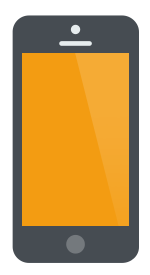
Конспект по Мобильной разработке №1

существует 3 режима отображения:

1. Портретный



2. Альбомный



3. Квадратный (*уже не используется*)



Портретный режим более привычный телефонам а альбомный планшетам.

Существует несколько видов поворотов: автоповорот и фиксированная ориентация (поворот, который существует изначально и не меняется).

**Фиксированная ориентация**

Для того, что бы сделать фиксированную ориентацию достаточно нескольких шагов.

Шаг 1

Дописать код в MainActivity (в метод onCreate):

***@Override***

***public void onCreate(Bundle savedInstanceState) {***

***super.onCreate(savedInstanceState);***

***setContentView(R.layout.main);***

***setRequestedOrientation(ActivityInfo.SCREEN\_ORIENTATION\_LANDSCAPE);***

***}***

Или же можно задать это ограничение в AndroidManifest.xml:

***<activity android:name="MainActivity"***

***android:label="@string/app\_name"***

***android:screenOrientation="landscape">***

***<intent-filter>***

***<action android:name="android.intent.action.MAIN"/>***

***<category android:name="android.intent.category.LAUNCHER"/>***

***</intent-filter>***

***</activity>***

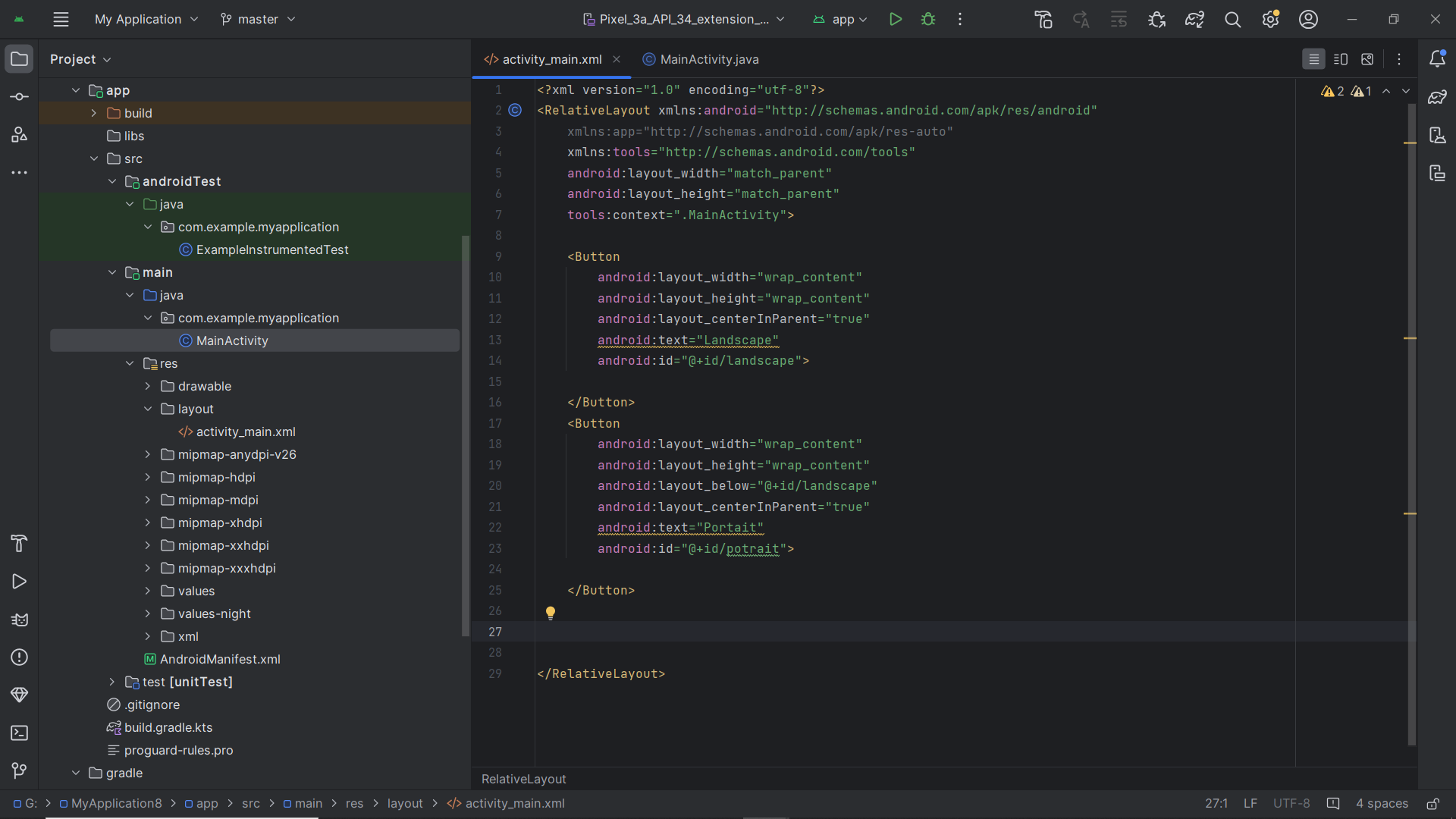
В строке 3 мы указываем что отображение будет только в альбомном виде.

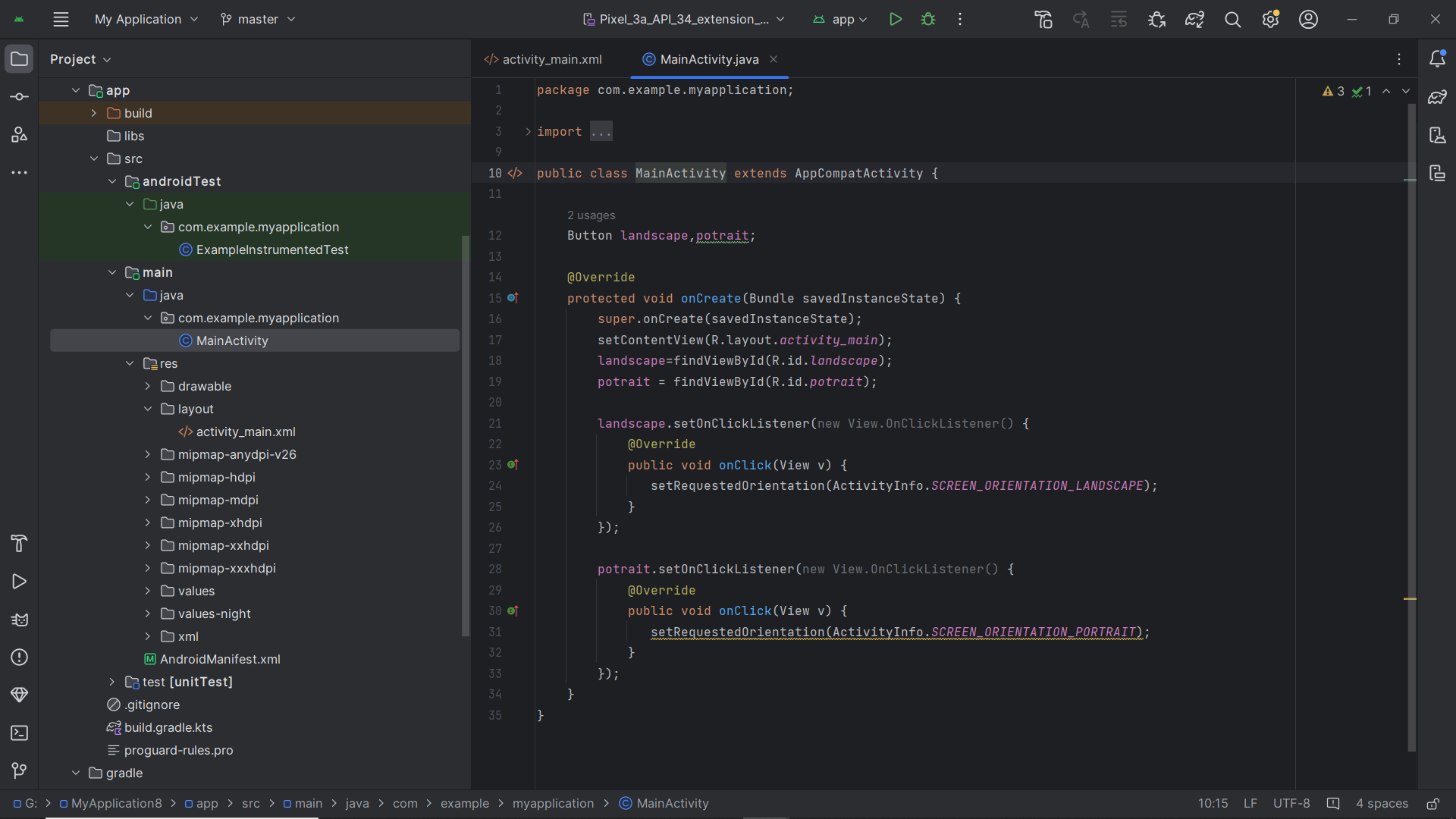
Для портретного режима стоит использовать:

***android:screenOrientation="portrait"***

**Переворот по нажатию на кнопку**

Также в видео показывался еще один способ.Он заключается в переключении ориентации экрана с помощью кнопки. Код который я перепечатала ниже:



\

**Автопереворот**

Для этой операции нам понадобится сделать несколько важных шагов. Сначала нужно создать новую подпапку в папке res. Выделяем папку res, вызываем из него контекстное меню и последовательно выбираем команды New | Android resource directory. В диалоговом окне из выпадающего списка Resource type: выбираем layout. В списке Available qualifiers: находим элемент Orientation и переносим его в правую часть Chosen qualifiers: с помощью кнопки с двумя стрелками. По умолчанию у вас появится имя папки layout-port в первой строке Directory Name:. Но нам нужен альбомный вариант, поэтому в выпадающем списке Screen orientation выбираем Landscape. Теперь название папки будет layout-land.

Далее Вызываем контекстное меню у папки layout-land и выбираем команды New | Layout Resource File. В диалоговом окне присваиваем имя activity\_main.xml, которое должно совпадать с именем существующего файла. Во втором поле вводим LinearLayout, по мере ввода появится подсказка, облегчающая выбор.

Откроем созданный файл и модифицируем его следующим образом.

Пример писался во времена Android 2.3. Сейчас проще создать нужную разметку через ConstraintLayout

<?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">

<LinearLayout

android:orientation="vertical"

android:layout\_height="wrap\_content"

android:layout\_width="match\_parent"

android:layout\_gravity="center"

android:paddingLeft="20dp"

android:paddingRight="20dp">

<EditText

android:layout\_height="wrap\_content"

android:id="@+id/editText"

android:layout\_width="match\_parent"/>

<TableLayout

android:layout\_height="wrap\_content"

android:layout\_width="wrap\_content"

android:layout\_gravity="center"

android:stretchColumns="\*">

<TableRow>

<Button

android:id="@+id/button1"

android:layout\_height="wrap\_content"

android:layout\_width="match\_parent"

android:text="Первая кнопка"/>

<Button

android:id="@+id/button2"

android:layout\_height="wrap\_content"

android:layout\_width="match\_parent"

android:text="Вторая кнопка"/>

</TableRow>

<TableRow>

<Button

android:id="@+id/button3"

android:layout\_height="wrap\_content"

android:layout\_width="match\_parent"

android:text="Третья кнопка"/>

<Button

android:id="@+id/button4"

android:layout\_height="wrap\_content"

android:layout\_width="match\_parent"

android:text="Четвёртая кнопка"/>

</TableRow>

<TableRow>

<Button

android:id="@+id/button5"

android:layout\_height="wrap\_content"

android:layout\_width="match\_parent"

android:text="Пятая кнопка"/>

<Button

android:id="@+id/button6"

android:layout\_height="wrap\_content"

android:layout\_width="match\_parent"

android:text="Шестая кнопка"/>

</TableRow>

</TableLayout>

</LinearLayout>

</LinearLayout>