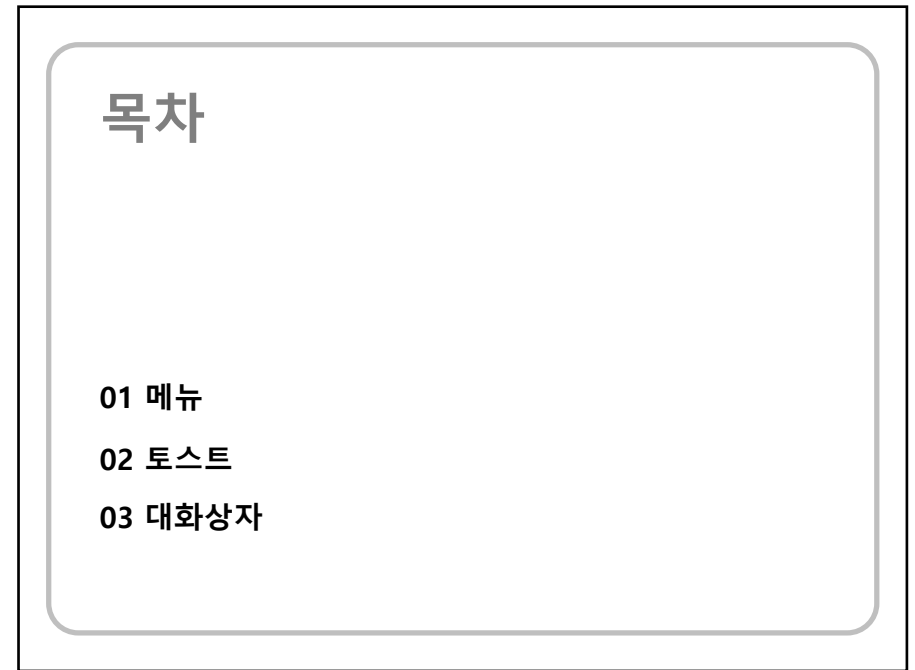
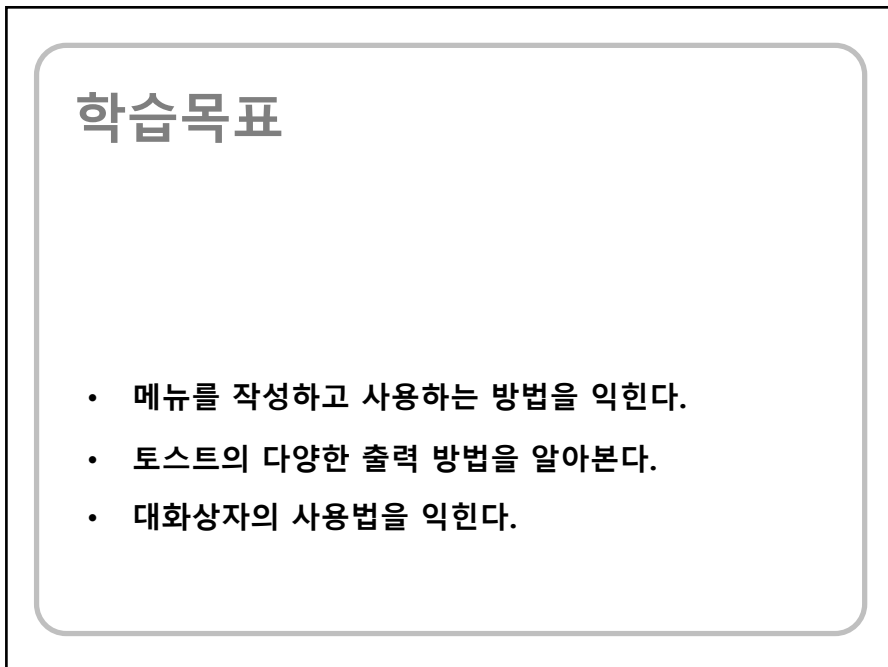




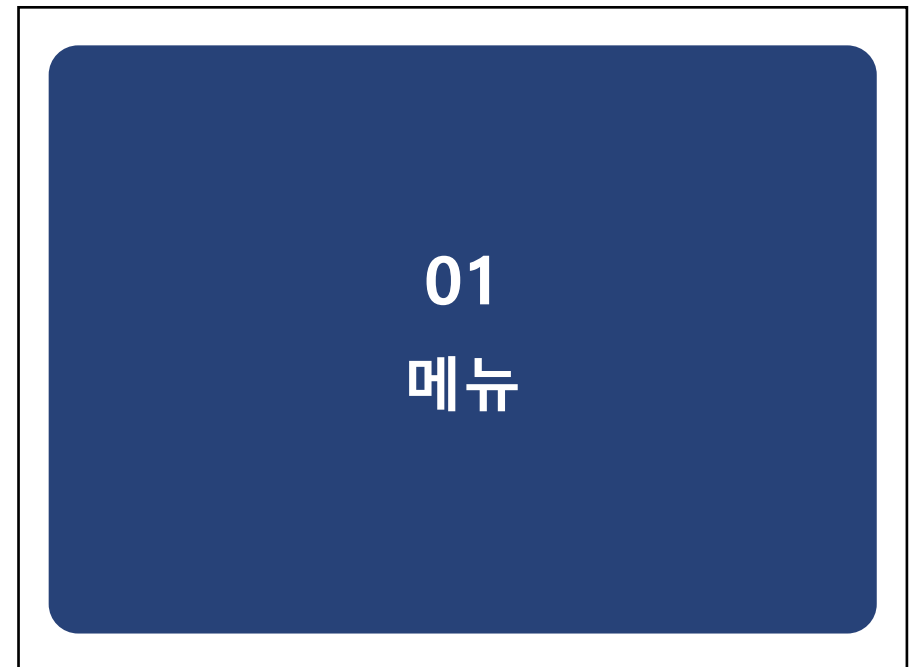
1



2



3

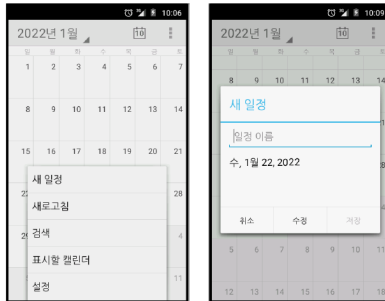


4

## 0. 메뉴

### ■ 메뉴의 종류

- 옵션 메뉴(option menu)
- 컨텍스트 메뉴(context menu)



(a) 옵션 메뉴 (b) 컨텍스트 메뉴(넙짜를 뽕글릭한 화면)

그림 7-1 옵션 메뉴와 컨텍스트 메뉴의 예

5 / 58

## 1. XML을 이용한 옵션 메뉴

### ■ 메뉴 XML 파일을 이용한 방식

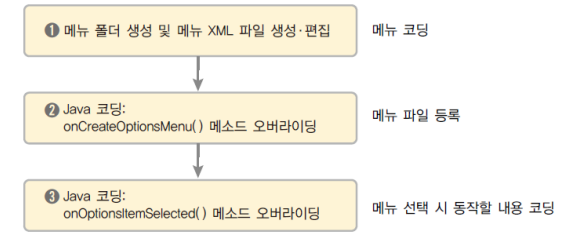


그림 7-2 옵션 메뉴 설정 순서(XML 파일 이용)

6 / 58

## 1. XML을 이용한 옵션 메뉴

### ■ 메뉴 XML 파일 형식

```
<menu>
  <item
    android:id="@id/항목1아이디"
    android:title="항목1 제목"/>
  <item
    android:id="@id/항목2아이디"
    android:title="항목2 제목"/>
</menu>
```

7 / 58

## 1. XML을 이용한 옵션 메뉴

### ■ onCreateOptionsMenu() 메소드 기본 형식

```
public boolean onCreateOptionsMenu(Menu menu) {
    super.onCreateOptionsMenu(menu);
    MenuInflater mInflater = getMenuInflater();
    mInflater.inflate(R.menu.메뉴XML아이디, menu);
    return true;
}
```

8 / 58

## 1. 메뉴 ▶ XML을 이용한 옵션 메뉴

### ■ onOptionsItemSelected() 메소드 기본 형식

- 메뉴를 선택했을 때 어떤 동작을 할 것인지 정의함

```
public boolean onOptionsItemSelected(MenuItem item) {  
    switch (item.getItemId()) {  
        case R.id.항목1아이디:  
            항목1을 선택했을 때 실행할 코드  
            return true;  
        case R.id.항목2아이디:  
            항목2를 선택했을 때 실행할 코드  
            return true;  
    }  
    return false;  
}
```

9 / 58

## 1. XML을 이용한 옵션 메뉴

### 자 만 다 인플레이터(Inflater)

인플레이터는 '부풀리는 장치' 또는 '자전거 등의 공기 펌프'라는 뜻이다. 즉 풍선에 바람을 넣어서 실제 객체로 만들어 사용한다는 의미이다(풍선은 바람을 넣기 전에는 별 의미가 없다). 안드로이드에서 사용되는 인플레이터를 이와 같이 생각하면, 정적으로 존재하는 XML 파일(풍선)을 Java 코드에서 접근하여(바람을 넣어) 실제 객체로 만들어서 사용하는 것이라고 볼 수 있다. 메뉴 인플레이터(MenuInflater) 객체는 메뉴 XML 파일을 Java 코드에서 가져와 사용하는 것이고, 레이아웃 인플레이터(LayoutInflater)는 레이아웃 XML 파일을 Java 코드에서 가져와 사용하는 것이다.

10 / 58

## 1. XML을 이용한 옵션 메뉴

### ■ 실습 7-1 배경색 바꾸기 앱 만들기

#### ■ 1 안드로이드 프로젝트 생성

- (1) 프로젝트 이름 : Project7\_1
- (2) 패키지 이름 : com.cookandroid.project7\_1

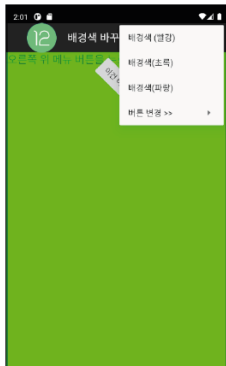


그림 7-3 배경색 바꾸기 앱 결과 화면

11 / 58

## 1. XML을 이용한 옵션 메뉴

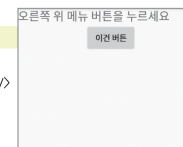
### ■ 실습 7-1 배경색 바꾸기 앱 만들기

#### ■ 2 화면 디자인 및 편집

- (1) 아래와 같이 화면 코딩
  - 바깥 리니어레이아웃의 id : baseLayout
  - 텍스트뷰 1개와 버튼 1개 생성, 버튼 id : button1

#### 예제 7-1 activity\_main.xml

```
1 <LinearLayout  
2     android:id="@+id/baseLayout" >  
3     <TextView  
4         android:text="오른쪽 위 메뉴 버튼을 누르세요" />  
5     <Button  
6         android:id="@+id/button1"  
7         android:text="이건 버튼" />  
8 </LinearLayout>
```



12 / 58

## 1. XML을 이용한 옵션 메뉴

### ■ 실습 7-1 배경색 바꾸기 앱 만들기

#### ■ 2 화면 디자인 및 편집

- (2) menu 폴더가 없다면 생성
  - [app]-[res]에서 마우스 오른쪽 버튼 클릭 - [New]-[Android Resource Directory] 선택
  - [new Resource Directory]창에서 Resource type으로 'menu' 선택하고 <OK> 클릭

13 / 58

## 1. XML을 이용한 옵션 메뉴

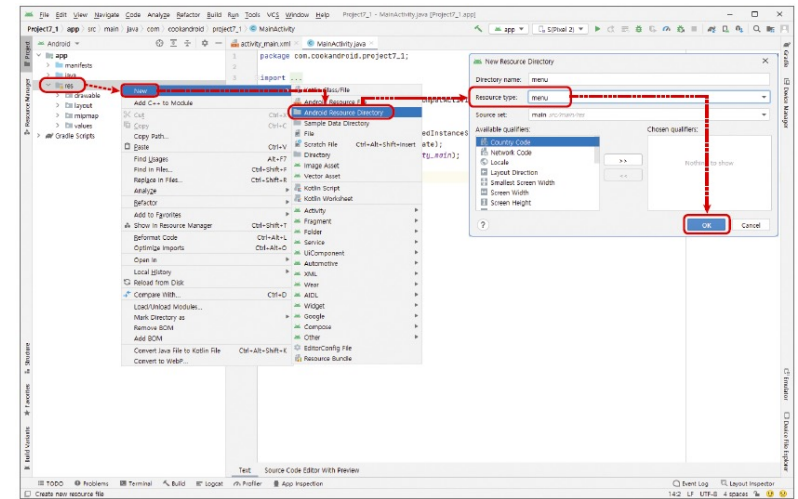


그림 7-4 메뉴 폴더 생성

14 / 58

## 1. XML을 이용한 옵션 메뉴

### ■ 실습 7-1 배경색 바꾸기 앱 만들기

#### ■ 2 화면 디자인 및 편집

- (3) 메뉴 XML 파일 만들기
  - [app]-[res]-[menu]에서 마우스 오른쪽 버튼 클릭 - [New]-[Menu resource file] 선택
  - [New Resource File] 창에서 새로운 파일 이름 'menu1.xml'을 입력하고 <OK> 클릭

15 / 58

## 1. XML을 이용한 옵션 메뉴

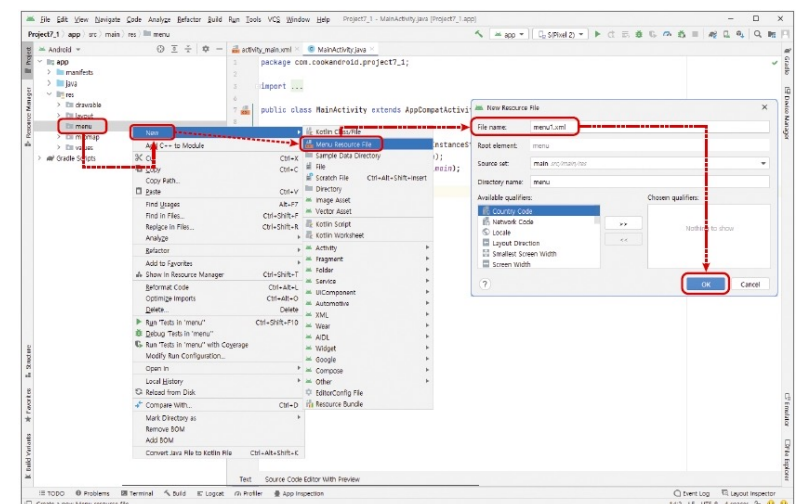


그림 7-5 메뉴 XML 파일 생성

16 / 58

## 1. XML을 이용한 옵션 메뉴

### ■ 실습 7-1 배경색 바꾸기 앱 만들기

#### ■ 2 화면 디자인 및 편집

##### ■ (4) 메뉴 XML 파일 코딩

예제 7-2 menu1.xml

```
1 <menu xmlns:android="http://schemas.android.com/apk/res/android" >
2
3     <item
4         android:id="@id/itemRed"
5         android:title="배경색(빨강)">
6     </item>
7     <item
8         android:id="@id/itemGreen"
9         android:title="배경색(초록)">
10    </item>
11    <item
12        android:id="@id/itemBlue"
13        android:title="배경색(파랑)">
14    </item>
15    <item android:title="버튼 변경 >>" >
16    <menu>
17        <item
18            android:id="@id/subRotate"
19            android:title="버튼 45도 회전">
20        <item
21            android:id="@id/subSize"
22            android:title="버튼 2배 확대">
23        </menu>
24    </item>
25
26 </menu>
```

17 / 58

## 1. XML을 이용한 옵션 메뉴

### ■ 실습 7-1 배경색 바꾸기 앱 만들기

#### ■ Java 코드 작성 및 수정

##### ■ (1) [MainActivity]에 다음 내용을 코딩

- activity\_main.xml의 레이아웃과 버튼에 대응할 전역변수 2개
- 메인 함수 onCreate() 안에서 위젯 변수 2개에 위젯 대입

예제 7-3 Java 코드 1

```
1 ----- 중간 생략(import문) -----
2 public class MainActivity extends AppCompatActivity {
3     LinearLayout baseLayout;
4     Button button1;
5
6     @Override
7     public void onCreate(Bundle savedInstanceState) {
8         super.onCreate(savedInstanceState);
9         setContentView(R.layout.activity_main);
10        setTitle("배경색 바꾸기");
11        baseLayout = (LinearLayout) findViewById(R.id.baseLayout);
12        button1 = (Button) findViewById(R.id.button1);
13    }
14
15 }
```

18 / 58

## 1. XML을 이용한 옵션 메뉴

### ■ 실습 7-1 배경색 바꾸기 앱 만들기

#### ■ Java 코드 작성 및 수정

##### ■ (2) onCreateOptionsMenu()를 Activity 클래스로부터 오버라이딩

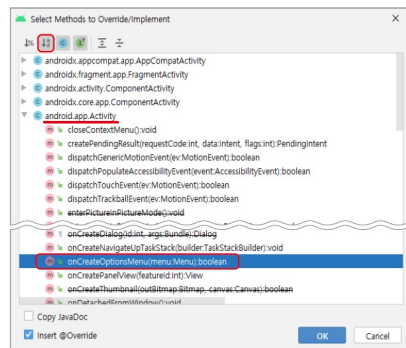


그림 7-6 onCreateOptionsMenu() 메소드 오버라이딩

19 / 58

## 1. XML을 이용한 옵션 메뉴

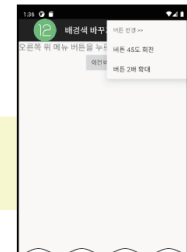
### ■ 실습 7-1 배경색 바꾸기 앱 만들기

#### ■ Java 코드 작성 및 수정

##### ■ (3) 자동 완성된 코드에 나머지 코딩

예제 7-4 Java 코드 2

```
1 ----- 생략 -----
2     baseLayout = (LinearLayout) findViewById(R.id.baseLayout);
3     button1 = (Button) findViewById(R.id.button1);
4
5
6     @Override
7     public boolean onCreateOptionsMenu(Menu menu) {
8         super.onCreateOptionsMenu(menu);
9         MenuInflater mInflater = getMenuInflater();
10        mInflater.inflate(R.menu.menu1, menu);
11        return true;
12    }
13
14 }
```



20 / 58

## 1. XML을 이용한 옵션 메뉴

### ■ 실습 7-1 배경색 바꾸기 앱 만들기

#### ■ Java 코드 작성 및 수정

- (4) 메뉴를 클릭했을 때 동작할 메소드 onOptionsItemSelected() 코딩

예제 7-5 Java 코드 3

```

1  ----- 중간 생략 -----
2  mInflater.inflate(R.menu.menu1, menu);
3  return true;
4  }
5
6  @Override
7  public boolean onOptionsItemSelected(MenuItem item) {
8      switch (item.getItemId()) {
9          case R.id.itemRed:
10             baseLayout.setBackgroundColor(Color.RED);
11             return true;
12         ----- 중간 생략(itemGreen, itemBlue의 case문) -----
13         case R.id.subRotate:
14             button1.setRotation(45);
15             return true;
16         case R.id.subSize:
17             button1.setScaleX(2);
18             return true;
19     }
20     return false;
21 }
22

```

21 / 58

## 2. Java 코드만 이용한 옵션 메뉴

### ■ XML 파일 없이 Java 코드로만 옵션 메뉴 사용하기

- onCreateOptionsMenu() 메소드 안에서 메뉴 XML 파일에 접근하는 대신 직접 menu.add() 메소드로 메뉴 항목 추가

예제 7-6 Java 코드로만 구성된 옵션 메뉴

```

1  @Override
2  public boolean onCreateOptionsMenu(Menu menu) {
3      super.onCreateOptionsMenu(menu);
4
5      menu.add(0, 1, 0, "배경색(빨강)");
6      menu.add(0, 2, 0, "배경색(초록)");
7      menu.add(0, 3, 0, "배경색(파랑)");
8
9      SubMenu sMenu = menu.addSubMenu("버튼 변경 >>");
10     sMenu.add(0, 4, 0, "버튼 45도 회전");
11     sMenu.add(0, 5, 0, "버튼 2배 확대");
12
13     return true;
14 }
15
16 @Override
17 public boolean onOptionsItemSelected(MenuItem item) {
18     switch (item.getItemId()) {
19     case 1:
20         baseLayout.setBackgroundColor(Color.RED);
21         return true;
22         ----- 중간 생략(case별 변경) -----

```

22 / 58

21

22

## 2. Java 코드만 이용한 옵션 메뉴

### ▶ 직접 알아보기 7-1

[실습 7-1]을 다음과 같이 수정하라.

- 레이아웃은 RelativeLayout을 사용하고 텍스트뷰, 에디트텍스트, 이미지뷰를 적절히 배치한다.
- 에디트텍스트에 각도를 입력하고 옵션 메뉴의 [그림 회전]을 선택하면 해당 각도만큼 이미지뷰가 회전한다.
- 한라산, 추자도, 범섬 옵션 메뉴는 라디오버튼과 같이 3개 중 하나만 체크되고, 메뉴를 선택하면 이미지뷰가 해당 이미지로 바뀐다(이미지나 이미지 이름이 달라도 된다).

※ 메뉴 XML에서 여러 항목을 묶으려면 <group>~</group>을 이용한다. 예를 들면 다음과 같다.

```

<group
    android:checkableBehavior="single">
    <item
        android:id="@+id/item1"
        android:title="한라산"
        android:checked="true">
    </item>
    ----- 생략 -----
</group>

```

※ 이미지뷰의 이미지를 변경하려면 setImageResource() 메소드를 사용한다.

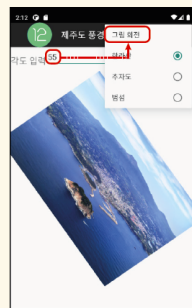


그림 7-7 제주도 풍경 앱

23 / 58

## 3. XML을 이용한 컨텍스트 메뉴

### ■ 컨텍스트 메뉴(Context Menu)

- 레이아웃 또는 버튼, 에디트텍스트 등의 위젯을 롱클릭하면 나타남
- Windows의 팝업창과 비슷함

### ■ 컨텍스트 메뉴 설정 순서

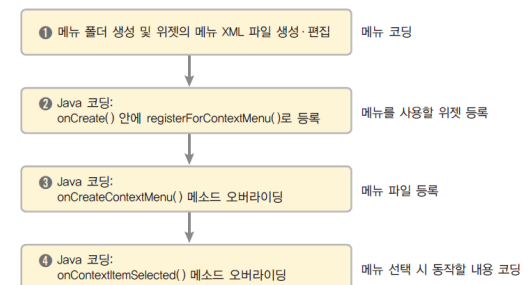


그림 7-8 컨텍스트 메뉴 설정 순서(XML 파일 이용)

24 / 58

23

24

### 3. XML을 이용한 컨텍스트 메뉴

#### ■ 컨텍스트 메뉴 XML 파일

- 컨텍스트 메뉴가 나오게 할 위젯마다 별도의 파일 생성해야 함
- 메뉴 XML 문법은 옵션 메뉴와 동일함
- onCreateContextMenu() 메소드에는 위젯별로 컨텍스트 메뉴가 나타나야 하므로, 위젯별 컨텍스트 메뉴를 if문으로 등록해야 함

```
public void onCreateContextMenu(ContextMenu menu,
    View v, ContextMenuInfo menuInfo) {
    super.onCreateContextMenu(menu, v, menuInfo);

    MenuInflater mInflater = getMenuInflater();
    if(v == 위젯1) {
        mInflater.inflate(R.menu.첫번째메뉴XML파일, menu);
    }
    if (v == 위젯2) {
        mInflater.inflate(R.menu.두번째메뉴XML파일, menu);
    }
}
```

25 / 58

### 3. XML을 이용한 컨텍스트 메뉴

#### ■ 실습 7-2 배경색 변경 및 버튼 변경 앱 만들기

#### ■ 1 안드로이드 프로젝트 생성

- (1) 프로젝트 이름 : Project7\_2
- (2) 패키지 이름 : com.cookandroid.project7\_2

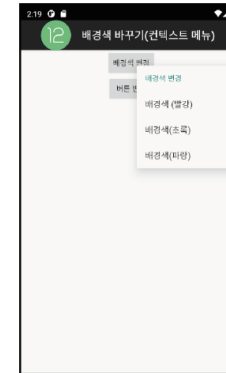


그림 7-9 배경색 변경 및 버튼 변경 앱 결과 화면

26 / 58

### 3. XML을 이용한 컨텍스트 메뉴

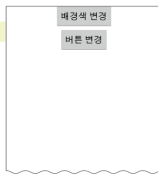
#### ■ 실습 7-2 배경색 변경 및 버튼 변경 앱 만들기

#### ■ 2 화면 디자인 및 편집

- (1) 화면을 간단히 코딩하기
  - 레이아웃의 id : baseLayout
  - 버튼 2개 생성, 버튼의 id : button1, button2

예제 7-7 activity\_main.xml

```
1 <LinearLayout
2   android:gravity="center_horizontal"
3   >
4   <Button
5     android:id="@+id/button1"
6     android:text="배경색 변경" />
7   <Button
8     android:id="@+id/button2"
9     android:text="버튼 변경" />
10 </LinearLayout>
```



27 / 58

### 3. XML을 이용한 컨텍스트 메뉴

#### ■ 실습 7-2 배경색 변경 및 버튼 변경 앱 만들기

#### ■ 2 화면 디자인 및 편집

- (2) [실습 7-1]을 참고하여 menu폴더와 menu1.xml, menu2.xml 파일 생성
- (3) menu1.xml : 배경색 변경과 관련된 3개의 항목 생성
- (4) menu2.xml : 버튼과 관련된 2개의 항목 생성

예제 7-8 menu1.xml

```
1 <menu xmlns:android="http://schemas.android.com/apk/res/android" >
2   <item
3     android:id="@+id/itemRed"
4     android:title="배경색(빨강)" />
5   </item>
6   ~~~~ 중간 생략(배경색 항목 2개) ~~~~
7 </menu>
```

예제 7-9 menu2.xml

```
1 <menu xmlns:android="http://schemas.android.com/apk/res/android" >
2   <item
3     android:id="@+id/subRotate"
4     android:title="버튼 45도 회전" />
5   <item
6     android:id="@+id/subSize"
7     android:title="버튼 2배 확대" />
8 </menu>
```

28 / 58

### 3. XML을 이용한 컨텍스트 메뉴

#### ■ 실습 7-2 배경색 변경 및 버튼 변경 앱 만들기

#### ■ 3 Java 코드 작성 및 수정

- (1) MainActivity에 다음 내용을 코딩
  - activity\_main.xml의 레이아웃과 버튼 2개에 대응할 전역변수 3개
  - 메인 함수 onCreate() 안에서 위젯 변수 3개에 위젯 대입
  - 2개의 버튼 위젯 변수를 registerForContextMenu()에 등록

29 / 58

### 3. XML을 이용한 컨텍스트 메뉴

예제 7-10 Java 코드 1

```
1  ~~~~ 중간 생략(import문) ~~~~
2  public class MainActivity extends Activity {
3      LinearLayout baseLayout;
4      Button button1, button2;
5
6      @Override
7      public void onCreate(Bundle savedInstanceState) {
8          super.onCreate(savedInstanceState);
9          setContentView(R.layout.activity_main);
10         setTitle("배경색 바꾸기(컨텍스트 메뉴)");
11         baseLayout = (LinearLayout) findViewById(R.id.baseLayout);
12         button1 = (Button) findViewById(R.id.button1);
13         registerForContextMenu(button1);
14         button2 = (Button) findViewById(R.id.button2);
15         registerForContextMenu(button2);
16     }
17
18 }
```

30 / 58

### 3. XML을 이용한 컨텍스트 메뉴

#### ■ 실습 7-2 배경색 변경 및 버튼 변경 앱 만들기

#### ■ Java 코드 작성 및 수정

- (2) 선택한 위젯에 따라 인플레이트하는 메뉴 XML 파일을 다르게 함
- (3) onOptionsItemSelected()를 같은 방식으로 자동 완성한 후 코딩

예제 7-11 Java 코드 2

```
1  @Override
2  public void onCreateContextMenu(ContextMenu menu, View v,
3      ContextMenuInfo menuInfo) {
4      // TODO: Auto-generated method stub
5      super.onCreateContextMenu(menu, v, menuInfo);
6
7      MenuInflater mInflater = getMenuInflater();
8      if (v == button1) {
9          menu.setHeaderTitle("배경색 변경");
10
11          mInflater.inflate(R.menu.menu1, menu);
12      }
13      if (v == button2) {
14          mInflater.inflate(R.menu.menu2, menu);
15      }
16 }
```

31 / 58

### 3. XML을 이용한 컨텍스트 메뉴

#### ▶ 직접 알아보기 7-2

[실습 7-2]에서 실습한 컨텍스트 메뉴를 메뉴 XML 파일 없이 Java 코드로만 완성하라. [예제 7-6]을 참조한다.

32 / 58



## 02 토스트

### 1. 토스트

#### ■ 토스트(toast)

- 화면에 잠깐 나타났다 사라지는 메시지
- 프로그래머가 디버깅 용도로 사용하기도 적당함

#### ■ 토스트의 일반적인 형태

```
Toast.makeText(Context context, String message, int duration).show();
```

- duration: 화면에 나타나는 시간

- setGravity( ) 메소드를 사용하면 위치를 변경할 수 있음

```
Toast.setGravity(int gravity, int xOffset, int yOffset);
```

- gravity: 화면의 위, 중앙, 아래 등을 지정
- xOffset, yOffset: 떨어진 거리

34 / 58

34

### 1. 토스트

#### ■ 토스트 메시지가 랜덤 위치에 출력되는 예제

예제 7-12 토스트 연습용 activity\_main.xml 코드

```
1 <LinearLayout
2     android:gravity="center" >
3
4     <Button
5         android:id="@+id/button1"
6         android:text="메시지 출력" />
7
8 </LinearLayout>
```



35 / 58

### 1. 토스트

예제 7-13 토스트 연습용 Java 코드

```
1 public void onCreate(Bundle savedInstanceState) {
2     super.onCreate(savedInstanceState);
3     setContentView(R.layout.activity_main);
4     setTitle("토스트 연습");
5
6     final Button button1 = (Button) findViewById(R.id.button1);
7
8     button1.setOnClickListener(new View.OnClickListener() {
9         public void onClick(View v) {
10             Toast tMsg = Toast.makeText(액티비티명.this, "토스트 연습",
11                                     Toast.LENGTH_SHORT);
12
13             Display display = ((WindowManager) getSystemService(WINDOW_SERVICE))
14                             .getDefaultDisplay();
15             int xOffset = (int) (Math.random() * display.getWidth());
16             int yOffset = (int) (Math.random() * display.getHeight());
17
18             tMsg.setGravity(Gravity.TOP | Gravity.LEFT, xOffset, yOffset);
19             tMsg.show();
20         }
21     });
22 }
```

36 / 58

36

# 03 대화상자

## 1. 기본 대화상자

### ■ 대화상자(Dialog)

- 사용자에게 중요한 사항을 알려준 후, 어떤 선택을 하게 하는 것이 목적

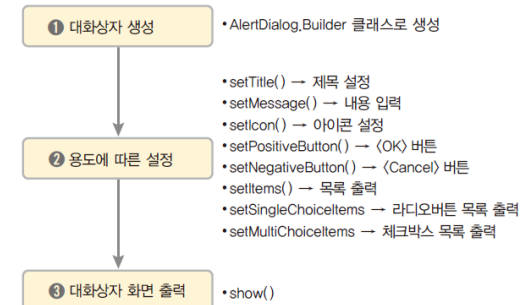


그림 7-10 대화상자 설정 순서

38 / 58

38

## 1. 기본 대화상자

### ■ 대화상자 연습을 위한 버튼 하나만 있는 예제

예제 7-14 대화상자 연습용 activity\_main.xml 코드

```

1 <LinearLayout
2     android:gravity="center_horizontal" >
3     <Button
4         android:id="@+id/button1"
5         android:text="대화상자" />
6 </LinearLayout>
  
```

39 / 58

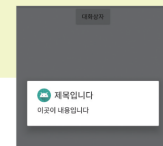
## 1. 기본 대화상자

### ■ 대화상자 Java 코드(버튼 없음)

예제 7-15 기본적인 대화상자의 Java 코드

```

1 public void onCreate(Bundle savedInstanceState) {
2     super.onCreate(savedInstanceState);
3     setContentView(R.layout.activity_main);
4     final Button button1 = (Button) findViewById(R.id.button1);
5     button1.setOnClickListener(new View.OnClickListener() {
6
7         public void onClick(View v) {
8             AlertDialog.Builder dlg = new AlertDialog.Builder(맥티버티명.this);
9             dlg.setTitle("제목입니다");
10            dlg.setMessage("이곳이 내용입니다");
11            dlg.setIcon(R.mipmap.ic_launcher);
12            dlg.show();
13        }
14    });
15 }
16 }
  
```



40 / 58

40

## 1. 기본 대화상자

### ■ 대화상자 Java 코드(버튼 1개)

예제 7-16 버튼이 1개인 대화상자의 Java 코드

```
1  ~~~~ 중간 생략 ~~~~
2  public void onClick(View v) {
3      AlertDialog.Builder dlg = new AlertDialog.Builder(액티비티명.this);
4      dlg.setTitle("제목입니다");
5      dlg.setMessage("여기에 내용입니다");
6      dlg.setIcon(R.mipmap.ic_launcher);
7      dlg.setPositiveButton("확인", null);
8      dlg.show();
9  }
10 ~~~~ 중간 생략 ~~~~
```



41 / 58

## 1. 기본 대화상자

### ■ 버튼 클릭 시 동작하는 대화상자의 Java 코드

- `setPositiveButton("문자열", 리스너)`의 리스너가 null이라 아무 동작도 안 하므로 해당 부분 수정

예제 7-17 버튼 클릭 시 동작하는 대화상자의 Java 코드

```
1  dlg.setPositiveButton("확인", new DialogInterface.OnClickListener() {
2      public void onClick(DialogInterface dialog, int which) {
3          Toast.makeText(액티비티명.this, "확인을 눌렀네요", Toast.LENGTH_SHORT).show();
4      }
5  });
```

42 / 58

## 2. 목록 대화상자

### ■ 가장 기본적인 목록을 만드는 형식

예제 7-18 기본적인 목록 대화상자의 Java 코드

```
1  ~~~~ 생략 ~~~~
2  public void onClick(View v) {
3
4      final String[] versionArray = new String[] { "Q(10)", "R(11)", "S(12) "};
5      AlertDialog.Builder dlg = new AlertDialog.Builder(액티비티명.this);
6      dlg.setTitle("좋아하는 버전은?");
7      dlg.setIcon(R.mipmap.ic_launcher);
8      dlg.setItems(versionArray,
9          new DialogInterface.OnClickListener() {
10             public void onClick(DialogInterface dialog, int which) {
11                 button1.setText(versionArray[which]);
12             }
13         });
14      dlg.setPositiveButton("닫기", null);
15      dlg.show();
16  }
17 ~~~~ 생략 ~~~~
```



43 / 58

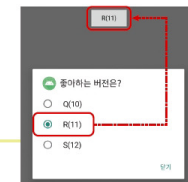
## 2. 목록 대화상자

### ■ 라디오버튼 목록 대화상자 Java 코드

- `setItems( )` 대신 `setSingleChoiceItems( )` 사용하면 라디오버튼 형태로 출력
- `setSingleChoiceItems(문자열 배열, 초기 선택 인덱스, 리스너)`로 파라미터가 3개임

예제 7-19 라디오버튼 목록 대화상자의 Java 코드

```
1  dlg.setSingleChoiceItems ( versionArray , 0,
2      new DialogInterface.OnClickListener() {
3          public void onClick(DialogInterface dialog, int which) {
4              button1.setText(versionArray[which]);
5          }
6      });
```



44 / 58

## 2. 목록 대화상자

### ■ 체크박스 목록 대화상자 Java 코드

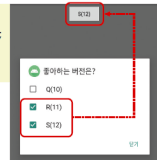
- 여러 개를 동시에 선택하려면 setMultiChoiceItems()를 사용

예제 7-20 체크박스 목록 대화상자의 Java 코드

```

1  ----- 생략 -----
2  public void onClick(View v) {
3
4      final String[] versionArray = new String[] { "Q(10)", "R(11)", "S(12) "};
5      final boolean[] checkArray = new boolean[] {true, false, false};
6      AlertDialog.Builder dlg = new AlertDialog.Builder(액티비티명.this);
7      dlg.setTitle("좋아하는 버전은?");
8      dlg.setIcon(R.mipmap.ic_launcher);
9      dlg.setMultiChoiceItems(versionArray, checkArray,
10         new DialogInterface.OnMultiChoiceClickListener() {
11             public void onClick(DialogInterface dialog,
12
13                 int which, boolean isChecked) {
14                     button1.setText(versionArray[which]);
15                 }
16             });
17     dlg.setPositiveButton("닫기", null);
18     dlg.show();
19 }
----- 생략 -----

```



45 / 58

45

## 2. 목록 대화상자

### ■ 실습 7-3 사용자 정보 입력 앱 만들기

#### ■ 1 안드로이드 프로젝트 생성

- (1) 프로젝트 이름 : Project7\_3
- (2) 패키지 이름 : com.cookandroid.project7\_3



그림 7-11 사용자 정보 입력 앱 결과 화면

46 / 58

46

## 2. 목록 대화상자

### ■ 실습 7-3 사용자 정보 입력 앱 만들기

#### ■ 2 화면 디자인 및 편집

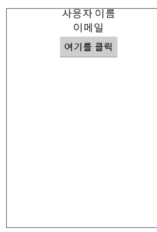
- (1) 아래와 같이 화면 코딩하기
  - 텍스트뷰 2개와 버튼 1개를 생성, 위젯의 id : tvName, tvEmail, button1

예제 7-21 activity\_main.xml

```

1  <LinearLayout
2      android:gravity="center_horizontal" >
3      <TextView
4          android:id="@+id/tvName"
5          android:text="사용자 이름" />
6      <TextView
7          android:id="@+id/tvEmail"
8          android:text="이메일" />
9      <Button
10         android:id="@+id/button1" />
11 </LinearLayout>

```



47 / 58

47

## 2. 목록 대화상자

### ■ 실습 7-3 사용자 정보 입력 앱 만들기

#### ■ 2 화면 디자인 및 편집

- (2) 대화상자에서 사용할 레이아웃 XML 파일 생성
  - [app]-[res]-[layout]에서 마우스 오른쪽 버튼을 클릭하고 [New]-[Layout Resource File] 선택
  - [New Resource File] 창에서 File name에는 'dialog1.xml'을, Root element에는 'LinearLayout'을 입력하고 <OK>를 클릭

48 / 58

48

## 2. 목록 대화상자

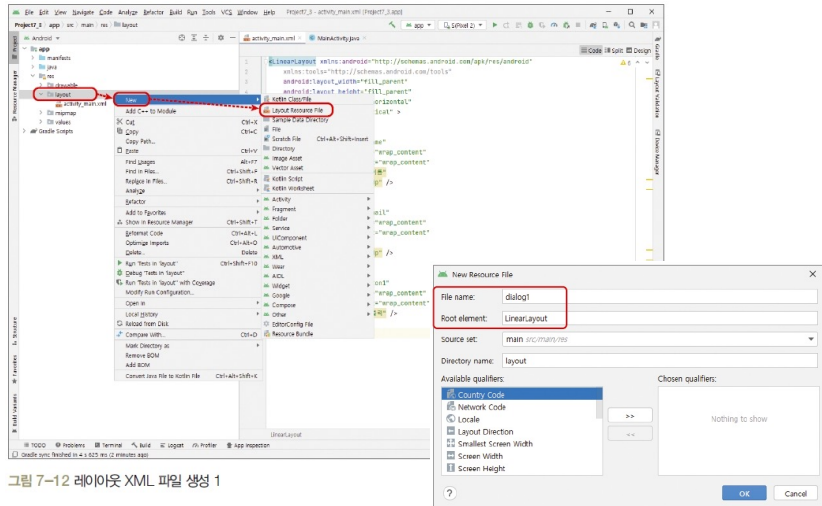


그림 7-12 레이아웃 XML 파일 생성 1

그림 7-13 레이아웃 XML 파일 생성 2

49 / 58

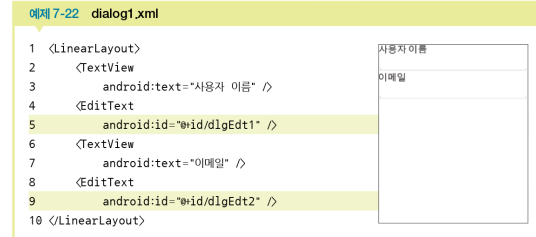
## 2. 목록 대화상자

### ■ 실습 7-3 사용자 정보 입력 앱 만들기

#### ■ 2 화면 디자인 및 편집

##### ■ (3) 대화상자용 dialog1.xml 파일을 코딩

- 텍스트뷰, 에디트텍스트, 텍스트뷰, 에디트텍스트의 순서로 생성
- 에디트텍스트만 위젯 id를 dlgEdt1, dlgEdt2로 지정



50 / 58

## 2. 목록 대화상자

### ■ 실습 7-3 사용자 정보 입력 앱 만들기

#### ■ 2 화면 디자인 및 편집

##### ■ (4) toast1.xml 파일을 생성하고 코딩

- 이미지뷰, 텍스트뷰, 이미지뷰의 순서로 생성
- 텍스트뷰의 id를 toastText1로 지정
- 레이아웃의 배경을 빨간색으로 지정



51 / 58

## 2. 목록 대화상자

### ■ 실습 7-3 사용자 정보 입력 앱 만들기

#### ■ 3 Java 코드 작성 및 수정

##### ■ (1) MainActivity에 다음 내용 코딩하기

- activity\_main.xml의 텍스트뷰 2개, 버튼 1개에 대응할 전역변수 3개
- dialog1.xml의 에디트텍스트에 대응할 전역변수 2개
- toast1.xml의 텍스트뷰 1개에 대응할 전역변수 1개
- dialog1.xml과 toast1.xml을 인플레이트할 뷰 변수 2개
- 메인 함수 onCreate() 안에서 activity\_main.xml의 위젯 변수 3개에 위젯 대입

52 / 58

## 2. 목록 대화상자

예제 7-24 Java 코드 1

```
1  ~~~~ 생략(import문) ~~~~
2  public class MainActivity extends Activity {
3      TextView tvName, tvEmail;
4      Button button1;
5      EditText dlgEdtName, dlgEdtEmail;
6      TextView toastText;
7      View dialogView, toastView;
8
9      @Override
10     public void onCreate(Bundle savedInstanceState) {
11         super.onCreate(savedInstanceState);
12         setContentView(R.layout.activity_main);
13         setTitle("사용자 정보 입력");
14
15         tvName = (TextView) findViewById(R.id.tvName);
16         tvEmail = (TextView) findViewById(R.id.tvEmail);
17         button1 = (Button) findViewById(R.id.button1);
18     }
19 }
20 }
```

53 / 58

## 2. 목록 대화상자

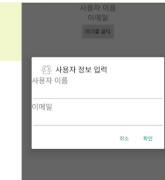
### ■ 실습 7-3 사용자 정보 입력 앱 만들기

#### ■ 3 Java 코드 작성 및 수정

- (2) button1 변수를 클릭했을 때 리스너를 작성
  - onClick() 메소드 안에 대화상자를 만들고 <확인>과 <취소>를 추가
  - <확인>, <취소> 버튼 클릭 시 실행할 내용은 일단 null로 입력

예제 7-25 Java 코드 2

```
1  button1.setOnClickListener(new View.OnClickListener() {
2      public void onClick(View v) {
3          dialogView = (View) View.inflate(MainActivity.this,
4              R.layout.dialog1, null);
5          AlertDialog.Builder dlg = new AlertDialog.Builder(MainActivity.this);
6          dlg.setTitle("사용자 정보 입력");
7          dlg.setIcon(R.drawable.ic_menu_allfriends);
8          dlg.setView(dialogView);
9          dlg.setPositiveButton("확인", null);
10         dlg.setNegativeButton("취소", null);
11         dlg.show();
12     }
13 });
```



54 / 58

## 2. 목록 대화상자

### ■ 실습 7-3 사용자 정보 입력 앱 만들기

#### ■ 3 Java 코드 작성 및 수정

- (3) 대화상자의 <확인>을 클릭하면 대화상자에서 입력한 사용자 이름과 이메일이 메인 화면(activity\_main.xml)의 텍스트뷰에 쓰이도록 코딩

예제 7-26 Java 코드 3

```
1  dlg.setPositiveButton("확인",
2      new DialogInterface.OnClickListener() {
3      public void onClick(DialogInterface dialog, int which) {
4          dlgEdtName = (EditText) dialogView.findViewById(R.id.dlgEdt1);
5          dlgEdtEmail = (EditText) dialogView.findViewById(R.id.dlgEdt2);
6
7          tvName.setText(dlgEdtName.getText().toString());
8          tvEmail.setText(dlgEdtEmail.getText().toString());
9      }
10 });
```

55 / 58

## 2. 목록 대화상자

### ■ 실습 7-3 사용자 정보 입력 앱 만들기

#### ■ 3 Java 코드 작성 및 수정

- (4) 대화상자의 <취소>를 클릭했을 때 toast1.xml이 토스트 메시지로 나오도록 설정

예제 7-27 Java 코드 4

```
1  dlg.setNegativeButton("취소",
2      new DialogInterface.OnClickListener() {
3      public void onClick(DialogInterface dialog, int which) {
4          Toast toast = new Toast(MainActivity.this);
5          toastView = (View) View.inflate(MainActivity.this,
6              R.layout.toast1, null);
7          toastText = (TextView) toastView.findViewById(R.id.toastText1);
8          toastText.setText("취소했습니다.");
9          toast.setView(toastView);
10         toast.show();
11     }
12 });
```



56 / 58

## 2. 목록 대화상자

### ▶ 작업 풀어보기 7-3

[실습 7-3]을 다음과 같이 수정하라.

- activity\_main.xml의 텍스트뷰를 에디트텍스트로 변경한다.
- <여기를 클릭>을 클릭하면 activity\_main.xml의 에디트텍스트 내용이 대화상자의 에디트텍스트에 나타난다.
- 대화상자에서 <확인>을 클릭하면 대화상자의 에디트텍스트 내용이 activity\_main.xml의 에디트텍스트 내용으로 변경된다.
- 대화상자에서 <취소>를 클릭하면 토스트가 화면의 임의 위치에 나타난다.

그림 7-14 수정된 사용자 정보 입력 앱



57 / 58

57

# Thank You !

 **한빛아카데미**  
Hanbit Academy, Inc.

Copyright © 2022 Hanbit Academy, Inc.  
All rights reserved.

58