canvas.drawCircle(cenX, cenY, 150, paint);

    }
}
}
```

## 2. 이미지 ▶ 이미지 활용[5/16]

### ❖ 컬러매트릭스

- ✓ 색상, 밝기 조절 위해 **ColorMatrix**와 **ColorMatrixColorFilter**클래스 사용

```
Paint paint = new Paint();
float[] array = { 4 x 5 배열 };
ColorMatrix cm = new ColorMatrix(array);
paint.setColorFilter(new ColorMatrixColorFilter(cm));
canvas.drawBitmap(...);
```

- ✓ **ColorMatrix**에 사용할 배열(Array)의 각 위치의 값

Red (1)	0	0	0	Brightness(0)
0	Green (1)	0	0	Brightness(0)
0	0	Blue (1)	0	Brightness(0)
0	0	0	Alpha(1)	0



## 2. 이미지 ▶ 이미지 활용[6/16]

### ❖ RGB 색상 대비를 2배로 변경하는 Java 코드

예제 9-10 컬러매트릭스의 Java 코드

```
1  protected void onDraw(Canvas canvas) {  
2  
3      super.onDraw(canvas);  
4  
5      Bitmap picture = BitmapFactory.decodeResource(getResources(),  
6          R.drawable.lena256);  
7  
8      int picX = (this.getWidth() - picture.getWidth()) / 2;  
9      int picY = (this.getHeight() - picture.getHeight()) / 2;  
10  
11     Paint paint = new Paint();  
12     float[] array = { 2, 0, 0, 0, -25,  
13                      0, 2, 0, 0, -25,  
14                      0, 0, 2, 0, -25,  
15                      0, 0, 0, 1, 0 };  
16     ColorMatrix cm = new ColorMatrix(array);  
17     paint.setColorFilter(new ColorMatrixColorFilter(cm));  
18     canvas.drawBitmap(picture, picX, picY, paint);  
19     picture.recycle();  
20 }
```



## Coding Exercise

---

```
package com.cookandroid.project9;

import android.support.v7.app.AppCompatActivity;
import android.content.Context;
import android.graphics.Bitmap;
import android.graphics.BitmapFactory;
import android.graphics.Canvas;
import android.graphics.ColorMatrix;
import android.graphics.ColorMatrixColorFilter;
import android.graphics.Paint;
import android.os.Bundle;
import android.view.View;
```

## Coding Exercise

---

```
public class MainActivity extends AppCompatActivity {

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(new MyGraphicView(this));
    }

    private static class MyGraphicView extends View {
        public MyGraphicView(Context context) {
            super(context);
        }
    }
}
```

## Coding Exercise

---

```
@Override
protected void onDraw(Canvas canvas) {
    super.onDraw(canvas);

    Bitmap picture = BitmapFactory.decodeResource(getResources(),
        R.drawable.lena256);

    int picX = (this.getWidth() - picture.getWidth()) / 2;
    int picY = (this.getHeight() - picture.getHeight()) / 2;

    Paint paint = new Paint();
    float[] array = { 2, 0, 0, 0, -25, 0, 2, 0, 0, -25, 0, 0, 2, 0,
        -25, 0, 0, 0, 1, 0 };
    ColorMatrix cm = new ColorMatrix(array);
    paint.setColorFilter(new ColorMatrixColorFilter(cm));
    canvas.drawBitmap(picture, picX, picY, paint);
    picture.recycle();
}
}
```

## 2. 이미지 ▶ 이미지 활용[7/16]

### 실습 9-2 미니 포토샵 앱 만들기

- ❖ 안드로이드 프로젝트 생성
  - ✓ 프로젝트 이름 : Project9\_2
  - ✓ 패키지 이름 : com.cookandroid.project9\_2
- ❖ 화면 디자인 및 편집
  - ✓ 사용할 그림 파일들을 /res/drawable에 복사
- ❖ AndroidManifest.xml에 아이콘으로 사용할 그림 파일의 id로 변경 및 하드웨어 가속기 기능 끄

```
android:icon="@drawable/그림 파일 id"  
android:hardwareAccelerated="false"
```

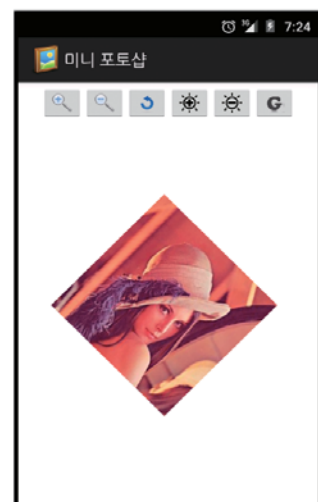


그림 9-6 미니 포토샵 앱 결과 화면

## 2. 이미지 ▶ 이미지 활용[8/16]

### 실습 9-2 미니 포토샵 앱 만들기

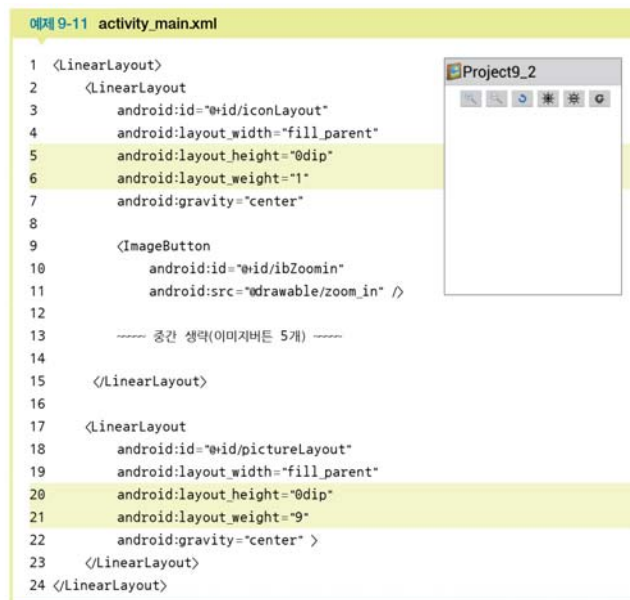
#### ❖ Activity\_main.xml 수정

- ✓ 바깥 리니어레이아웃 안에 2개의 리니어레이아웃 생성
- ✓ 두 리니어레이아웃의 layout\_weight는 1:9 정도로 설정
- ✓ 위쪽 리니어레이아웃에 이미지 버튼 6개를 생성
- ✓ 위젯의 id를 다음과 같이 선언
  - 리니어레이아웃 : iconLayout, pictureLayout
  - 이미지버튼 : ibZoomin, ibZoomout, ibRotate, ibBright, ibDark, ibGray

## 2. 이미지 ▶ 이미지 활용[9/16]

### 실습 9-2 미니 포토샵 앱 만들기

#### ❖ Activity\_main.xml



## 2. 이미지 ▶ 이미지 활용[10/16]

---

### 실습 9-2 미니 포토샵 앱 만들기

#### ❖ MainActivity.java 코딩

- ✓ 이미지버튼에 대응할 6개 위젯 변수 선언
- ✓ MyGraphicView 클래스 변수를 선언
- ✓ MyGraphicView 정의 : 그림 파일을 중앙에 비트맵으로 출력
- ✓ pictureLayout을 인플레이트한 후 MyGraphicView를 추가



## 2. 이미지 ▶ 이미지 활용[11/16]

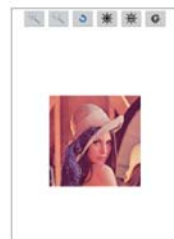
### 실습 9-2 미니 포토샵 앱 만들기

#### ❖ MainActivity.java

예제 9-12 미니 포토샵의 Java 코드 1

```
1 public class MainActivity extends AppCompatActivity {
2     ImageButton ibZoomin, ibZoomout, ibRotate, ibBright, ibDark, ibGray;
3     MyGraphicView graphicView;
4
5     @Override
6     public void onCreate(Bundle savedInstanceState) {
7         super.onCreate(savedInstanceState);
8         setContentView(R.layout.activity_main);
9         setTitle("미니 포토샵");
10
11         LinearLayout pictureLayout = (LinearLayout) findViewById(R.id.pictureLayout);
12         graphicView = (MyGraphicView) new MyGraphicView(this);
13         pictureLayout.addView(graphicView);
14
15     }
16
17     private static class MyGraphicView extends View {
18         public MyGraphicView(Context context) {
19             super(context);
20         }
21     }
```

```
21     @Override
22     protected void onDraw(Canvas canvas) {
23         super.onDraw(canvas);
24
25         Bitmap picture = BitmapFactory.decodeResource(getResources(),
26                                     R.drawable.lena256);
27         int picX = (this.getWidth() - picture.getWidth()) / 2;
28         int picY = (this.getHeight() - picture.getHeight()) / 2;
29         canvas.drawBitmap(picture, picX, picY, null);
30
31         picture.recycle();
32     }
33 }
34 }
```



## 2. 이미지 ▶ 이미지 활용[12/16]

### 실습 9-2 미니 포토샵 앱 만들기

#### ❖ 확대 아이콘 코딩

- ✓ 축척에 사용될 전역변수를 선언
- ✓ `clickIcons()` 메소드를 정의하고 확대 아이콘 클릭 리스너를 생성
- ✓ `clickIcons()` 메소드를 호출
- ✓ `onDraw()`에 `Canvas.scale()` 메소드를 추가

예제 9-13 미니 포토샵의 Java 코드 2

```
1  ----- 중간 생략 -----  
2  static float scaleX=1, scaleY=1;  
3  
4  ----- 중간 생략 -----  
5  private void clickIcons() {
```

```
6      ibZoomin = (ImageButton) findViewById(R.id.ibZoomin);  
7      ibZoomin.setOnClickListener(new View.OnClickListener() {  
8          public void onClick(View v) {  
9              scaleX = scaleX + 0.2f;  
10             scaleY = scaleY + 0.2f;  
11             graphicView.invalidate();  
12         }  
13     });  
14  
15 }  
16 ----- 중간 생략 -----  
17 clickIcons();  
18  
19 ----- 중간 생략 -----  
20 int cenX = this.getWidth() / 2 ;  
21 int cenY = this.getHeight() / 2;  
22 canvas.scale(scaleX, scaleY, cenX, cenY);  
23
```



## 2. 이미지 ▶ 이미지 활용[13/16]

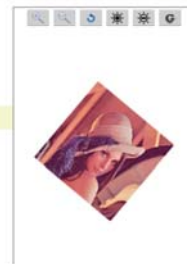
### 실습 9-2 미니 포토샵 앱 만들기

#### ❖ 회전 아이콘 코딩

- ✓ 회전에 사용될 전역변수를 선언
- ✓ 회전 아이콘 클릭 리스너를 생성
- ✓ `onDraw( )`에 `Cavas.rotate( )` 메소드를 추가

예제 9-14 미니 포토샵의 Java 코드 3

```
1  ~~~~~ 중간 생략 ~~~~~
2  static float angle=0;
3
4  ~~~~~ 중간 생략 ~~~~~
5      ibRotate = (ImageButton) findViewById(R.id.ibRotate);
6      ibRotate.setOnClickListener(new View.OnClickListener() {
7          public void onClick(View v) {
8              angle = angle + 20;
9              graphicView.invalidate();
10         }
11     });
12
13 ~~~~~ 중간 생략 ~~~~~
14 canvas.rotate(angle, cenX, cenY);
```



## 2. 이미지 ▶ 이미지 활용[14/16]

### 실습 9-2 미니 포토샵 앱 만들기

#### ❖ 밝게 하기 아이콘 코딩

- ✓ 화면 밝기에 사용될 전역변수 선언
- ✓ 밝게 하기 아이콘 클릭 리스너 생성
- ✓ `onDraw( )`에 컬러매트릭스 적용

예제 9-15 미니 포토샵의 Java 코드 4

```
1  ----- 중간 생략 -----
2  static float color=1;
3
4  ----- 중간 생략 -----
5  ibBright = (ImageButton) findViewById(R.id.ibBright);
6  ibBright.setOnClickListener(new View.OnClickListener() {
7      public void onClick(View v) {
8          color = color * 0.2f;
9          graphicView.invalidate();
10     }
11 });
12
13 ----- 중간 생략 -----
14     Paint paint = new Paint();
15     float[] array = { color, 0, 0, 0, 0,
16                     0, color, 0, 0, 0,
17                     0, 0, color, 0, 0,
18                     0, 0, 0, 1, 0 };
19     ColorMatrix cm = new ColorMatrix(array);
20     paint.setColorFilter(new ColorMatrixColorFilter(cm));
21
22 ----- 중간 생략 -----
23     canvas.drawBitmap(picture, picX, picY, paint);
```



## 2. 이미지 ▶ 이미지 활용[15/16]

### 실습 9-2 미니 포토샵 앱 만들기

#### ❖ 회색 영상으로 이미지를 변경하는 아이콘 코딩

- ✓ 채도에 사용될 전역변수 선언
- ✓ 회색 영상 아이콘 클릭 리스너 생성
- ✓ onDraw( )에 채도 설정 적용

예제 9-16 미니 포토샵의 Java 코드 5

```
1  ----- 중간 생략 -----
2  static float satur=1;
3
4  ----- 중간 생략 -----
5      ibGray = (ImageButton) findViewById(R.id.ibGray);
6      ibGray.setOnClickListener(new View.OnClickListener() {
7          public void onClick(View v) {
8              if (satur == 0) satur = 1;
9              else satur = 0;
10             graphicView.invalidate();
11         }
12     });
13
14 ----- 중간 생략 -----
15 if (satur == 0) cm.setSaturation(satur);
```



## Coding Exercise 실습 9-2 : activity\_main.xml

---

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:orientation="vertical" >

    <LinearLayout
        android:id="@+id/iconLayout"
        android:layout_width="fill_parent"
        android:layout_height="0dip"
        android:layout_weight="1"
        android:gravity="center"
        android:orientation="horizontal" >

        <ImageButton
            android:id="@+id/ibZoomin"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:src="@drawable/zoom_in" />
```

## Coding Exercise

---

```
<ImageButton
    android:id="@+id/ibZoomout"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:src="@drawable/zoom_out" />

<ImageButton
    android:id="@+id/ibRotate"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:src="@drawable/rotate" />

<ImageButton
    android:id="@+id/ibBright"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:src="@drawable/bright" />
```

## Coding Exercise

---

```
<ImageButton
    android:id="@+id/ibDark"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:src="@drawable/dark" />

<ImageButton
    android:id="@+id/ibGray"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:src="@drawable/gray" />
</LinearLayout>
```



## Coding Exercise

---

```
<LinearLayout
    android:id="@+id/pictureLayout"
    android:layout_width="fill_parent"
    android:layout_height="0dip"
    android:layout_weight="9"
    android:gravity="center"
    android:orientation="horizontal" >
</LinearLayout>

</LinearLayout>
```

## Coding Exercise 실습 9-2 : MainActivity.java

---

```
package com.cookandroid.project9_2;

import android.support.v7.app.AppCompatActivity;
import android.content.Context;
import android.graphics.Bitmap;
import android.graphics.BitmapFactory;
import android.graphics.Canvas;
import android.graphics.ColorMatrix;
import android.graphics.ColorMatrixColorFilter;
import android.graphics.Paint;
import android.os.Bundle;
import android.view.View;
import android.widget.ImageButton;
import android.widget.LinearLayout;
```

## Coding Exercise

---

```
public class MainActivity extends AppCompatActivity {  
  
    ImageButton ibZoomin, ibZoomout, ibRotate, ibBright, ibDark, ibGray;  
    MyGraphicView graphicView;  
  
    static float scaleX = 1, scaleY = 1;  
    static float angle = 0;  
    static float color = 1;  
    static float satur = 1;
```

## Coding Exercise

---

```
@Override
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    setTitle("미니 포토샵");

    LinearLayout pictureLayout = (LinearLayout) findViewById(R.id.pictureLayout);
    graphicView = (MyGraphicView) new MyGraphicView(this);
    pictureLayout.addView(graphicView);

    clickIcons();
}
```

## Coding Exercise

---

```
private void clickIcons() {  
    ibZoomin = (ImageButton) findViewById(R.id.ibZoomin);  
    ibZoomin.setOnClickListener(new View.OnClickListener() {  
        public void onClick(View v) {  
            scaleX = scaleX + 0.2f;  
            scaleY = scaleY + 0.2f;  
            graphicView.invalidate();  
        }  
    });  
  
    ibZoomout = (ImageButton) findViewById(R.id.ibZoomout);  
    ibZoomout.setOnClickListener(new View.OnClickListener() {  
        public void onClick(View v) {  
            scaleX = scaleX - 0.2f;  
            scaleY = scaleY - 0.2f;  
            graphicView.invalidate();  
        }  
    });  
}
```

## Coding Exercise

---

```
ibRotate = (ImageButton) findViewById(R.id.ibRotate);
ibRotate.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {
        angle = angle + 20;
        graphicView.invalidate();
    }
});

ibBright = (ImageButton) findViewById(R.id.ibBright);
ibBright.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {
        color = color + 0.2f;
        graphicView.invalidate();
    }
});
```

## Coding Exercise

---

```
        ibDark = (ImageButton) findViewById(R.id.ibDark);
        ibDark.setOnClickListener(new View.OnClickListener() {
            public void onClick(View v) {
                color = color - 0.2f;
                graphicView.invalidate();
            }
        });

        ibGray = (ImageButton) findViewById(R.id.ibGray);
        ibGray.setOnClickListener(new View.OnClickListener() {
            public void onClick(View v) {
                if (satur == 0)
                    satur = 1;
                else
                    satur = 0;
                graphicView.invalidate();
            }
        });
    }
}
```

## Coding Exercise

---

```
private static class MyGraphicView extends View {  
    public MyGraphicView(Context context) {  
        super(context);  
    }  
  
    @Override  
    protected void onDraw(Canvas canvas) {  
        super.onDraw(canvas);  
  
        int cenX = this.getWidth() / 2;  
        int cenY = this.getHeight() / 2;  
        canvas.scale(scaleX, scaleY, cenX, cenY);  
        canvas.rotate(angle, cenX, cenY);  
    }  
}
```



## Coding Exercise

---

```
Paint paint = new Paint();
float[] array = { color, 0, 0, 0, 0, 0, color, 0, 0, 0, 0, 0,
                 color, 0, 0, 0, 0, 0, 1, 0 };
ColorMatrix cm = new ColorMatrix(array);

if (satur == 0)
    cm.setSaturation(satur);

paint.setColorFilter(new ColorMatrixColorFilter(cm));

Bitmap picture = BitmapFactory.decodeResource(getResources(),
    R.drawable.lena256);
int picX = (this.getWidth() - picture.getWidth()) / 2;
int picY = (this.getHeight() - picture.getHeight()) / 2;
canvas.drawBitmap(picture, picX, picY, paint);

picture.recycle();
    }
}
}
```

## 2. 이미지 ▶ 이미지 활용[16/16]

### ▶ 직접 풀어보기 9-3

[실습 9-2]를 다음과 같이 수정하라.

- 회색 영상 아이콘을 없앤다.
- 밝게 하기 아이콘을 클릭하면 채도가 높아지고, 어둡게 하기 아이콘을 클릭하면 채도가 낮아지도록 코드를 수정한다.
- 블러링, 엠보싱 아이콘을 추가하고, 클릭하면 블러링 또는 엠보싱 기능이 온/오프되게 한다.



**Question?**