

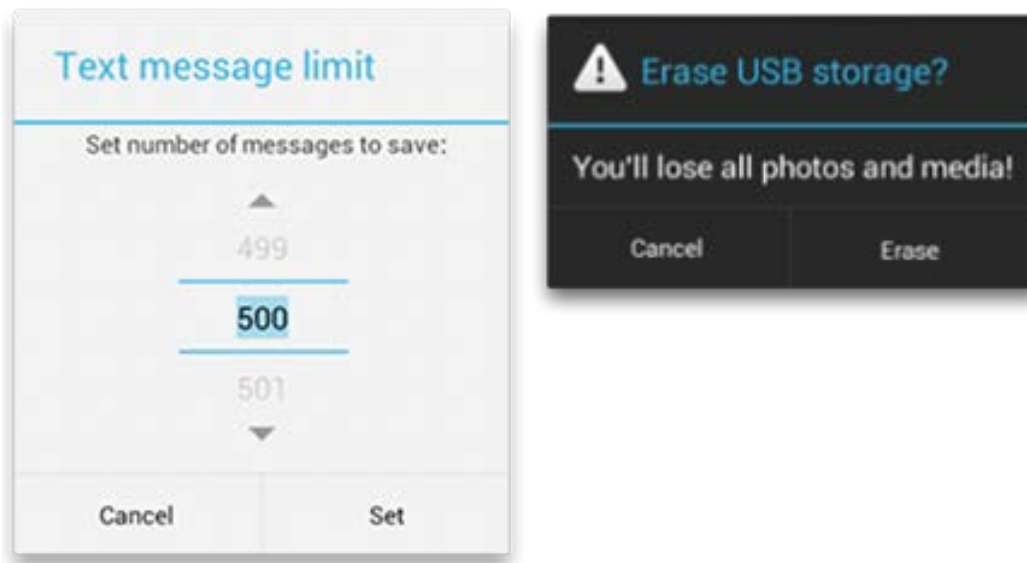


CHAP 7. 메뉴와 대화상자(2)

1

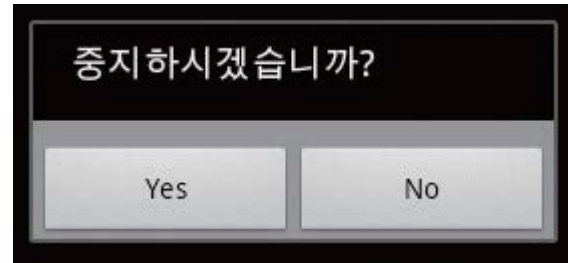
대화 상자

- 대화 상자(dialog)는 사용자에게 메시지를 출력하고 사용자로부터 입력을 받아들이는 아주 보편적인 사용자 인터페이스



대화 상자의 종류

- AlertDialog



- DatePickerDialog

- TimePickerDialog



대화 상자 생성, 표시, 제거 메카니즘

```
public class DialogTestActivity extends ActionBarActivity {
```

```
...
```

```
protected Dialog onCreateDialog(int id) {
```

```
    switch (id) {
```

```
        case DIALOG_PAUSED_ID:
```

```
            return new AlertDialog.Builder(DialogTest.this).create();
```

```
            ...
```

```
        }
```

```
    return null;
```

```
}
```

```
...
```

```
showDialog(DIALOG_PAUSED_ID);
```

```
...
```

```
dismissDialog(DIALOG_PAUSED_ID);
```

```
...
```

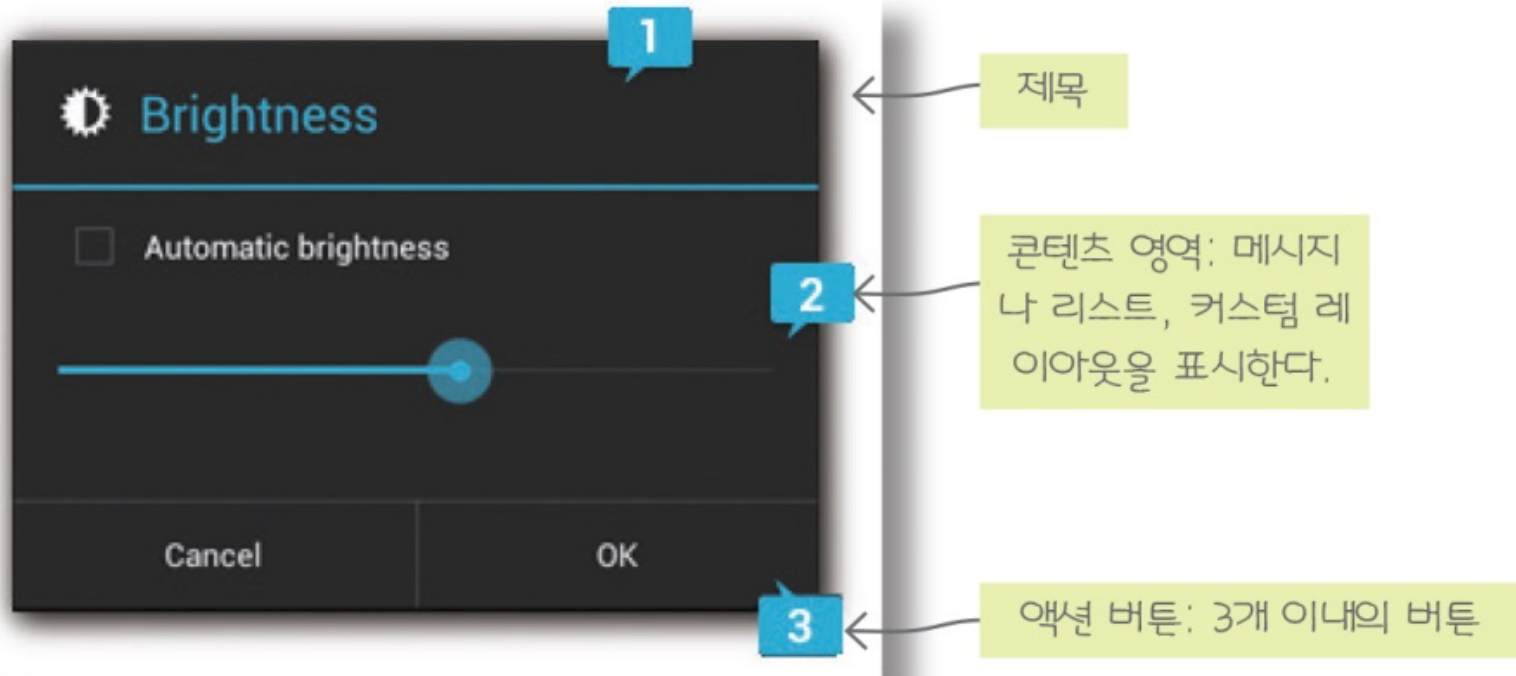
```
}
```

각 식별자에 해당하는
대화 상자를 생성한다.

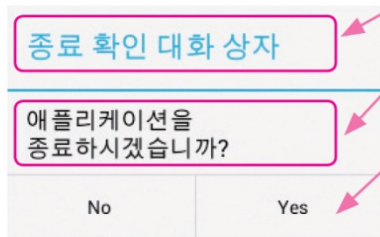
대화 상자가 필요할 때
마다 호출하여 대화 상
자를 화면에 표시한다.

대화 상자를 화면에서
제거한다.

ALERTDIALOG



ALERTDIALOG



```
AlertDialog.Builder builder = new AlertDialog.Builder(this);  
builder.setTitle("종료 확인 대화 상자")           // 제목  
    .setMessage("애플리케이션을 종료하시겠습니까?") // 메시지  
    .setCancelable(false)  
    .setPositiveButton("Yes",                     // "Yes" 버튼  
        new DialogInterface.OnClickListener() {  
            public void onClick(DialogInterface dialog, int whichButton) {  
                AlertDialog2Activity.this.finish();  
            }  
        })  
    .setNegativeButton("No",  
        new DialogInterface.OnClickListener() {  
            public void onClick(DialogInterface dialog, int whichButton) {  
                dialog.cancel();  
            }  
        })  
    );  
AlertDialog alert = builder.create();  
return alert;
```

빌더 객체 생성

대화 상자 객체 생성

목록을 사용하는 대화상자

```
...
public class AlertDialogTest03 extends ActionBarActivity {
...
    protected Dialog onCreateDialog(int id) {
        switch (id) {
            case DIALOG_YES_NO_MESSAGE:
                final CharSequence[] items = { "Red", "Green", "Blue" };

                AlertDialog.Builder builder = new AlertDialog.Builder(this);
                builder.setTitle("색상을 선택하십시오");

                builder.setItems(items, new
                    DialogInterface.OnClickListener() {
                        public void onClick(DialogInterface dialog, int item) {
                            Toast.makeText(getApplicationContext(), items[item],
                                Toast.LENGTH_SHORT).show();
                        }
                    });

                AlertDialog alert = builder.create();
                return alert;
            }
        return null;
    }
...
}
```

색상을 선택하십시오

Red

Green

Blue

체크박스를 사용하는 대화상자

색상을 선택하십시오

- ☐ Red
- ☒ Green
- ☐ Blue

setSingleChoiceItems()의
두 번째 매개변수는 초기
에 선택된 항목을 나타내
는 정수로, -1은 아직 선
택된 항목이 없음을 나타
낸다.

```
...
public class AlertDialogTest04 extends ActionBarActivity {
...
    protected Dialog onCreateDialog(int id) {
        switch (id) {
            case DIALOG_YES_NO_MESSAGE:
                final CharSequence[] items = { "Red", "Green", "Blue" };

                AlertDialog.Builder builder = new AlertDialog.Builder(this);
                builder.setTitle("색상을 선택하십시오");

                builder.setSingleChoiceItems(items, -1, new
                    DialogInterface.OnClickListener() {
                        public void onClick(DialogInterface dialog, int item) {
                            Toast.makeText(getApplicationContext(), items[item],
                                Toast.LENGTH_SHORT).show();
                        }
                    });

                AlertDialog alert = builder.create();
                return alert;
            }
        return null;
    }
...
}
```



```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">

    <Button
        android:id="@+id/finish"
        android:onClick="click"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="종료" />

    <Button
        android:id="@+id/color"
        android:onClick="click"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="색상" />

    <Button
        android:id="@+id/color_check"
        android:onClick="click"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="색상 체크박스" />

</LinearLayout>

```

```

public class MainActivity extends AppCompatActivity {
    final int DIALOG_PAUSED_ID = 1;
    final int DIALOG_YES_NO_MESSAGE = 2;
    final int DIALOG_CHECK_MESSAGE = 3;
    final CharSequence[] items = {"Red", "Green", "Blue"};

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }

    protected Dialog onCreateDialog(int id) {
        AlertDialog.Builder builder = new AlertDialog.Builder(context, this);
        switch (id) {
            case DIALOG_PAUSED_ID:
                builder.setTitle("종료 확인 대화상자")
                    .setMessage("애플리케이션을 종료하시겠습니까?")
                    .setCancelable(false)
                    .setPositiveButton(text, "Yes",
                        new DialogInterface.OnClickListener() {
                            @Override
                            public void onClick(DialogInterface dialogInterface, int i) {
                                MainActivity.this.finish();
                            }
                        })
                    .setNegativeButton(text, "No",
                        new DialogInterface.OnClickListener() {
                            @Override
                            public void onClick(DialogInterface dialogInterface, int i) {
                                dialogInterface.cancel();
                            }
                        });
                break;

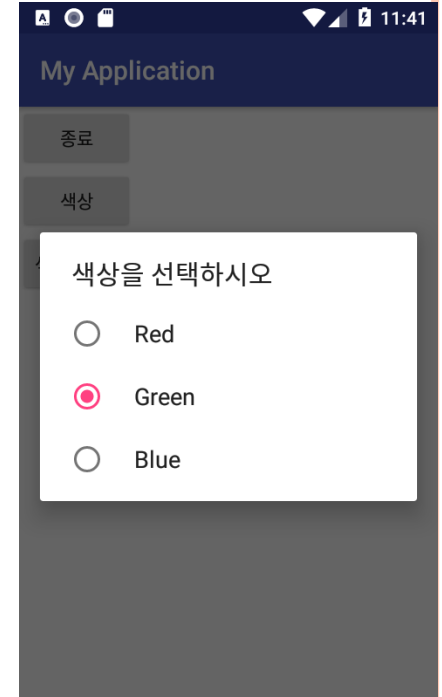
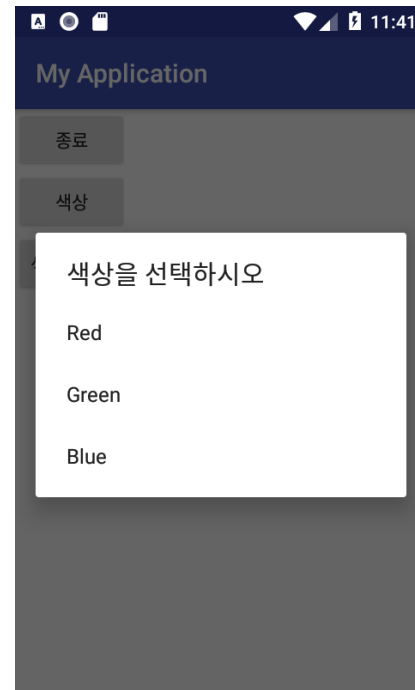
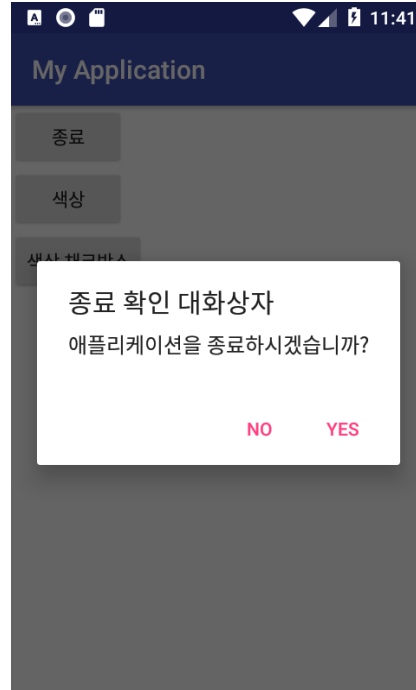
```

```

        case DIALOG_YES_NO_MESSAGE:
            builder.setTitle("색상을 선택하십시오");
            builder.setItems(items, new DialogInterface.OnClickListener() {
                @Override
                public void onClick(DialogInterface dialogInterface, int i) {
                    Toast.makeText(getApplicationContext(), items[i], Toast.LENGTH_SHORT).show();
                }
            });
            break;
        case DIALOG_CHECK_MESSAGE:
            builder.setTitle("색상을 선택하십시오");
            builder.setSingleChoiceItems(items, checkedItem, -1, new DialogInterface.OnClickListener() {
                @Override
                public void onClick(DialogInterface dialogInterface, int i) {
                    Toast.makeText(getApplicationContext(), items[i], Toast.LENGTH_SHORT).show();
                }
            });
            break;
    }
    return builder.create();
}

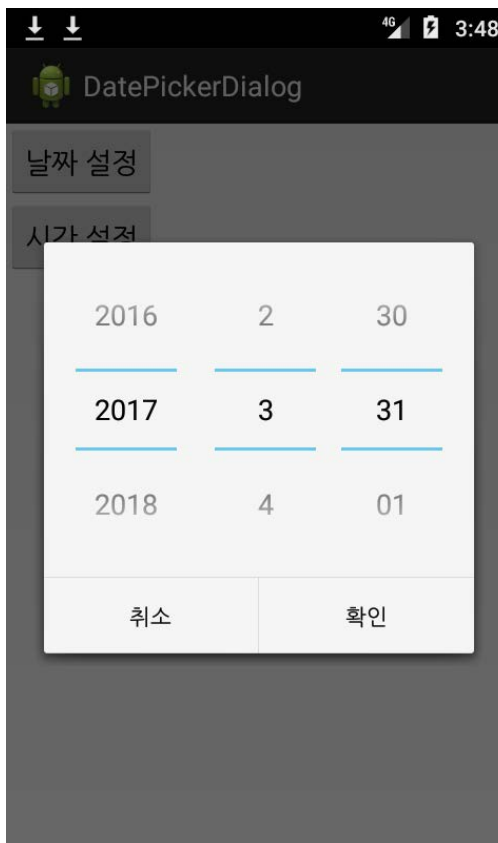
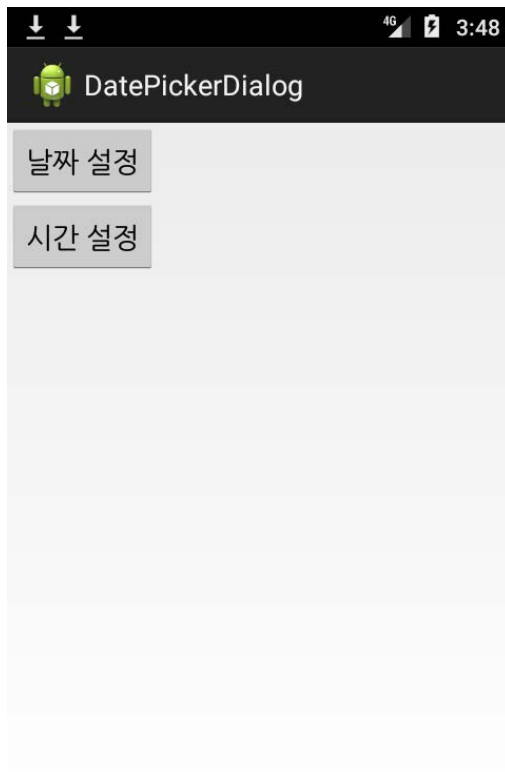
public void click(View view) {
    switch (view.getId()) {
        case R.id.finish:
            showDialog(DIALOG_PAUSED_ID);
            break;
        case R.id.color:
            showDialog(DIALOG_YES_NO_MESSAGE);
            break;
        case R.id.color_check:
            showDialog(DIALOG_CHECK_MESSAGE);
    }
}
}

```



DATEPICKERDIALOG

- 날짜와 시간을 입력받는 대화 상자



DATEPICKERDIALOG

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">

    <Button
        android:id="@+id/button1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="날짜"/>

    <Button
        android:id="@+id/button2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="시간"/>

</LinearLayout>
```

DATEPICKERDIALOG

```
public class MainActivity extends AppCompatActivity {  
    Button btnSelectDate, btnSelectTime;  
  
    static final int DATE_DIALOG_ID = 0;  
    static final int TIME_DIALOG_ID = 1;  
  
    public int year, month, day, hour, minute;  
    private int mYear, mMonth, mDay, mHour, mMinute;  
  
    public MainActivity(){  
        final Calendar c = Calendar.getInstance();  
        mYear = c.get(Calendar.YEAR);  
        mMonth = c.get(Calendar.MONTH);  
        mDay = c.get(Calendar.DAY_OF_MONTH);  
        mHour = c.get(Calendar.HOUR_OF_DAY);  
        mMinute = c.get(Calendar.MINUTE);  
    }  
}
```

DATEPICKERDIALOG

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    btnSelectDate = (Button) findViewById(R.id.button1);
    btnSelectTime = (Button) findViewById(R.id.button2);
    btnSelectDate.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            showDialog(DATE_DIALOG_ID);
        }
    });

    btnSelectTime.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            showDialog(TIME_DIALOG_ID);
        }
    });
}
```



```

private DatePickerDialog.OnDateSetListener mListener = new DatePickerDialog.OnDateSetListener() {
    @Override
    public void onDateSet(DatePicker datePicker, int i, int i1, int i2) {
        year = i;
        month = i1 + 1;
        day = i2;
        Toast.makeText(getApplicationContext(),
            text "날짜: " + year + "-" + month + "-" + day, Toast.LENGTH_SHORT).show();
    }
};

```

```

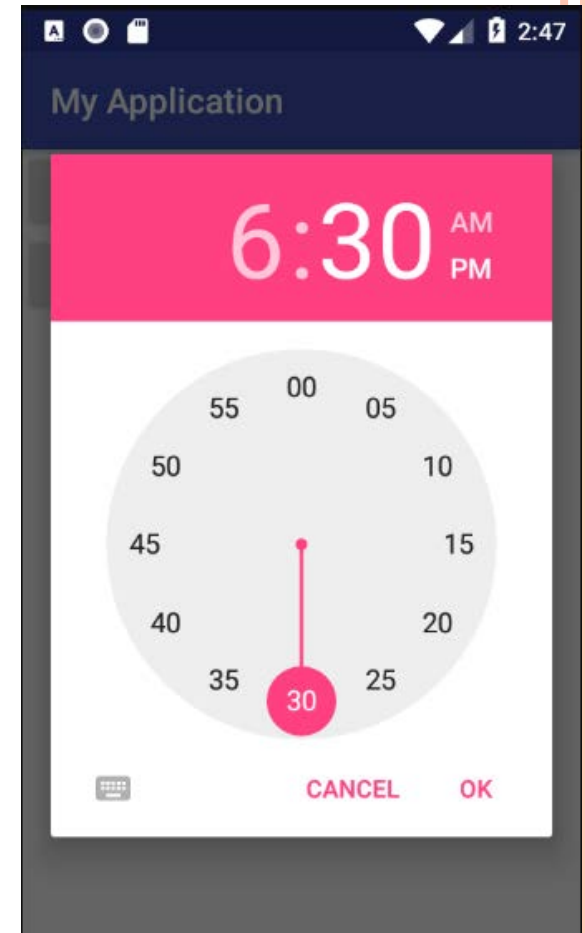
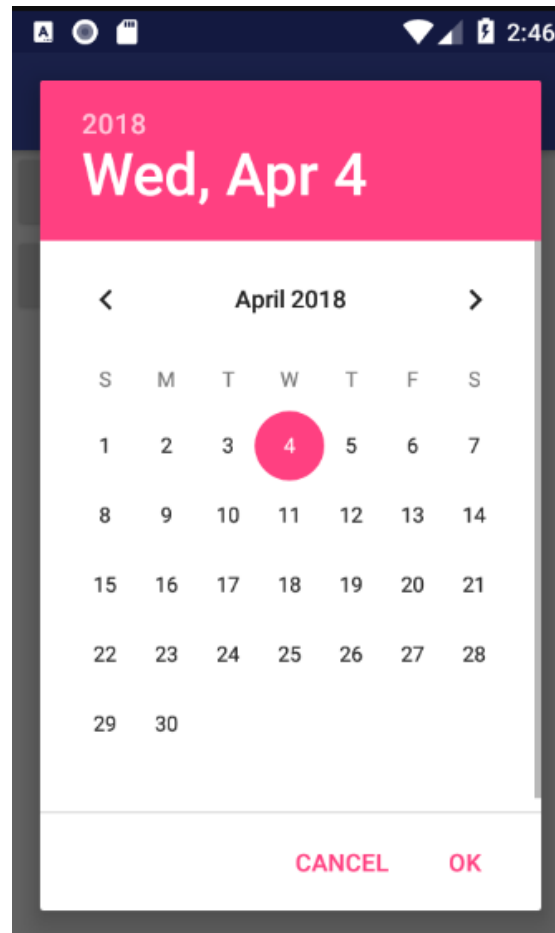
private TimePickerDialog.OnTimeSetListener mListener = new TimePickerDialog.OnTimeSetListener() {
    @Override
    public void onTimeSet(TimePicker timePicker, int i, int i1) {
        hour = i;
        minute = i1;
        Toast.makeText(getApplicationContext(),
            text "시간: " + hour + "-" + minute, Toast.LENGTH_SHORT).show();
    }
};

```

```

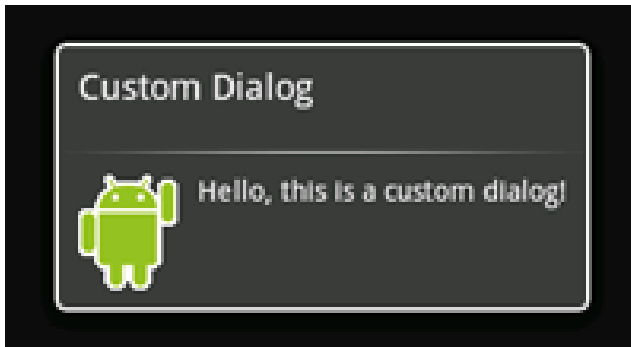
@Override
protected Dialog onCreateDialog(int id) {
    switch (id) {
        case DATE_DIALOG_ID:
            return new DatePickerDialog(context, mListener, year, month, day);
        case TIME_DIALOG_ID:
            return new TimePickerDialog(context, mListener, hour, minute, is24HourView: false);
    }
    return null;
}
}

```

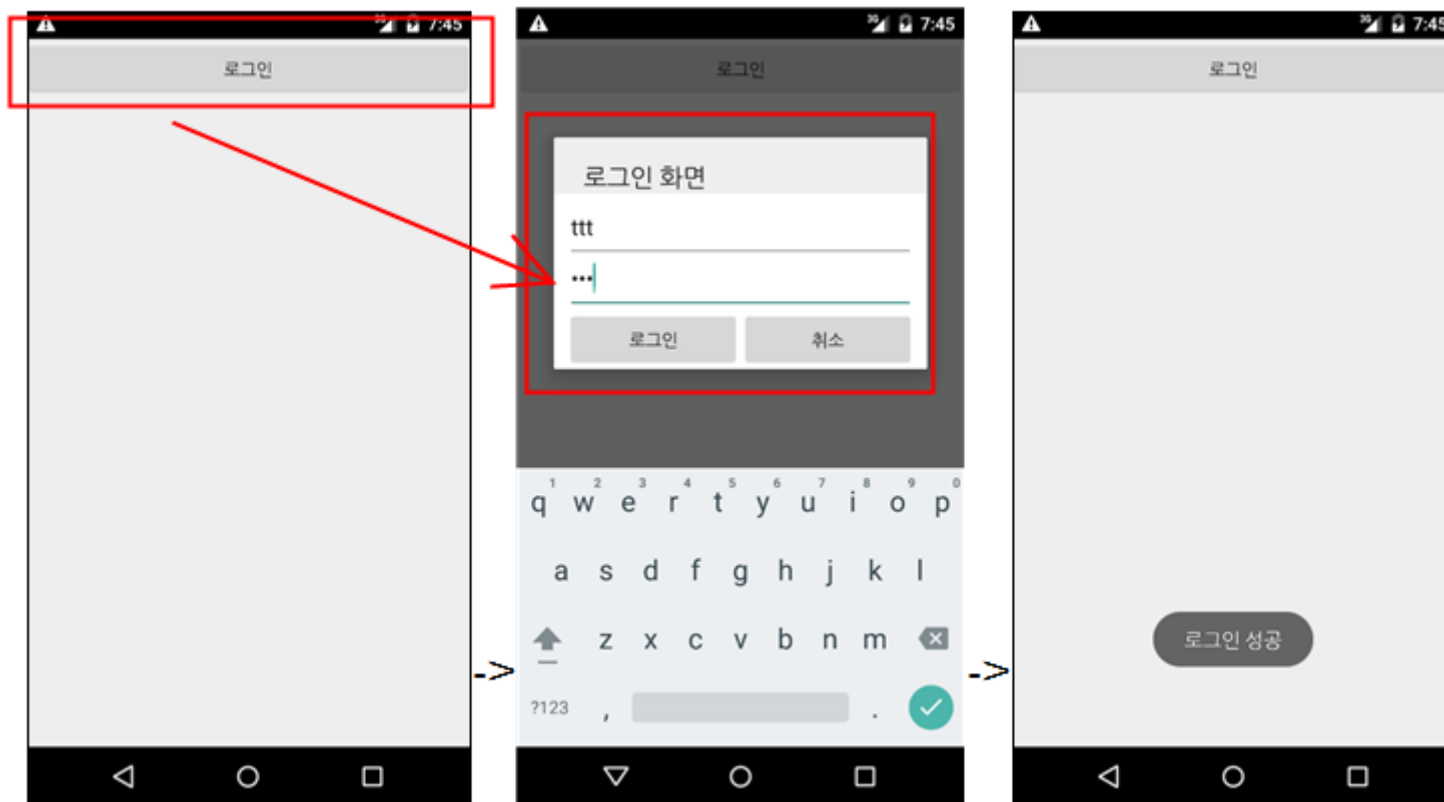


커스텀 대화 상자

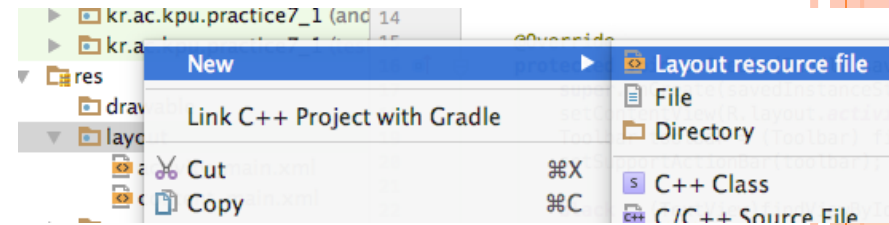
- 사용자가 마음대로 대화 상자의 내용을 디자인할 수 있는 대화 상자



예제: 로그인 대화 상자



대화 상자를 XML로 정의



```
<?xml version="1.0" encoding="utf-8"?>
```

```
<LinearLayout
```

```
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="200dp"
    android:background="#ffffff"
    android:layout_height="wrap_content">
```

```
    <EditText
```

```
        android:id="@+id/username"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="username"
        <requestFocus/>
```

```
    </EditText>
```

```
    <EditText
```

```
        android:id="@+id/password"
        android:hint="password"
        android:inputType="textPassword"
        android:layout_width="match_parent"
        android:layout_height="wrap_content" />
```

로그인 화면

username

password

로그인

취소

```
<LinearLayout
```

```
    android:layout_width="match_parent"
    android:layout_height="wrap_content">
```

```
    <Button
```

```
        android:id="@+id/login"
        android:layout_weight="1"
        android:text="로그인"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
```

```
    <Button
```

```
        android:id="@+id/cancel"
        android:layout_weight="1"
        android:text="취소"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
```

```
</LinearLayout>
```

```
</LinearLayout>
```

```

public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }

    public void onClick(View view) {
        final Dialog loginDialog = new Dialog(context, this);
        loginDialog.setContentView(R.layout.custom_dialog);
        loginDialog.setTitle("로그인 화면");

        Button login = (Button) loginDialog.findViewById(R.id.login);
        Button cancel = (Button) loginDialog.findViewById(R.id.cancel);

        final EditText username = (EditText) loginDialog.findViewById(R.id.username);
        final EditText password = (EditText) loginDialog.findViewById(R.id.password);

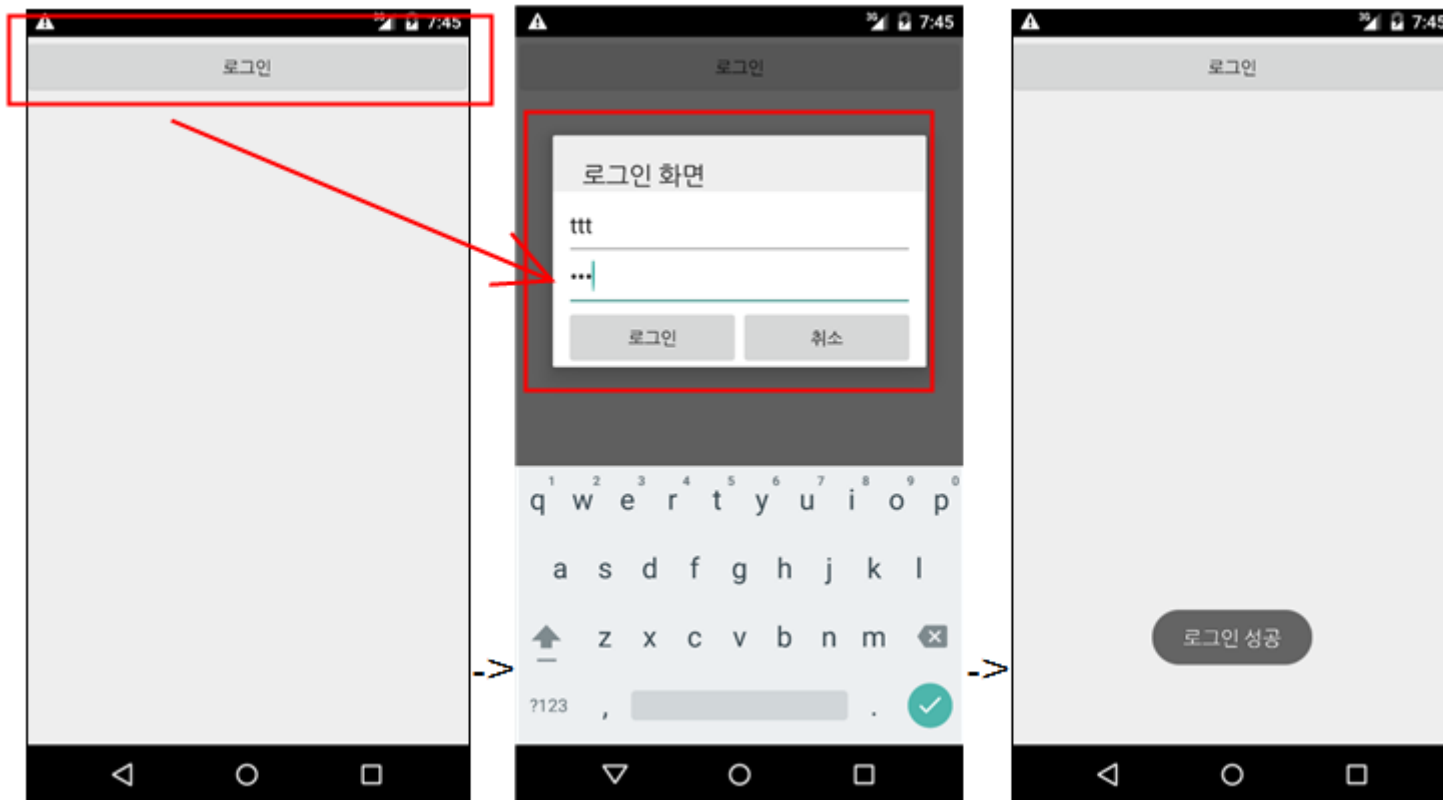
        login.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                if (username.getText().toString().trim().length() > 0 &&
                    password.getText().toString().trim().length() > 0) {
                    Toast.makeText(getApplicationContext(), text "로그인 성공", Toast.LENGTH_SHORT).show();
                    loginDialog.dismiss();
                } else {
                    Toast.makeText(getApplicationContext(), text "다시 입력하십시오.", Toast.LENGTH_SHORT).show();
                }
            }
        });

        cancel.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                loginDialog.dismiss();
            }
        });

        loginDialog.show();
    }
}

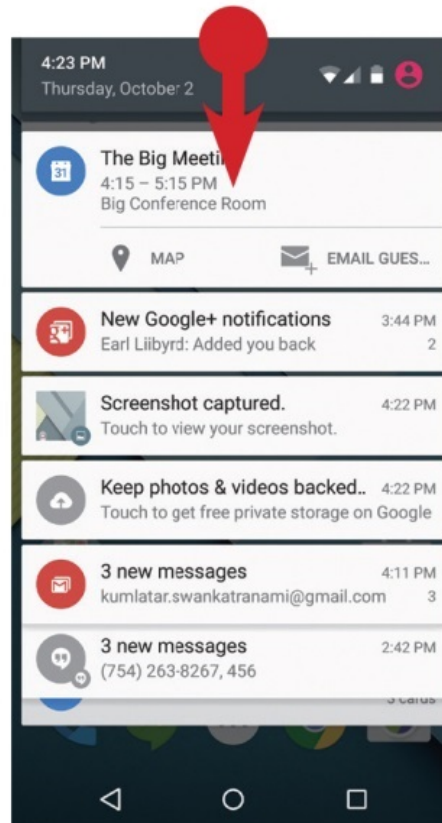
```

실행 결과 (실습)



알림 기능

- 알림기능(notification)은 어떤 이벤트가 발생하였을 때, 앱이 사용자에게 전달하는 메시지이다.



예제: 버튼을 누르면 알람을 보내는 앱

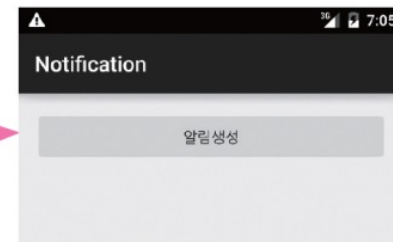
main.xml

<RelativeLayout

```
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:paddingBottom="@dimen/activity_vertical_margin"
android:paddingLeft="@dimen/activity_horizontal_margin"
android:paddingRight="@dimen/activity_horizontal_margin"
android:paddingTop="@dimen/activity_vertical_margin"
tools:context=".NotificationActivity" >
```

<Button

```
android:id="@+id/button"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:layout_alignParentLeft="true"
android:layout_alignParentStart="true"
android:layout_alignParentTop="true"
android:onClick="sendNotification"
android:text="알림생성" />
```



</RelativeLayout>

```
public void sendNotification(View view) {  
  
    // NotificationCompat.Builder를 사용하여 알림을 설정한다.  
    NotificationCompat.Builder builder = new NotificationCompat.Builder(  
        this);  
  
    // 알림에 나타나는 아이콘  
    builder.setSmallIcon(R.drawable.ic_launcher);  
  
    // 알림이 클릭되면 이 인텐트가 보내진다.  
    Intent intent = new Intent(Intent.ACTION_VIEW,  
        Uri.parse("http://www.google.com/"));  
    PendingIntent pendingIntent = PendingIntent.getActivity(this, 0,  
        intent, 0);  
  
    // 사용자가 알림을 터치하면 인텐트가 보내진다.  
    builder.setContentIntent(pendingIntent);  
  
    // 알림에 표시되는 큰 아이콘  
    builder.setLargeIcon(BitmapFactory.decodeResource(getResources(),  
        R.drawable.ic_launcher));  
}
```

```
// 알림 제목
builder.setTitle("알려드립니다.");

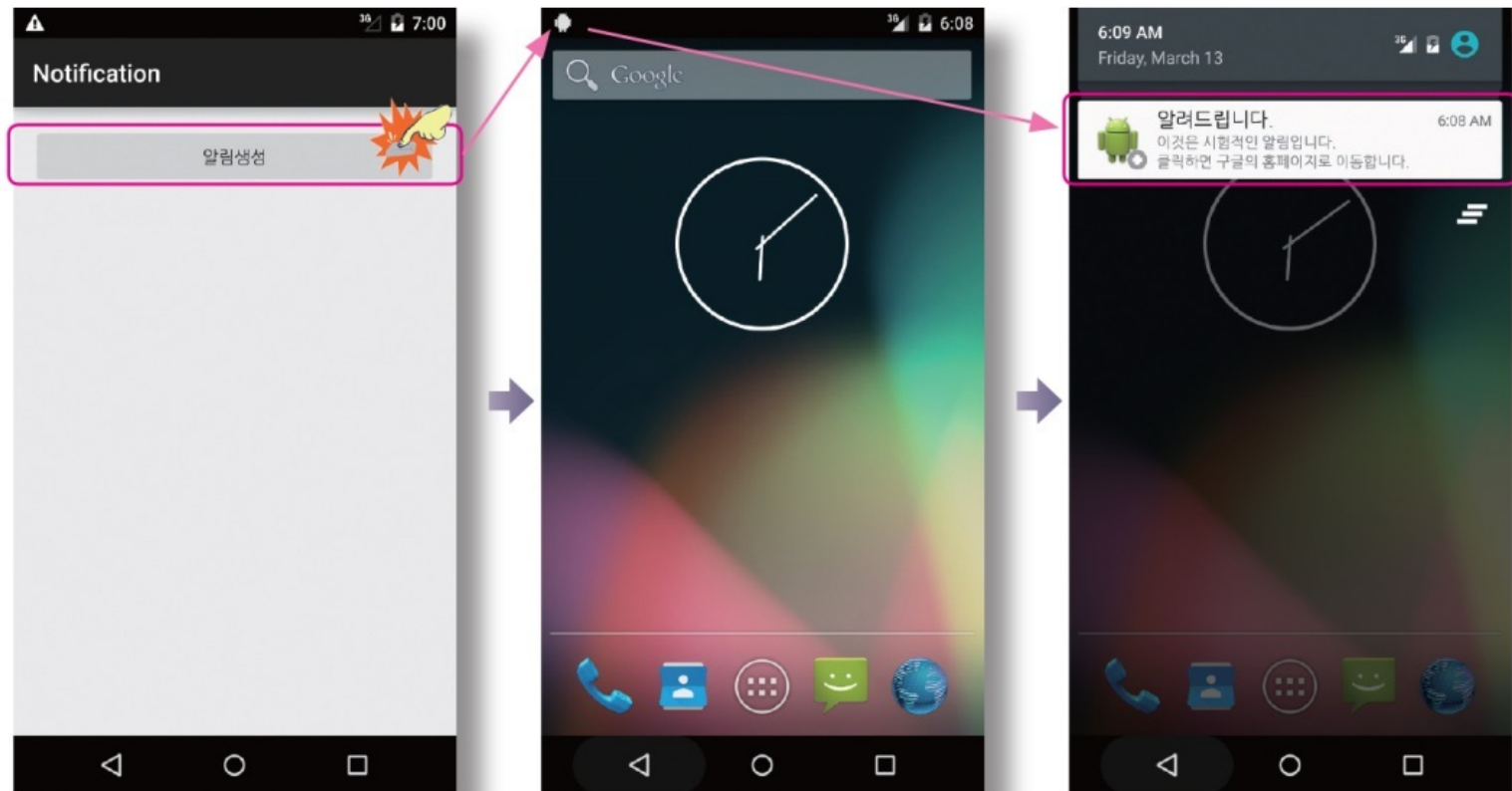
// 알림 콘텐츠
builder.setText("이것은 시험적인 알림입니다.");

// 4.2 이상인 경우에 보여지는 서브 텍스트
builder.setSubText("클릭하면 구글의 홈페이지로 이동합니다.");

NotificationManager notificationManager = (NotificationManager)
getSystemService(NOTIFICATION_SERVICE);

// 알림바에 알림을 표시한다.
notificationManager.notify(NOTIFICATION_ID, builder.build());
}
}
```

실행 결과 (TARGETSDKVERSION을 25로 할 것)



LAB: 예약 앱 작성(실습(2)):

힌트: `TIMEPICKERDIALOG`를 사용하시오.

○ 식당 예약 앱

