

Общая информация про акселерометр

Акселерометр или G-сенсор – это датчик, определяющий угол наклона электронного устройства по отношению к земной поверхности. На основании данных от датчика программное обеспечение понимает положение смартфона, и поворачивает изображение на дисплее. Иными словами, именно акселерометр способствует автоматическому повороту экрана в альбомную ориентацию при повороте телефона.

Также этот датчик фиксирует ускорение перемещения устройства в пространстве, одновременно сопоставив три пространственные координаты. Можно сказать, что сенсор измеряет разницу между проекциями абсолютного и гравитационного ускорения.

Акселерометр в смартфоне позволяет использовать:

1. Шагомеры или другие подобные сервисы. Благодаря возможности измерять положение устройства в пространстве, а также его ускорение, сенсор обеспечивает корректную работу шагомера. Это незаменимый помощник для поклонников пробежек или прогулок. Нет необходимости покупать отдельный фитнес-трекер, поскольку в телефон можно установить приложение и использовать его в конкретных целях.

2. Игры. Благодаря G-сенсору процесс игры становится настоящим удовольствием, ведь датчик мгновенно реагирует на минимальную смену положения телефона. Можно отказаться от классической консольной системы управления, поскольку корректировать положение можно путем изменения положения телефона в пространстве.

3. Удобный интерфейс. При смене положения смартфона датчик сразу повернет интерфейс устройства в нужное положение. Эксплуатация устройства максимально удобная и комфортная. Особенно удобен автоповорот экрана при просмотре видео или фильмов.

Подробнее по ссылке:

<https://v-androide.com/instruktsii/obslyuzhivanie/kak-rabotaet-akselerometr-v-smartfone.html>

Создаем новый проект – gravity2

1. В MainActivity

Создаем `SensorManager` и инициализируем его:

```
lateinit var sManager: SensorManager
sManager = getSystemService(Context.SENSOR_SERVICE) as SensorManager
```

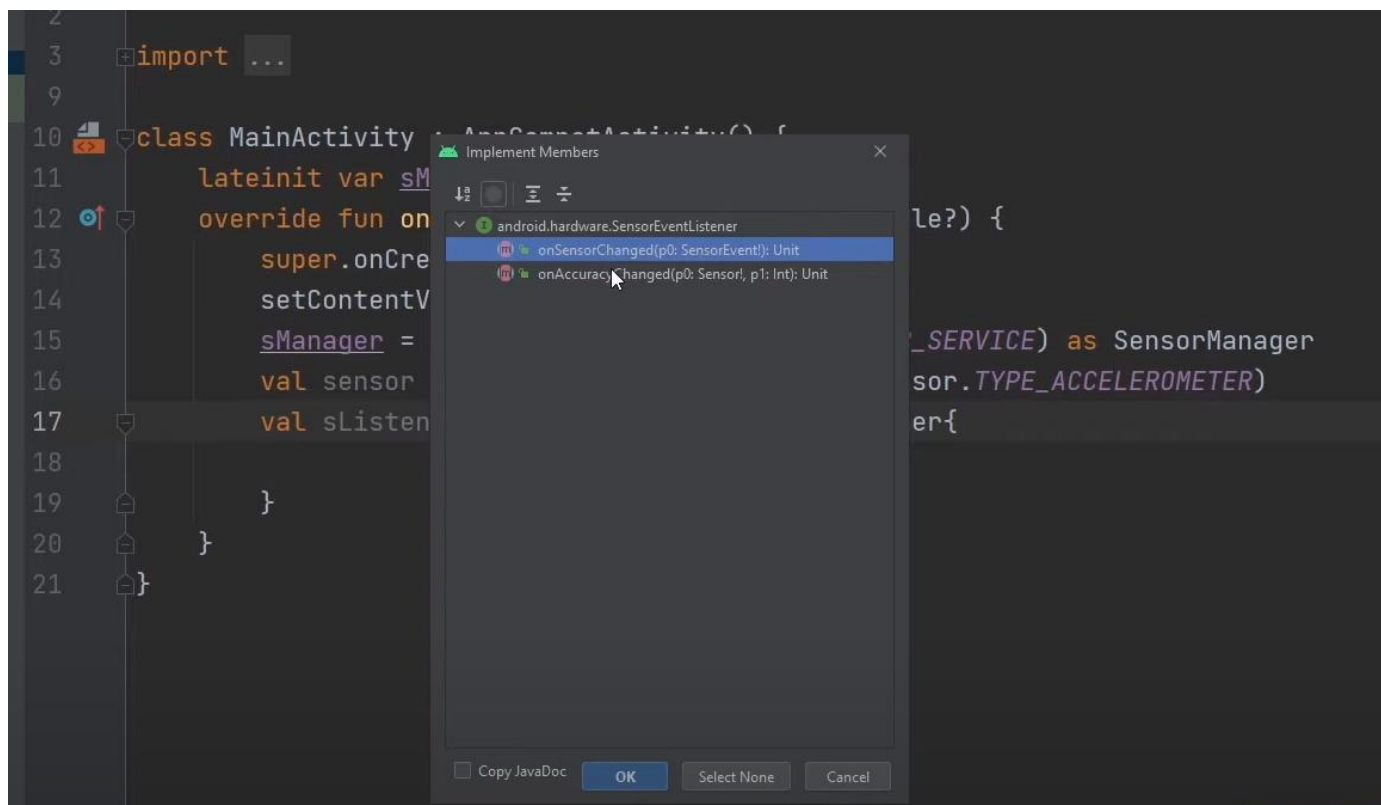
Получим сам сенсор, который будем использовать, в скобках указываем сенсор который хотим использовать – `Sensor.TYPE_ACCELEROMETER`:

```
val sensor = sManager.getDefaultSensor(Sensor.TYPE_ACCELEROMETER)
```

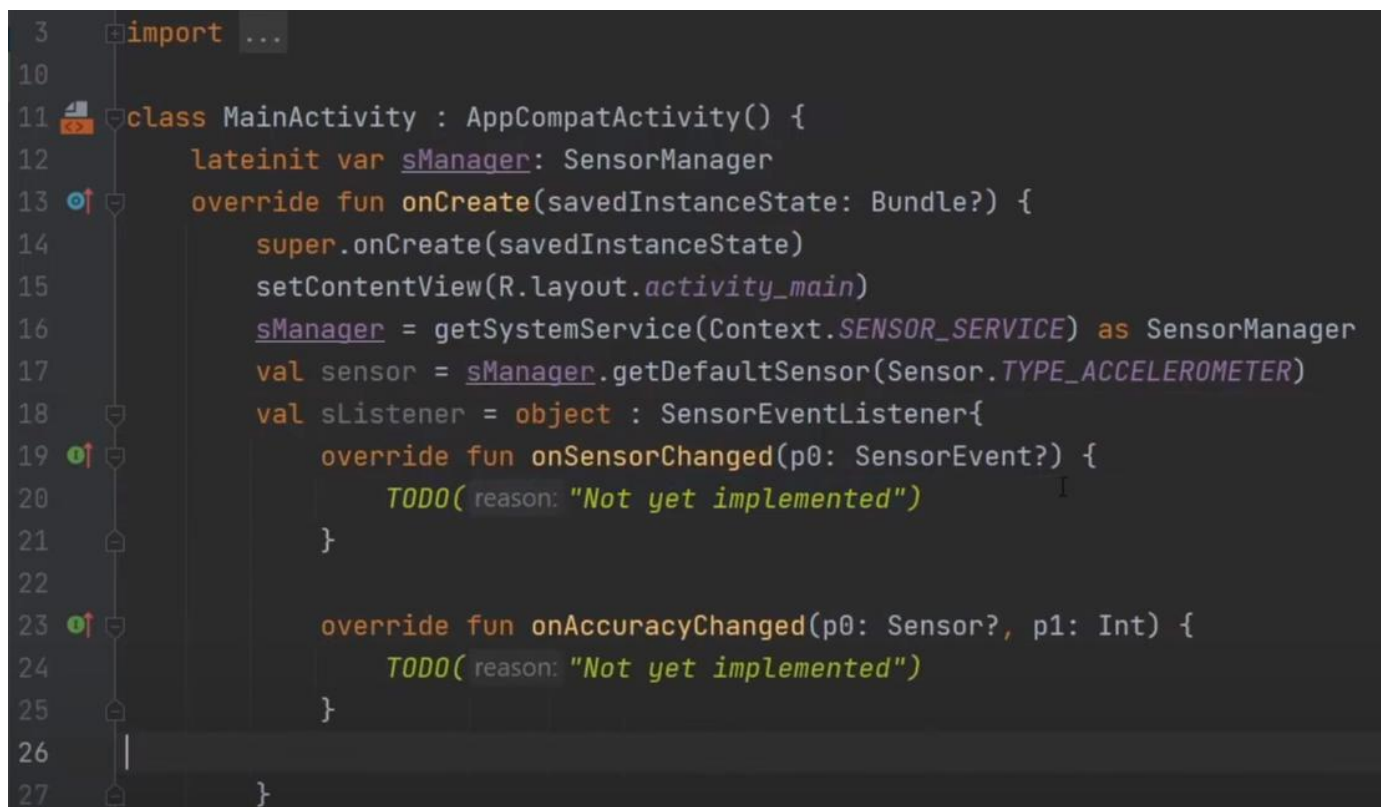
Необходимо чтобы в Активити мы могли получить данные с сенсора `sensor`

```
val sListener = object : SensorEventListener{
```

object подчеркнут красным цветом – добавляем оба метода – Implement Members



Получается:



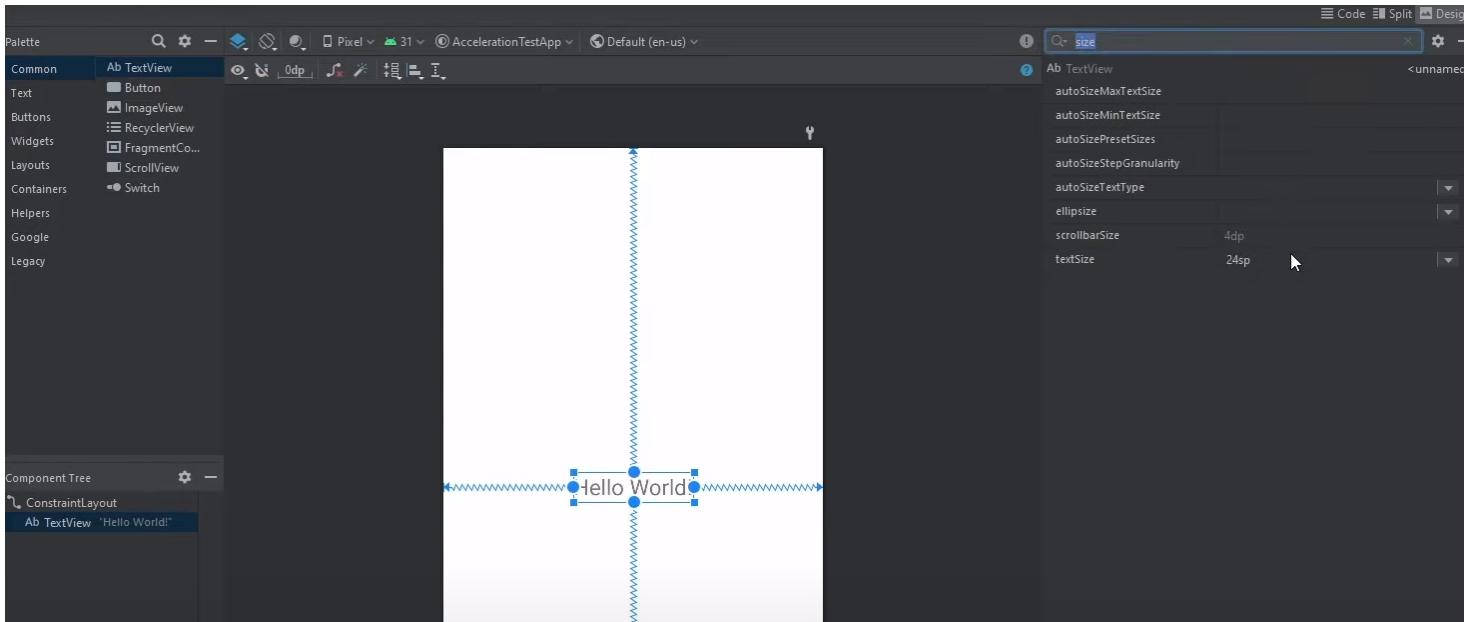
В onAccuracyChanged точность измерения будет указываться.

В onSensorChanged будут приходить данные.

Теперь необходимо получать данные и что-то с ними делать.

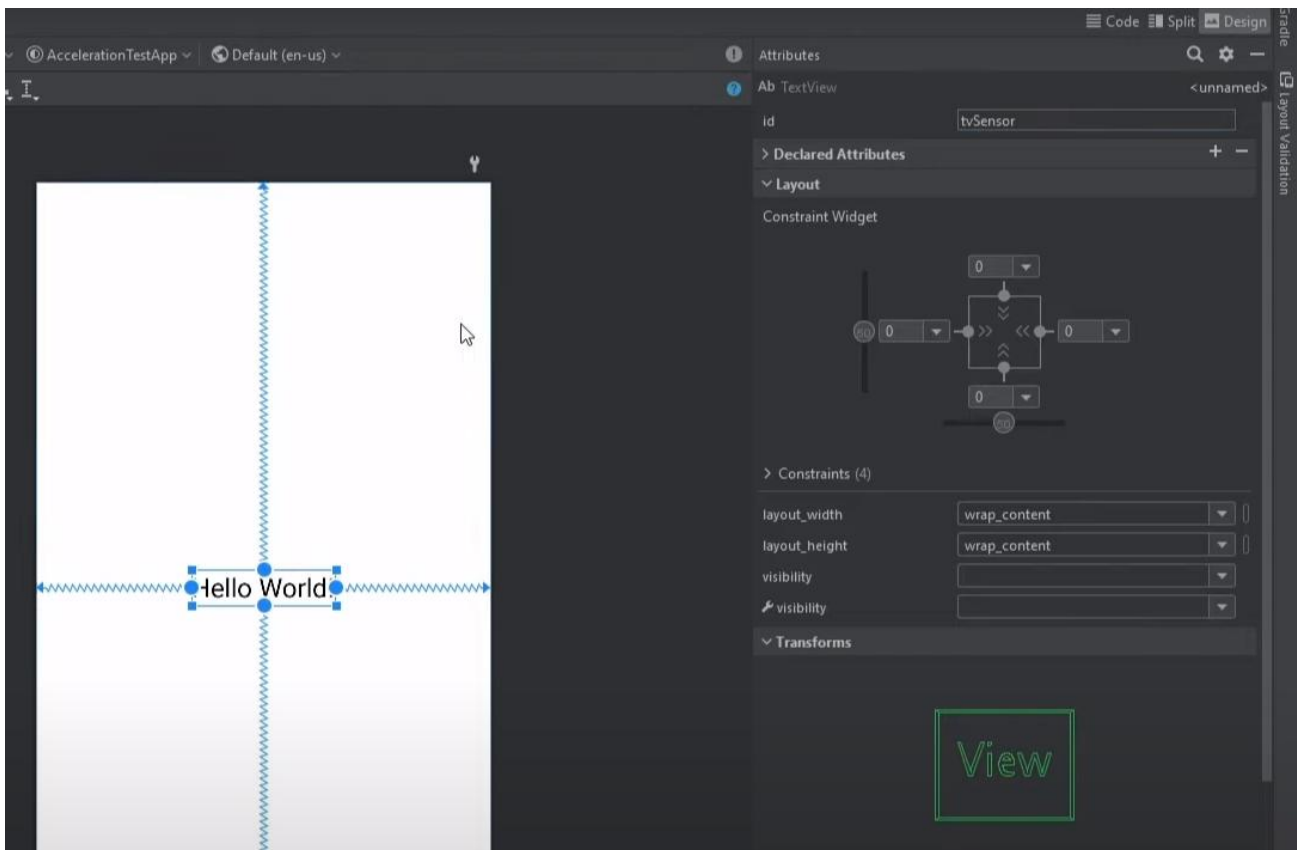
2. Переходим в activity_main

Делаем размер существующего TextView 24sp



Делаем текст черным.

Указываем id – tvSensor



3. Возвращаемся в MainActivity

Пишем код для поиска созданного tvSensor

```
val tvSensor = findViewById<TextView>(R.id.tvSensor)

12 class MainActivity : AppCompatActivity() {
13     lateinit var sManager: SensorManager
14     override fun onCreate(savedInstanceState: Bundle?) {
15         super.onCreate(savedInstanceState)
16         setContentView(R.layout.activity_main)
17         val tvSensor = findViewById<TextView>(R.id.tvSensor)
18
19         sManager = getSystemService(Context.SENSOR_SERVICE) as SensorManager
20         val sensor = sManager.getDefaultSensor(Sensor.TYPE_ACCELEROMETER)
21         val sListener = object : SensorEventListener{
22             override fun onSensorChanged(p0: SensorEvent?) {
23
24             }
25
26             override fun onAccuracyChanged(p0: Sensor?, p1: Int) {
27
28             }
29
30         }
31     }
```

Необходимо указать что мы ждем данные с сенсора, для этого используем сенсор менеджер – sManager

```
sManager.registerListener(sListener, sensor,
SensorManager.SENSOR_DELAY_NORMAL)
```

В строке кода выше: передаем наш sListener, сенсор с которого будем ждать данные sensor, время обновление – SensorManager.SENSOR_DELAY_NORMAL

```
12 class MainActivity : AppCompatActivity() {
13     lateinit var sManager: SensorManager
14     override fun onCreate(savedInstanceState: Bundle?) {
15         super.onCreate(savedInstanceState)
16         setContentView(R.layout.activity_main)
17         val tvSensor = findViewById<TextView>(R.id.tvSensor)
18
19         sManager = getSystemService(Context.SENSOR_SERVICE) as SensorManager
20         val sensor = sManager.getDefaultSensor(Sensor.TYPE_ACCELEROMETER)
21         val sListener = object : SensorEventListener{
22             override fun onSensorChanged(p0: SensorEvent?) {
23
24             }
25
26             override fun onAccuracyChanged(p0: Sensor?, p1: Int) {
27
28             }
29
30         }
31         sManager.registerListener(sListener, sensor, SensorManager.SENSOR_DELAY_NORMAL)
32     }
```

В `onSensorChanged` поменяем переменную `p0` на `sEvent`.

Мы будем брать значения с позиции 0, 1 и 2. Где с позиции 0 будет значение с оси X, на позиции 1 будет значение с оси Y, на позиции 2 будет значение с оси Z.

Эти значение будем брать и все показывать в наше `TextView`.

Пишем для оси X:

```
val value = sEvent?.values
val sData = "X: ${value?.get(0)}"
tvSensor.text = sData
```

Получаем:

```
12 class MainActivity : AppCompatActivity() {
13     lateinit var sManager: SensorManager
14     override fun onCreate(savedInstanceState: Bundle?) {
15         super.onCreate(savedInstanceState)
16         setContentView(R.layout.activity_main)
17         val tvSensor = findViewById<TextView>(R.id.tvSensor)
18
19         sManager = getSystemService(Context.SENSOR_SERVICE) as SensorManager
20         val sensor = sManager.getDefaultSensor(Sensor.TYPE_ACCELEROMETER)
21         val sListener = object : SensorEventListener {
22             override fun onSensorChanged(sEvent: SensorEvent?) {
23                 val value = sEvent?.values
24                 val sData = "X: ${value?.get(0)}"
25                 tvSensor.text = sData
26             }
27
28             override fun onAccuracyChanged(p0: Sensor?, p1: Int) {
29
30             }
31
32         }
33         sManager.registerListener(sListener, sensor, SensorManager.SENSOR_DELAY_NORMAL)
34     }
35 }
```

Теперь если запустим приложение мы увидим, как меняются значения по оси X от 0 до 9,81

Дописываем код для оси Y и Z. Ставим символы `\n` чтобы значения на экране отображались с новой строки (т.е. в столбик):

```
val sData = "X: ${value?.get(0)}\nY: ${value?.get(1)}\nZ: ${value?.get(2)}"
```

Получаем:

```
14 override fun onCreate(savedInstanceState: Bundle?) {
15     super.onCreate(savedInstanceState)
16     setContentView(R.layout.activity_main)
17     val tvSensor = findViewById<TextView>(R.id.tvSensor)
18
19     sManager = getSystemService(Context.SENSOR_SERVICE) as SensorManager
20     val sensor = sManager.getDefaultSensor(Sensor.TYPE_ACCELEROMETER)
21     val sListener = object : SensorEventListener{
22         override fun onSensorChanged(sEvent: SensorEvent?) {
23             val value = sEvent?.values
24             val sData = "X: ${value?.get(0)}\nY: ${value?.get(1)}\nZ: ${value?.get(2)}"
25             tvSensor.text = sData
26         }
27
28         override fun onAccuracyChanged(p0: Sensor?, p1: Int) {
29
30         }
31     }
32 }
33 sManager.registerListener(sListener, sensor, SensorManager.SENSOR_DELAY_NORMAL)
34 }
```

Тестируем на телефоне