

MainActivity.java

```
package org.androidtown.events;
import android.app.Activity;
import android.os.Bundle;
import android.view.GestureDetector;
import android.view.Menu;
import android.view.MotionEvent;
import android.view.View;
import android.view.ViewTreeObserver;
import android.widget.TextView;
/**
 * 터치 이벤트를 제스처로 인식하여 처리하는 방법에 대해 알 수 있습니다.
 *
 * @author Mike
 */
public class MainActivity extends Activity {
    TextView TextView01;
    private GestureDetector mGestures = null;
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        TextView01 = (TextView) findViewById(R.id.TextView01);

        // 텍스트뷰를 가지고 있는 레이아웃
        View rootLayout = findViewById(R.id.rootLayout);

        // 터치 모드가 바뀌는 경우 알 수 있도록 표시합니다.
        ViewTreeObserver observer = rootLayout.getViewTreeObserver();
        observer.addOnTouchModeChangeListener(new
ViewTreeObserver.OnTouchModeChangeListener() {
            public void onTouchModeChanged(boolean isTouchMode)
{
                TextView01.append("\n터치 모드가 바뀌었습니다. : "
+ isTouchMode);
            }
        });
        // 길게 눌렀을 때 알 수 있도록 합니다.
        TextView01.setOnLongClickListener(new
View.OnLongClickListener() {
```

```

        public boolean onLongClick(View v) {
            TextView01.append("\nonLongClick: " +
v.toString());
            return true;
        }
    });
    // 제스처로 인식하면 복잡한 이벤트 처리를 좀 더 간단하게 할 수 있습
니다.
    mGestures = new GestureDetector(this,
        new GestureDetector.SimpleOnGestureListener() {
            // fling 이벤트가 발생할 때 처리합니다.
            public boolean onFling(MotionEvent e1,
MotionEvent e2, float velocityX, float velocityY) {
                TextView01.append("\nonFling
\n\tvelocityX = " + velocityX + "\n\tvelocityY=" + velocityY);
                return super.onFling(e1, e2, velocityX,
velocityY);
            }
            // scroll 이벤트가 발생할 때 처리합니다.
            public boolean onScroll(MotionEvent e1,
MotionEvent e2, float distanceX, float distanceY) {
                TextView01.append("\nonScroll
\n\tdistanceX = " + distanceX + "\n\tdistanceY = " + distanceY);
                return super.onScroll(e1, e2, distanceX,
distanceY);
            }
        });
    // 포커스가 바뀌었을 때 알 수 있도록 리스너를 설정합니다.
    TextView01.setOnFocusChangeListener(new
View.OnFocusChangeListener() {
        // focus가 바뀌었을 때 호출됩니다.
        public void onFocusChange(View v, boolean hasFocus) {
            if (hasFocus) {
                TextView01.append("\nonFocusChange, hasFocus : " +
hasFocus);
            } else {
                TextView01.append("\nonFocusChange, hasFocus : " +
hasFocus);
            }
        }
    })
}

```

```
});  
}  
/**  
 * 터치 이벤트를 제스처로 인식할 수 있도록 합니다.  
 */  
public boolean onTouchEvent(MotionEvent event) {  
    if (mGestures != null) {  
        return mGestures.onTouchEvent(event);  
    } else {  
        return super.onTouchEvent(event);  
    }  
}  
  
@Override  
public boolean onCreateOptionsMenu(Menu menu) {  
    // Inflate the menu; this adds items to the action bar if it is present.  
    getMenuInflater().inflate(R.menu.main, menu);  
    return true;  
}  
  
}
```