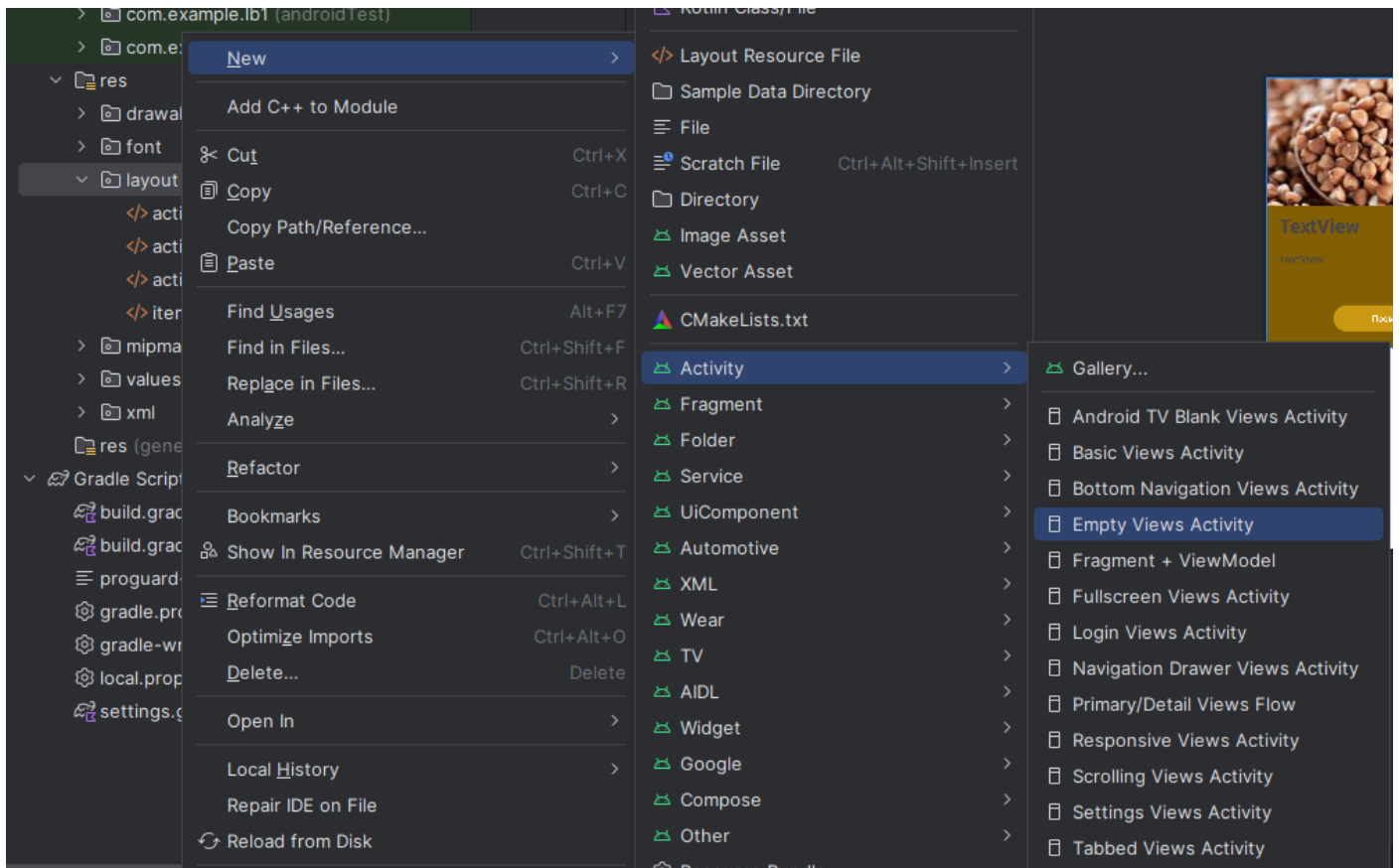


1. Добавляем новую страницу (Активности) на которую в последствии будем переходить при выборе определенной услуги/товара



Имя страницы указываем «ItemActivity»

2. В коде страницы activity\_item.xml:

- поменяем основной слой на LinearLayout;
- укажем ориентацию "vertical";
- задний фон как для всего проекта "#e2a600".

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout 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:id="@+id/main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:background="#e2a600"
    tools:context=".ItemActivity">

</LinearLayout>
```

3. Копируем два первых сверху TextView и кнопку button из активити item\_in\_list.xml и вставляем в activity\_item.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout 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:id="@+id/main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:background="#e2a600"
    tools:context=".ItemActivity">

    <TextView
        android:id="@+id/item_list_title"
        android:paddingHorizontal="20dp"
        android:paddingVertical="10dp"
        android:textStyle="bold"
        android:textSize="30sp"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="TextView" />

    <TextView
        android:id="@+id/item_list_desc"
        android:paddingHorizontal="20dp"
        android:paddingVertical="10dp"
        android:textSize="17sp"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="TextView" />

    <Button
        android:id="@+id/item_list_button"
        android:layout_width="200dp"
        android:layout_gravity="center"
        android:layout_height="wrap_content"
        android:backgroundTint="#CA9811"
        android:text="Посмотреть" />

</LinearLayout>
```

## 4. Редактируем кнопку TextView

### 4.1. Меняем текст на кнопки на «Купить»

### 4.2. Меняем id кнопки на button\_buy

```
<Button  
    android:id="@+id/button_buy"  
    android:layout_width="200dp"  
    android:layout_gravity="center"  
    android:layout_height="wrap_content"  
    android:backgroundTint="#CA9811"  
    android:text="Купить" />
```

### 4.3. Меня id TextView:

– верхнего на item\_list\_title\_one

– нижнего на item\_list\_text

На эту страницу будем переходить при нажатии на услугу или товар и в поля здесь будем динамически подставлять название товара и полный текст описания.

На этой странице дизайн немного поменять по своему вкусу – шрифты, цвета и прочее. Попробуйте поменять, не оставляйте дизайн как есть.

## 5. Переходим в ItemsAdapter

Надо создать переменную которая будет ссылаться на кнопку «Посмотреть» в item\_in\_list.xml.

Далее при нажатии на эту кнопку будем переходить на новую страницу куда будем передавать определенные данные и выводить их на странице. Эти данные будут разные – в зависимости от того на какую услугу/товар мы нажмем.

5.1. Создаем новую переменную на основе Button:

```
val btn: Button = view.findViewById(R.id.item_list_button)
```

Место создания переменной:

```
class ItemsAdapter (var items: List<Item>, var context: Context) :  
RecyclerView.Adapter<ItemsAdapter.MyViewHolder>(){  
    class MyViewHolder(view: View): RecyclerView.ViewHolder(view){  
        val image: ImageView = view.findViewById(R.id.item_list_image)  
        val title: TextView = view.findViewById(R.id.item_list_title)  
        val desc: TextView = view.findViewById(R.id.item_list_desc)  
        val price: TextView = view.findViewById(R.id.item_list_price)  
        val btn: Button = view.findViewById(R.id.item_list_button)  
    }  
}
```

В onBindViewHolder указываем: обращаемся к holder, к самой кнопке btn и для кнопки добавляем обработчик событий setOnClickListener:

```
holder.btn.setOnClickListener {
```

Здесь же создаем переменную intent, указываем что обращаемся к классу Intent и сообщаем – из того контекста который был сюда передан context (со страницы на которой мы находимся) будем переходить на страницу ItemActivity:

```
val intent = Intent(context, ItemActivity::class.java)
```

Указываем здесь же следующее: в эту новую активити будем передавать новые определенные данные:

```
intent.putExtra("itemTitle", items[position].title)
```

Функция putExtra принимает 2-а параметра – ключ “ itemTitle ” и значение.

Копируем тоже самое для полного описания:

```
intent.putExtra("itemText", items[position].text)
```

Также можно передать и ImageId и price.

Также обращаемся к контексту `context` и функции `startActivity` и передаем сам объект `intent`:

```
context.startActivity(intent)
```

В итоге `onBindViewHolder`:

```
override fun onBindViewHolder(holder: MyViewHolder, position: Int) {
    holder.title.text = items[position].title
    holder.desc.text = items[position].desc
    holder.price.text = items[position].price.toString() + "$"

    val imageId = context.resources.getIdentifier(
        items[position].image,
        "drawable",
        context.packageName
    )

    holder.image.setImageResource(imageId)

    holder.btn.setOnClickListener {
        val intent = Intent(context, ItemActivity::class.java)

        intent.putExtra("itemTitle", items[position].title)
        intent.putExtra("itemText", items[position].text)

        context.startActivity(intent)
    }
}
```

## 6. Переходим в ItemActivity

Будем подставлять те данные, которые получили через функцию putExtra

Создаем 2-е переменные:

```
val title: TextView = findViewById(R.id.item_list_title_one)
val text: TextView = findViewById(R.id.item_list_text)
```

Обращаемся к тексту и подставляем то что получили с самого перехода между страницами:

```
title.text = intent.getStringExtra("itemTitle")
text.text = intent.getStringExtra("itemText")
```

В итоге:

```
package com.example.lb1

import android.os.Bundle
import android.widget.TextView
import androidx.activity.enableEdgeToEdge
import androidx.appcompat.app.AppCompatActivity
import androidx.core.view.ViewCompat
import androidx.core.view.WindowInsetsCompat

class ItemActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        enableEdgeToEdge()
        setContentView(R.layout.activity_item)

        val title: TextView = findViewById(R.id.item_list_title_one)
        val text: TextView = findViewById(R.id.item_list_text)

        title.text = intent.getStringExtra("itemTitle")
        text.text = intent.getStringExtra("itemText")
    }
}
```

ТЕСТИРУЕМ