# Содержание

[**Assignment #5: Создание второй Activity и перехода между экранами**](#_mx6yb3y3dra0) **2**

[Step 1](#_ciidrxmeyosq) 3

[Step 2](#_hucjrmg0zlh6) 3

[Step 3](#_ynnpcfmzv3oe) 4

[Step 4](#_jfbg0h3ofa0u) 5

[Step 5](#_9f7lkn9a4aa7) 6

[Step 6](#_rywsor96p3ja) 7

[Step 7](#_3dpx6byt5247) 7

[Step 9](#_vifqgea9s5g2) 8

[**Assignment #6: Передача параметров между Activity**](#_1klh9cc98nua) **10**

[Step 1](#_79zn7r3m7bpn) 11

[Step 2](#_siftu8wk2p6l) 11

[Step 3](#_tifbrk5wdxim) 12

[Step 4](#_bd7en3ncfcig) 12

[Step 5](#_otaih3prnywm) 13

[Step 6](#_wv5m4up6colt) 13

Description

В этом воркшопе мы добавим в проект дополнительной activity; научимся переходить от одной activity к другой, и назад; передавать данные между ними.

# 

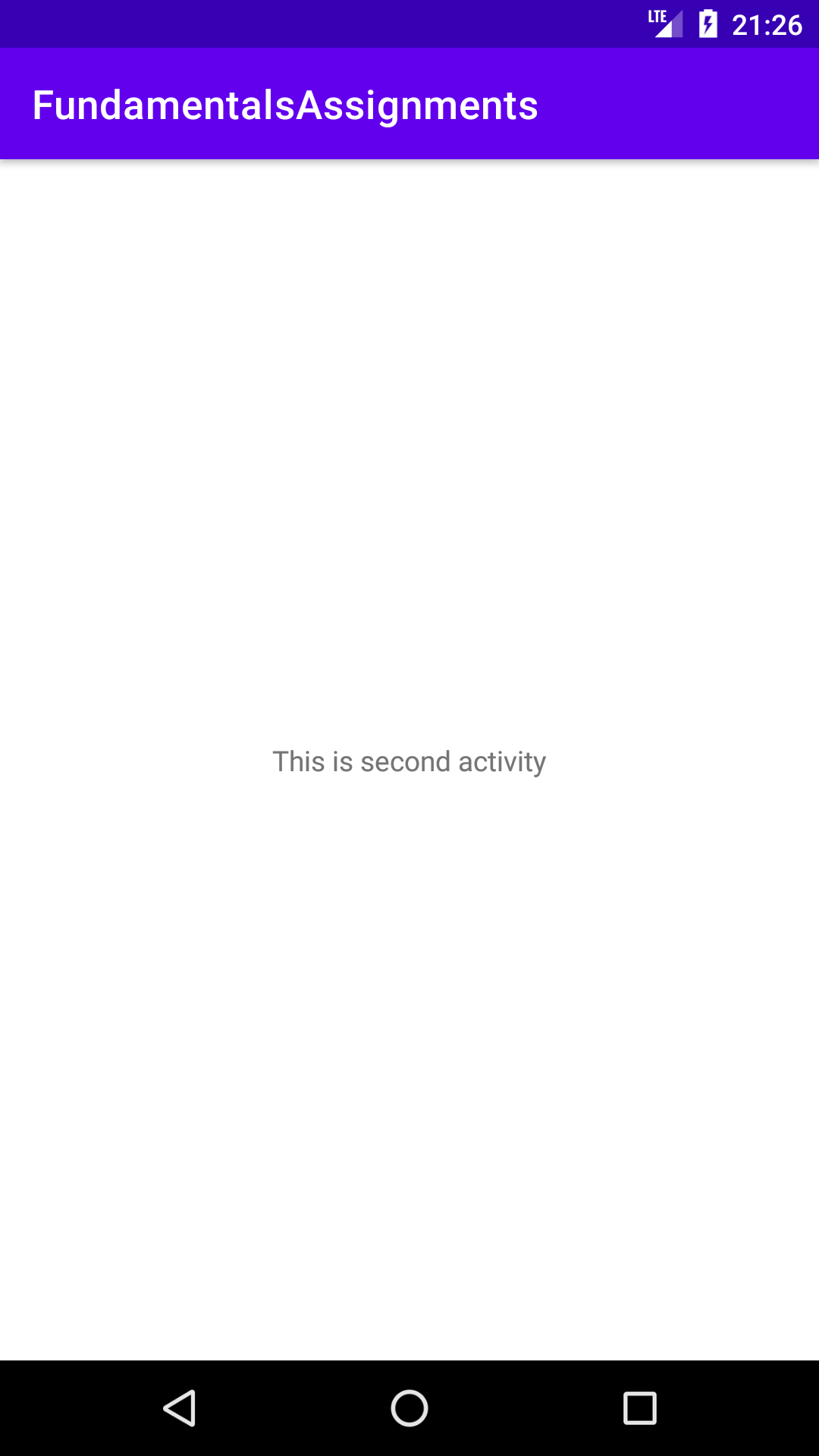
# Assignment #5: Создание второй Activity и перехода между экранами

**Задача:**

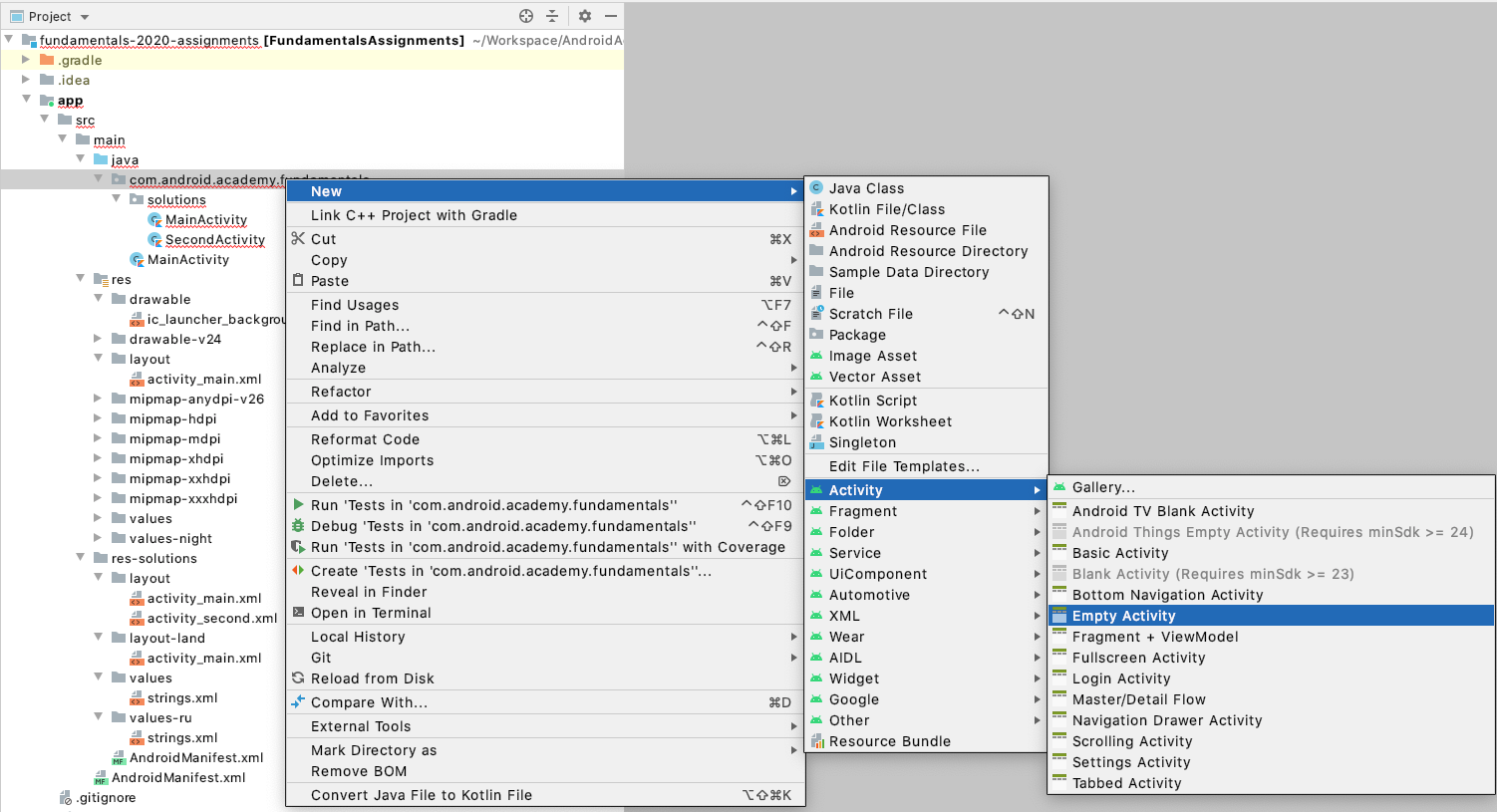
*Создание второй activity и переход на нее.*

**Ответ**:

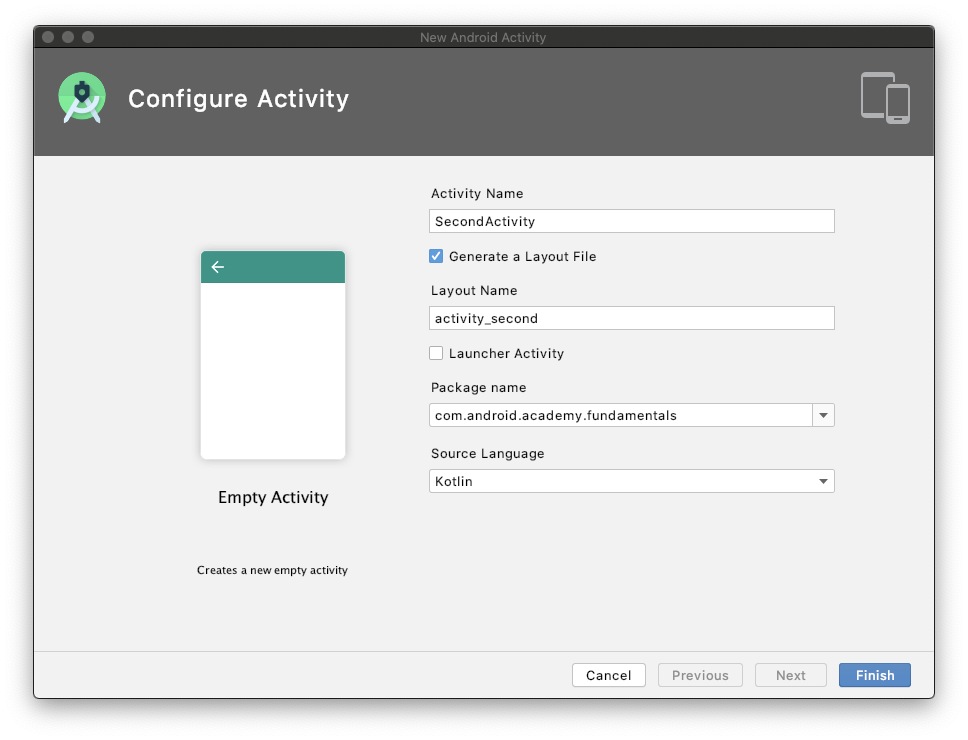
*В конце задания после перехода на другой экран вы увидите следующую картинку:*



## Step 1

* Нажмите правой кнопкой мыши на package где планируется создать активити. В нашем случае это "com.android.academy.fundamentals"
* В выпавшем меню выберите "New -> Activity -> Empty Activity" как показано на картинке ниже:  
    
  

## Step 2

* В появившемся диалоге укажите имя новой активити, например, "SecondActivity"
* Укажите имя layout файла, например "activity\_second"
* В качестве языка программирования "Source Language" выбираем Kotlin. На нем мы дальше будем писать логику в Activity  
    
  **Дополнительно:**  
  Есть галочка "Launcher Activity", которая отвечает за то, что эта активити станет стартовой в приложении. Пока нам это не нужно, но вы можете уже поспрашивать своих менторов зачем это.  
    
  

## Step 3

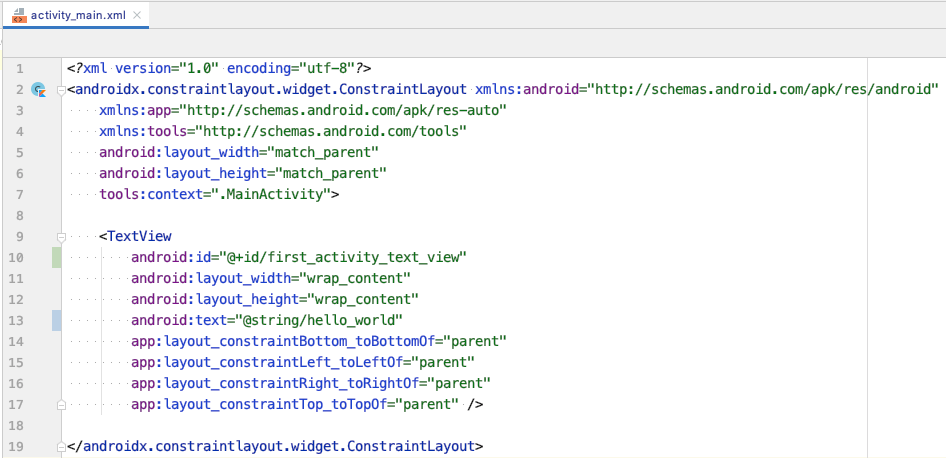
Проверим что в манифесте появилась новая Activity и что категория Launcher стоит у MainActivity:



## Step 4

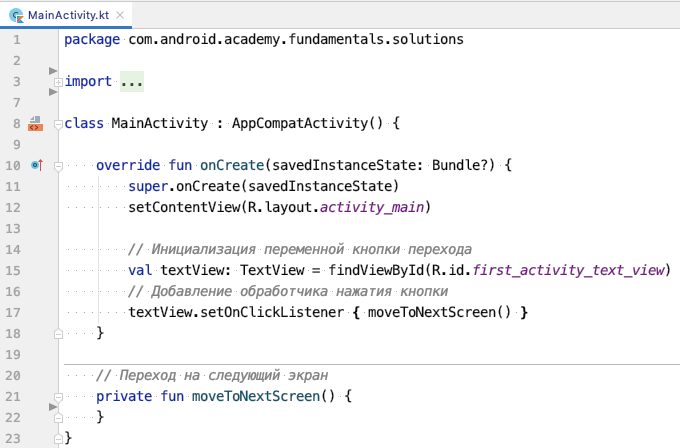
Вам уже показали ранее что такое id у View. Сейчас нам нужно добавить id для TextView в "activity\_main.xml", чтобы по клику на текст можно было переходить на другое activity

* Открываем "activity\_main.xml" и добавляем id у TextView. Нужно добавить android:id="@+id/first\_activity\_text\_view" как показано на картинке:



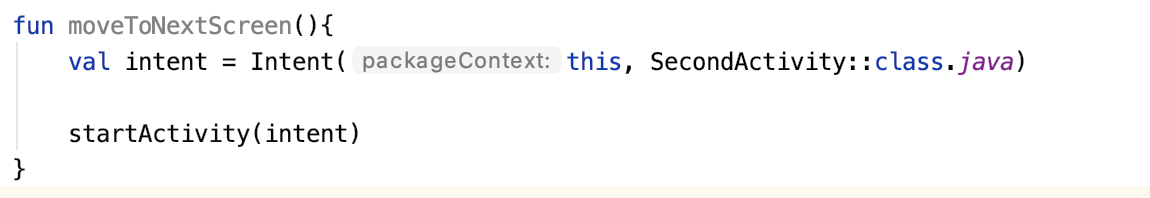
## Step 5

* Добавим обработчик нажатия на TextView в главном активити. Как добавлять можете глянуть или в лекции или на картинке внизу.
* Позже добавим функцию, которую будем вызывать в только что добавленном обработчике. Мы ее назвали "moveToNextScreen"



## Step 6

* Добавим логику перехода ко второму экрану. Для этого нужно создать объект Intent
* Передать в него текущий контекст - это наше текущее "MainActivity". А вторым параметром объект класса нашего, созданного activity в шаге 2.
* Осталось вызвать "startActivity" с этим intent в параметре:



## Step 7

* Откроем файл "activity\_second.xml" в ресурсной папке
* Содержимое этого файла заменим следующим кодом:

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

<androidx.constraintlayout.widget.ConstraintLayout

xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

tools:context=".SecondActivity">

<TextView

android:id="@+id/second\_activity\_text\_view"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="This is second activity"

app:layout\_constraintBottom\_toBottomOf="parent"

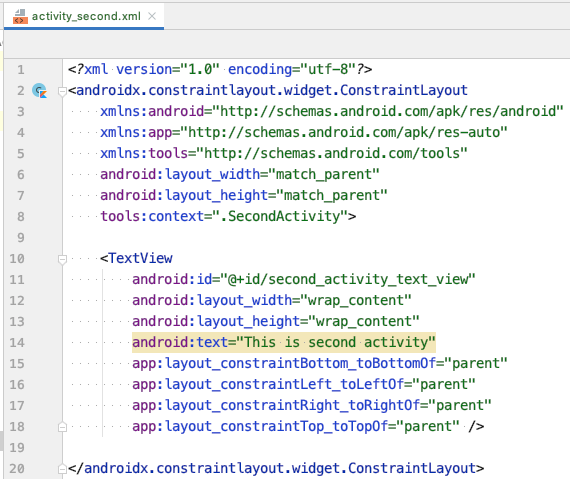
app:layout\_constraintLeft\_toLeftOf="parent"

app:layout\_constraintRight\_toRightOf="parent"

app:layout\_constraintTop\_toTopOf="parent" />

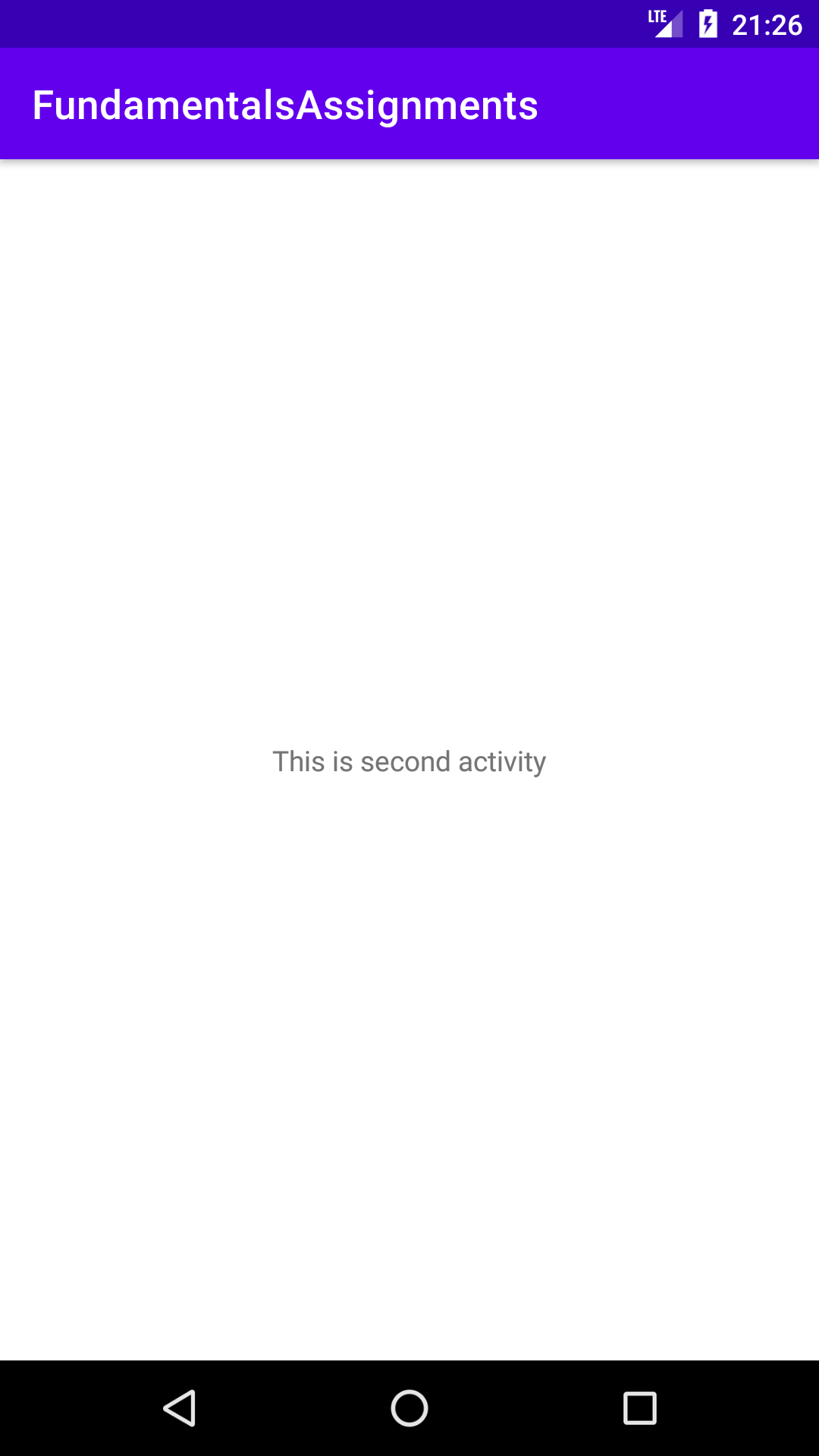
</androidx.constraintlayout.widget.ConstraintLayout>

Заметьте, что у TextView есть id: android:id="@+id/second\_activity\_text\_view"



## Step 9

* Запускаем проект, нажав на кнопку Run  
    
  
* После этого нажимаем на текст на экране и должны перейти на другой экран, где увидим следующее:



* Вернуться назад не предыдущий экран можно с помощью системной или хардварной кнопки "Back"

# 

# Assignment #6: Передача параметров между Activity

**Описание:**

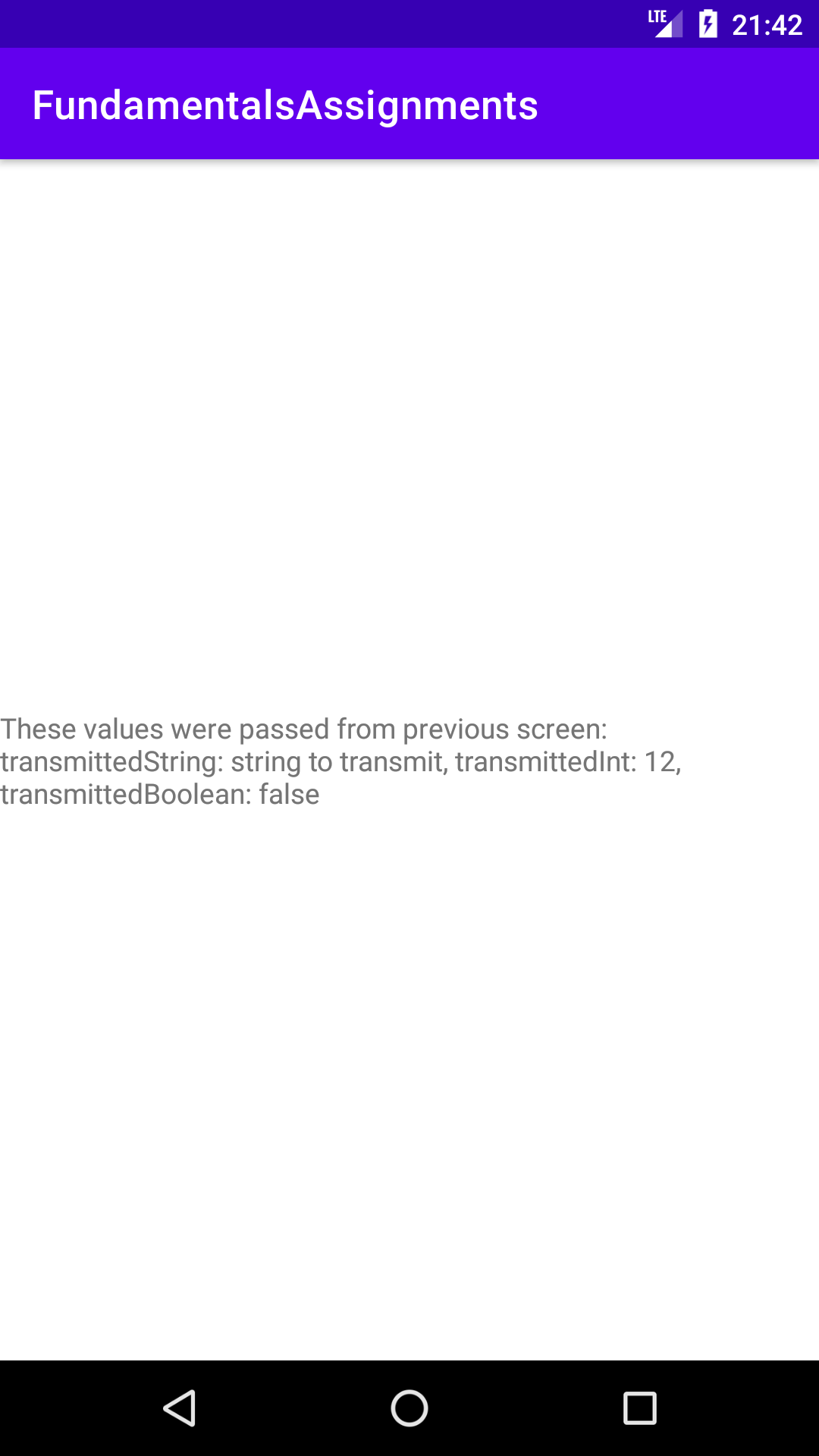
Объекты между активити можно передавать с помощью метода объекта Intent "*putExtra(key, value)"*. Value в extra может быть разных типов. В основном это примитивы либо *serializable/parcelable* объекты, а также массивы Int, строк.

**Задача:**

*Передать объекты от первого экрана на второй и распечатать эти значения в TextView*

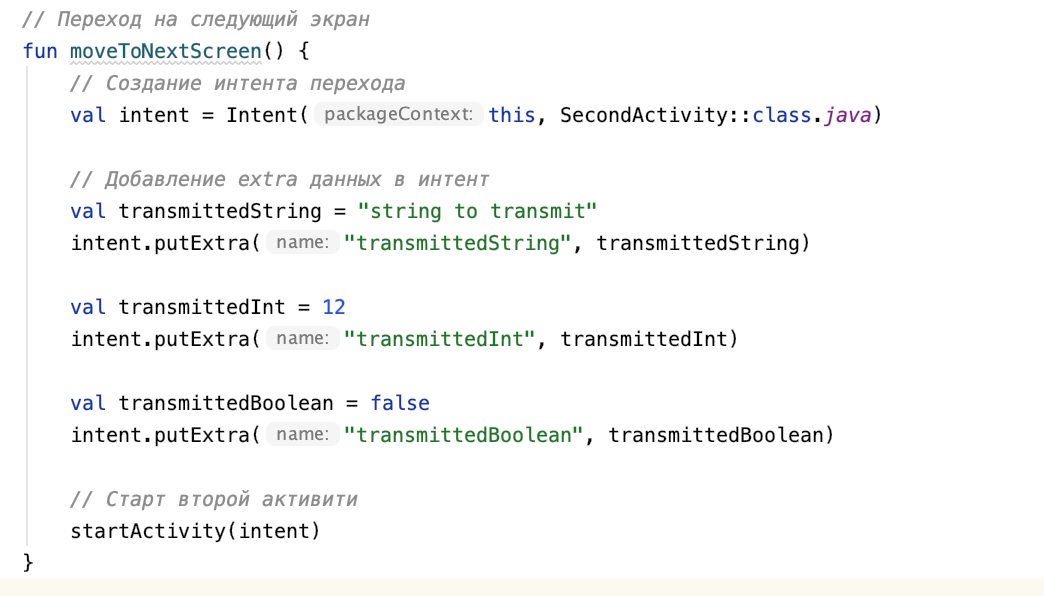
**Ответ:**

В конце задания после перехода на второй экран вы увидите следующее:

****

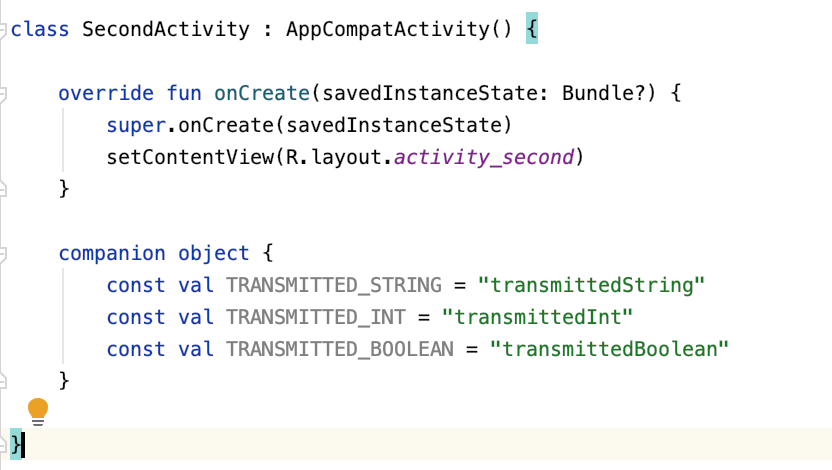
## Step 1

* Передадим три различных значения: Int, String и Boolean на второй экран. Чтобы передать значения на другой экран надо воспользоваться функцией putExtra у объекта intent. Она всегда принимаем ключ и значение
* Для String определим ключ: "transmittedString", а значение может быть любое. В нашем примере это "string to transmit"
* Для Int определим ключ: "transmittedInt", а значение может быть любое. В нашем примере это 12
* Для Boolean определим ключ: "transmittedBoolean", а значение может быть любое. В нашем примере это false
* Сверьтесь с картинкой ниже:



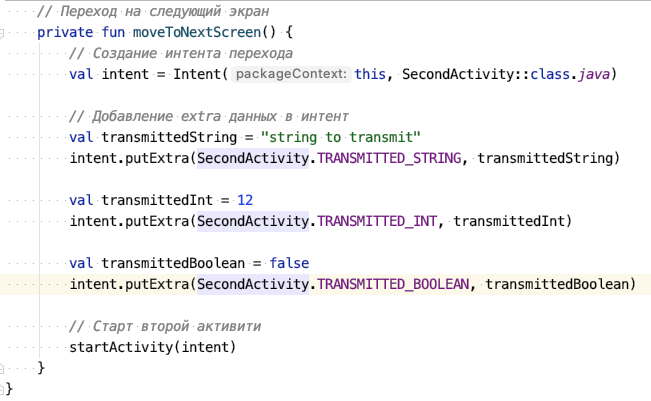
## Step 2

Для того, чтобы не запутаться в названиях ключей extra - лучше их выносить в константы. Давайте добавим эти строковые константы в "SecondActivity":



## Step 3

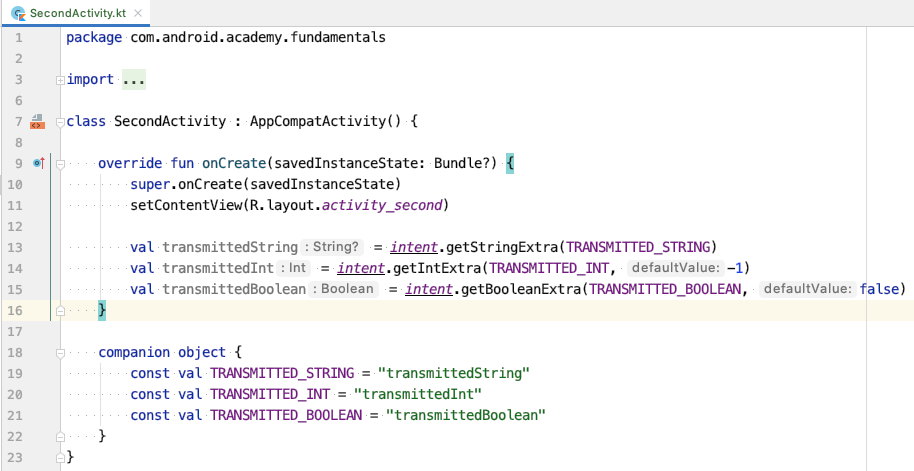
Заменим ключи в "MainActivity" на константы, определенные в шаге 2:



## Step 4

Теперь мы достаем переданные данные на стороне "SecondActivity".

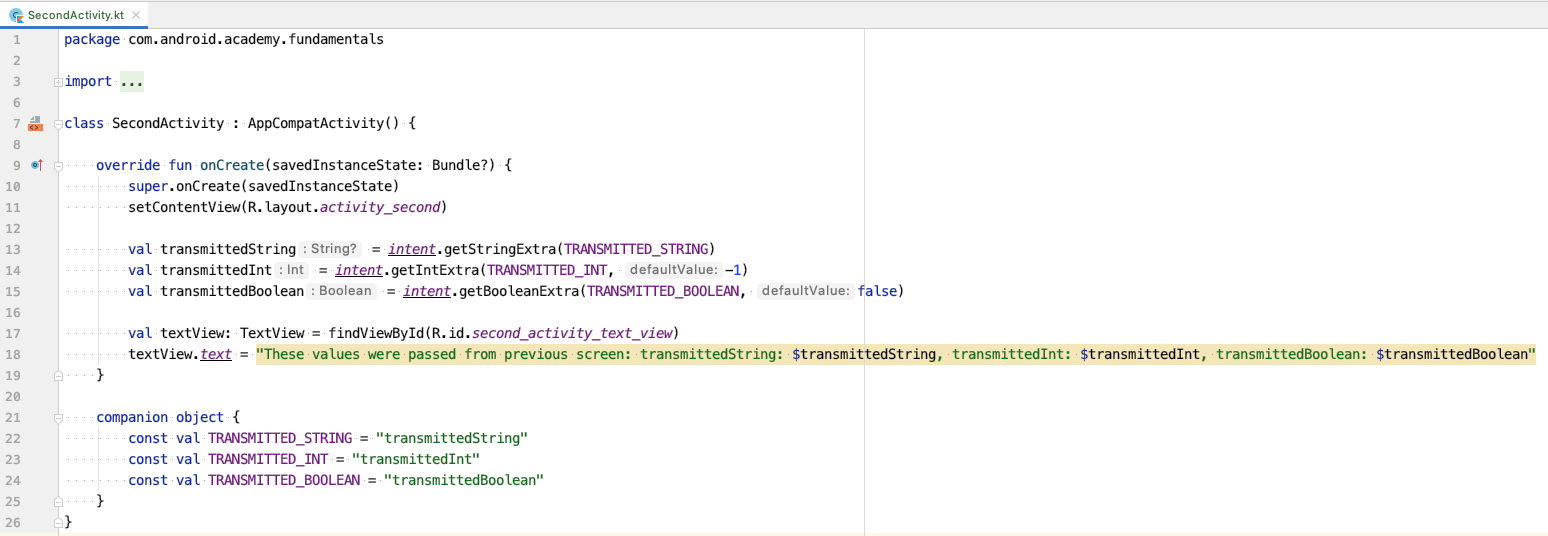
* В Activity есть метод getIntent, который возвращает объект intent, который мы создали ранее. Берем этот объект
* У этого объекта есть набор методов, чтобы получить те extra, что мы передали в шаге 1. Для получения каждого типа переданных данных существует соответствующий метод. Возьмем корректные методы для каждого типа данных:  
  для String - это метод getStringExtra  
  для Int - это метод getIntExtra  
  для Boolean - это метод getBooleanExtra



## Step 5

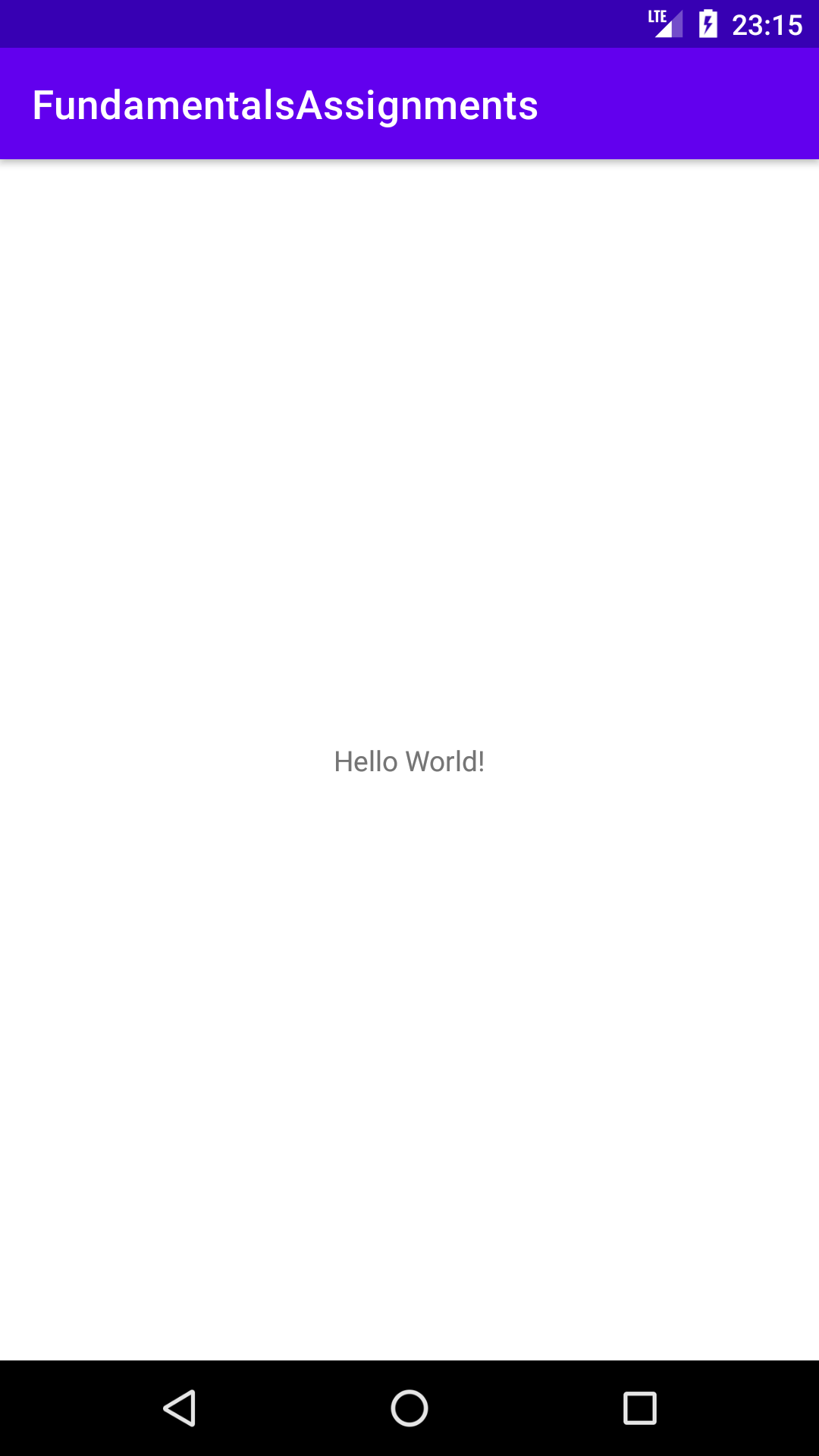
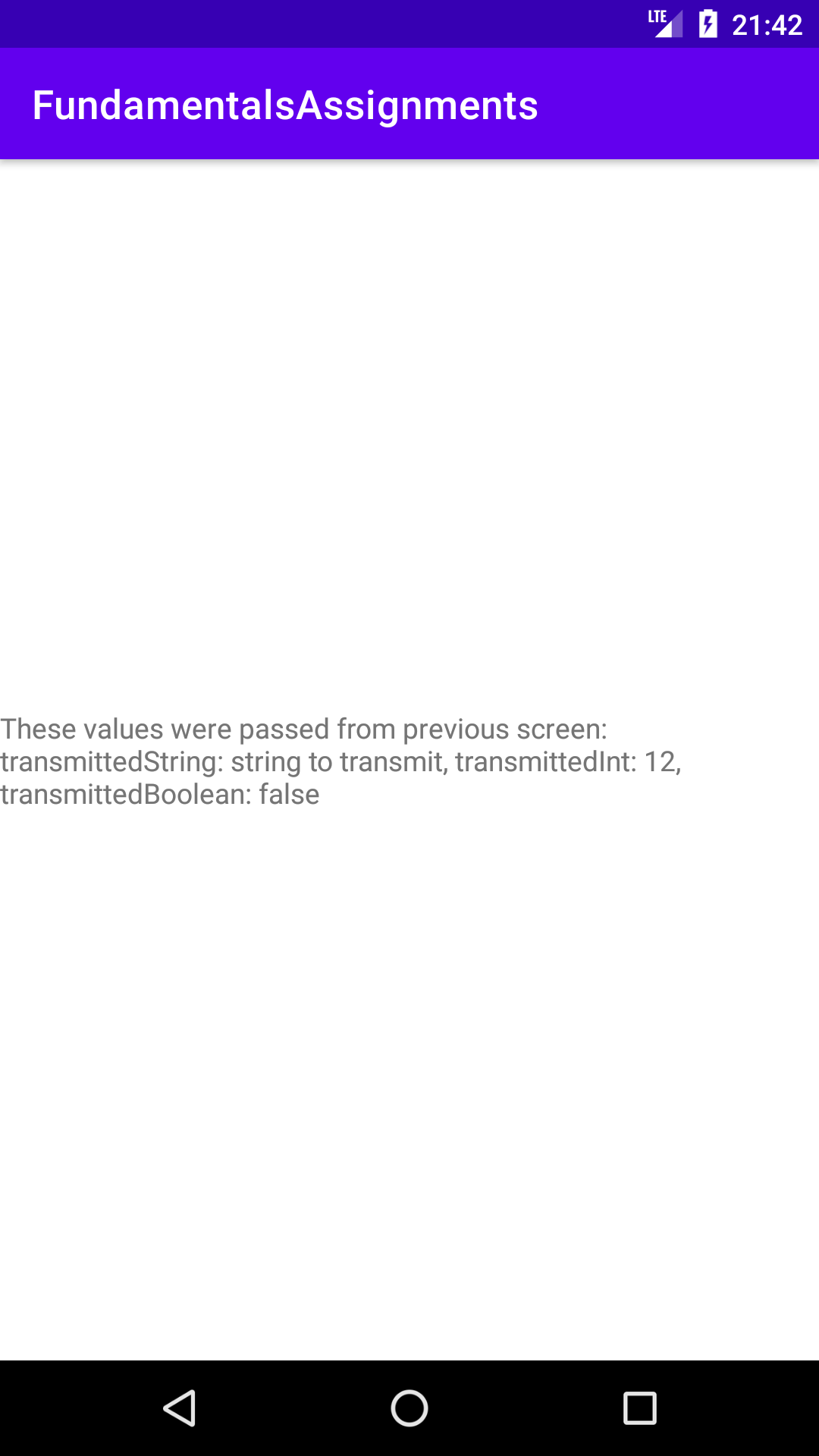
Теперь можем вывести эти значения в TextView второй активити. Для этого:

* Найдем наше второе TextView по id "R.id.*second\_activity\_text\_view*"
* Установим TextView значение, что достали в шаге 4 -   
  "These values were passed from previous screen: transmittedString: $transmittedString, transmittedInt: $transmittedInt, transmittedBoolean: $transmittedBoolean"



## Step 6

Запускаем приложение и проверяем его работу. Теперь данные с первого экрана передаются на второй. Ожидаемый результат достигнут:

-> Click "Hello World!"->****