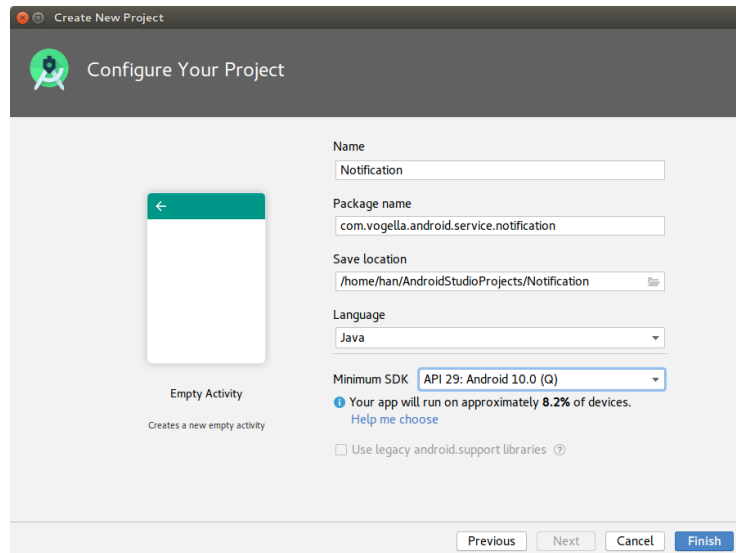
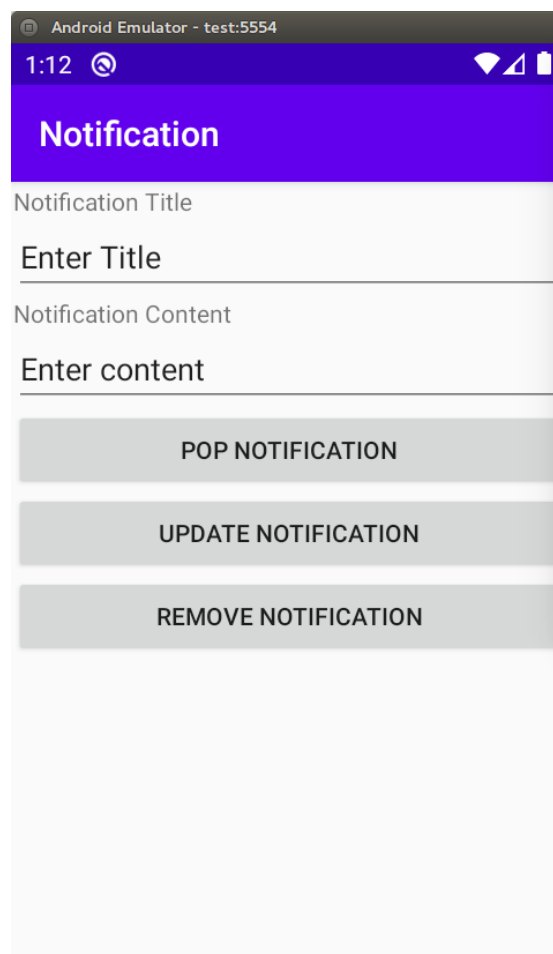


Embedded System Practice Lab12



이번 실습 진행을 위해 새로운 Project "Notification"을 생성해 주었습니다.



위와 같이 실습을 위한 Layer를 추가해 주었습니다.

```
public class MainActivity extends AppCompatActivity {

    private static final String CHANNEL_ID = "Simple Notification";
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        createNotificationChannel();
        Button b1 = findViewById(R.id.button1);
        Button b2 = findViewById(R.id.button2);
        Button b3 = findViewById(R.id.button3);
        b1.setOnClickListener( new Button.OnClickListener(){
            public void onClick(View v){
                addNotification();
            }
        });
        b2.setOnClickListener( new Button.OnClickListener(){
            public void onClick(View v){
                UpdateNotification();
            }
        });
        b3.setOnClickListener( new Button.OnClickListener(){
            public void onClick(View v){
                Notificationoff();
            }
        });
    }
}
```

위는 이번 실습의 onCreate 생성자입니다. Button은 위에서부터 각각 Button 1,2,3으로 순서대로 Notification pop, Notification Update, Notification Off을 수행하도록 하였습니다. 각 button은 시연 영상을 참고하여 클릭했을 때 각각 실행될 method로 연결하여 주었습니다.

```

private void addNotification(){
    EditText Title = findViewById(R.id.editText2);
    EditText Text = findViewById(R.id.editText3);

    Notification.Builder builder = new Notification.Builder( context: this,CHANNEL_ID)
        .setSmallIcon(R.drawable.notification_icon_background)
        .setContentTitle(Title.getText())
        .setContentText(Text.getText());
    NotificationManager manager = (NotificationManager)getSystemService(Context.NOTIFICATION_SERVICE);
    manager.notify( id: 0,builder.build());
}

```

위는 1번 button클릭시 실행되는 addNotification입니다. Text입력으로부터 Title과 Text를 받아와 해당 내용의 Notification을 생성하도록 하였습니다.

```

private void UpdateNotification(){
    EditText Title = findViewById(R.id.editText2);
    EditText Text = findViewById(R.id.editText3);
    Notification.Builder builder = new Notification.Builder( context: this,CHANNEL_ID)
        .setSmallIcon(R.drawable.notification_icon_background)
        .setContentTitle(Title.getText())
        .setContentText(Text.getText());
    NotificationManagerCompat manager = NotificationManagerCompat.from(this);
    manager.notify( id: 0, builder.build());
}

```

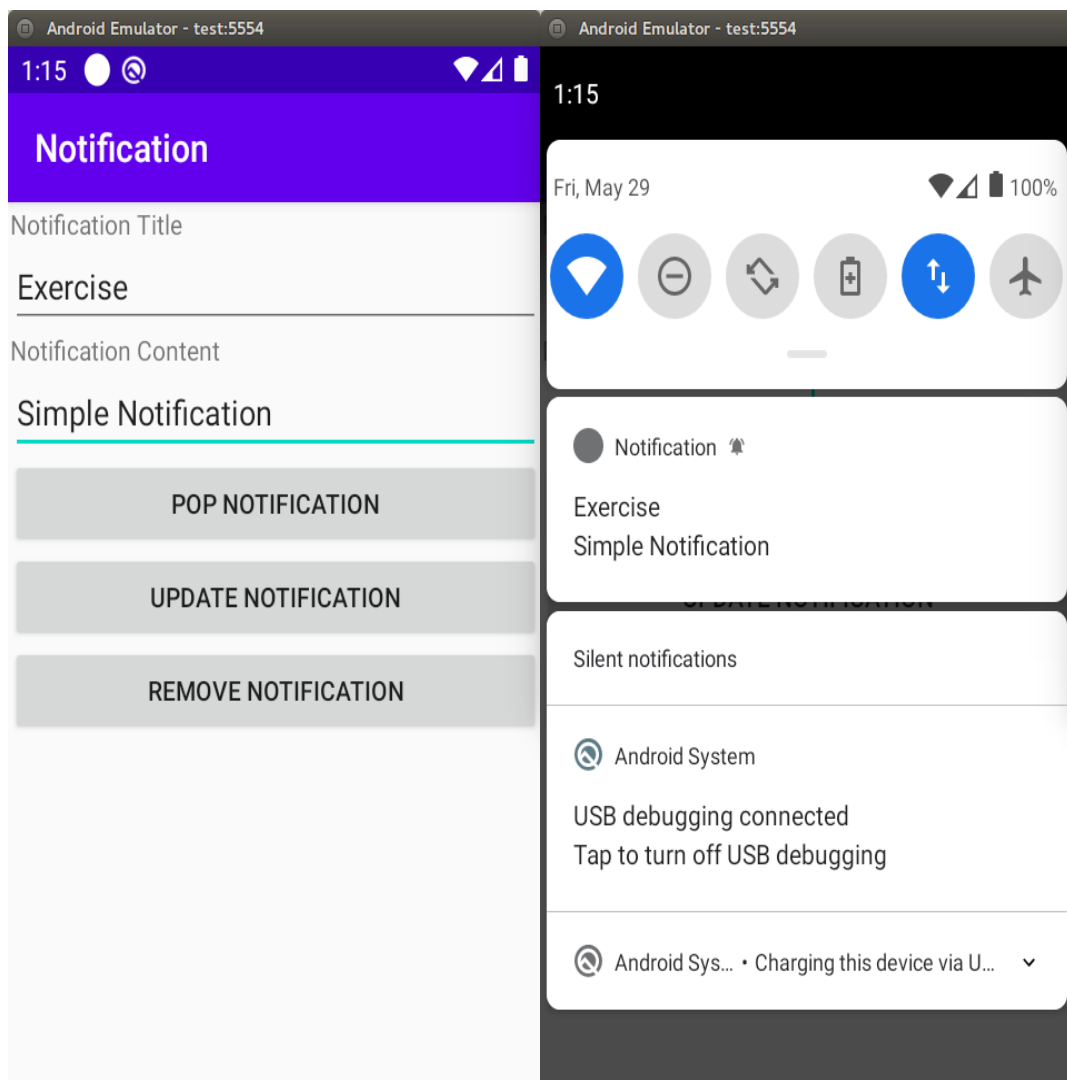
위는 2번 Button 클릭 시 실행되는 UpdateNotification입니다. 현 창의 text들을 다시 받아와 띄어져 있던 Notification을 Update해주는 함수입니다.

```

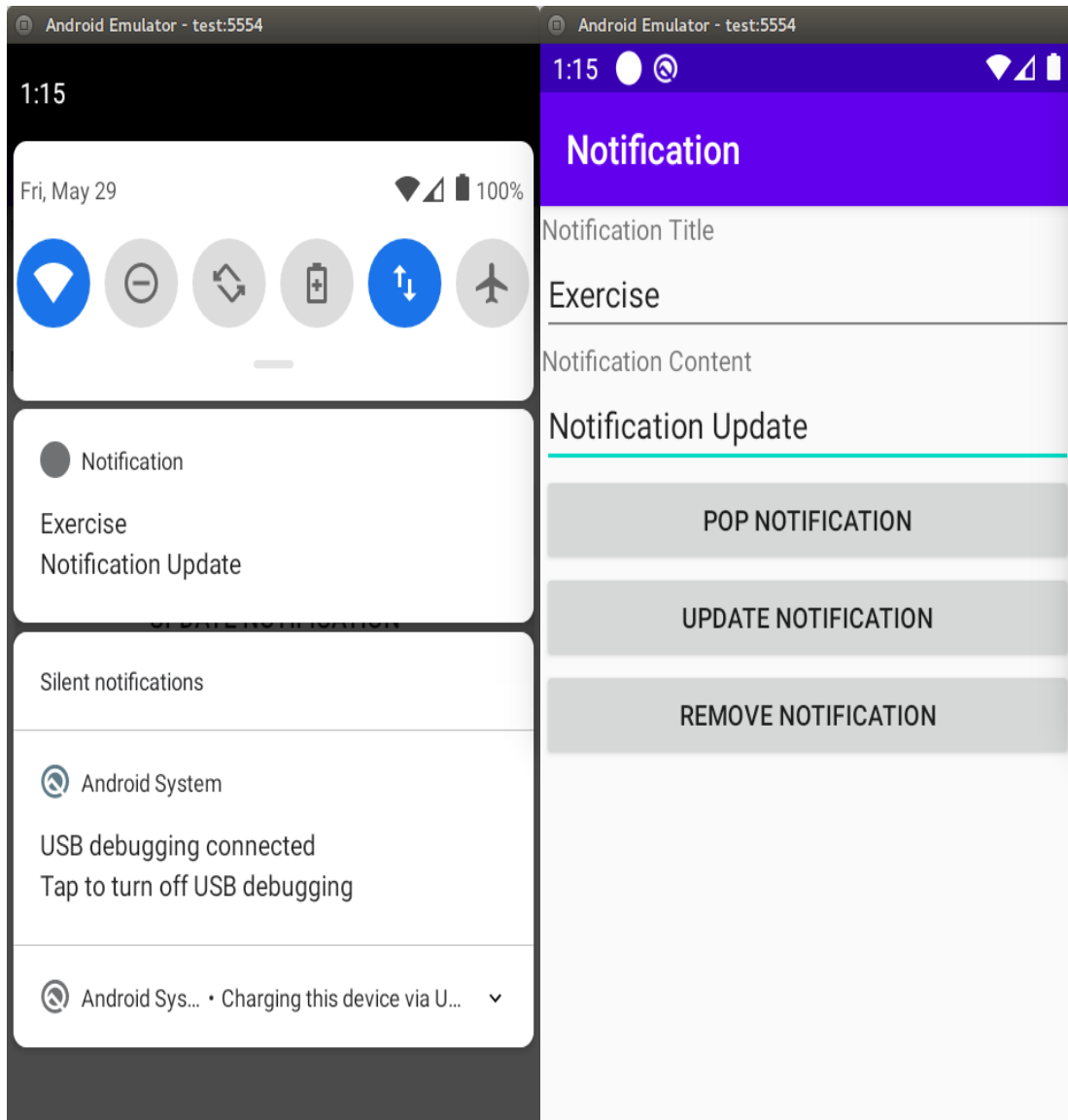
private void Notificationoff(){
    NotificationManager manager = (NotificationManager)getSystemService(Context.NOTIFICATION_SERVICE);
    manager.cancel( id: 0);
}

```

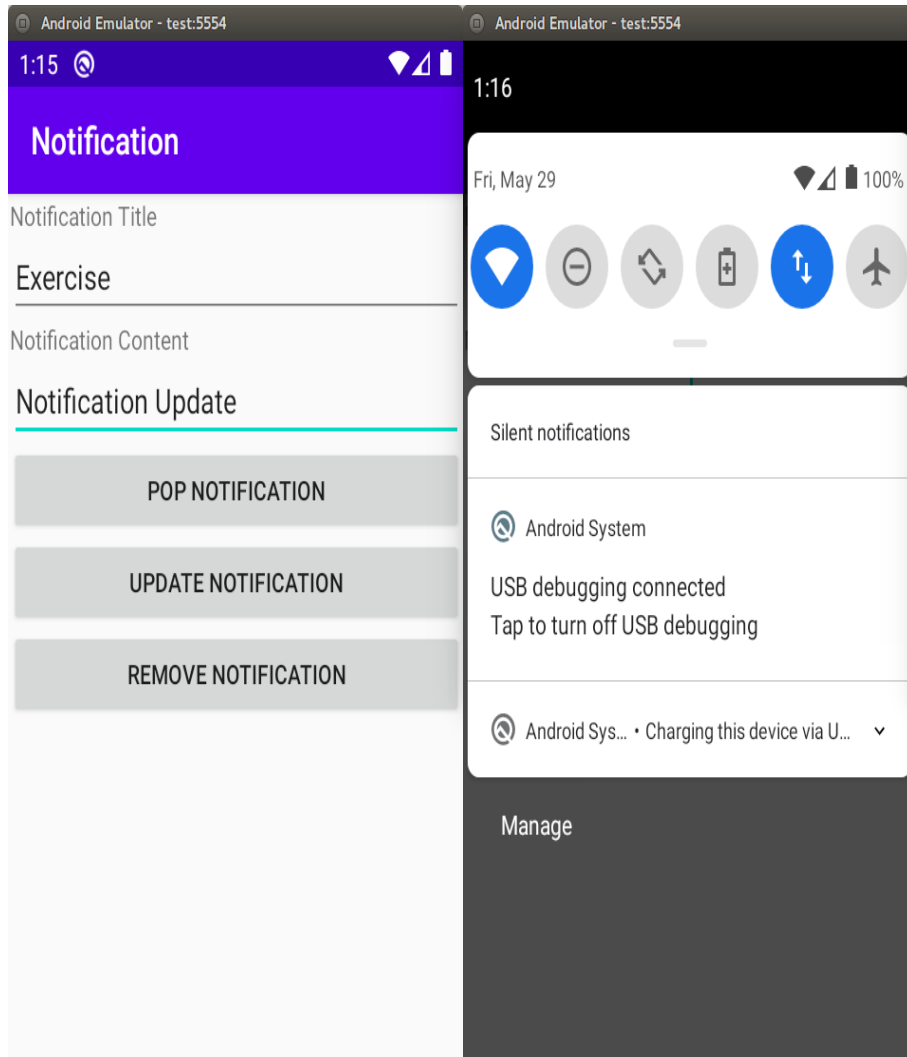
위는 마지막 3번 Button 클릭 시 실행되는 함수로 띄어져 있는 Notification을 제거하는 함수입니다.



위는 첫번째 버튼인 POP NOTIFICATION을 클릭했을 때의 동작입니다.



위는 두 번째 버튼인 UPDATE NOTIFICATION을 클릭했을 때의 동작입니다.



위는 마지막 버튼인 REMOVE NOTIFICATION을 클릭했을 때의 동작입니다.