## Создание второго окна приложения

### Цель работы

Ознакомиться со способом создания и запуска второго окна в составе мобильного приложения.

### Контрольные вопросы

1. Зачем делить приложение на несколько окно? Почему нельзя использовать разные расположения?

Каждое окно наполнено своим содержанием и функционалом. Приложение необходимо делить на несколько окон, потому что обычно оно имеет несколько «состояний», в зависимости от которых показываются разные компоненты, реализована разная логика. Достаточно проблематично делать такую реализацию с помощью скрытия и перестановки компонентов на одном экране. Гораздо проще и логичнее разделить их в разные окна. К тому же пользователь легко сможет перемещаться по приложению, подобно тому, как он перемещается по страницам веб-сайта.

1. Что такое интент и зачем он нужен?

Intent представляет собой объект описания операции, которую необходимо выполнить через систему Android, то есть необходимо сначала описать некоторую операцию в виде объекта Intent, после чего отправить её на выполнение. Наиболее распространённый сценарий использования Intent — запуск другой Activity в своём приложении.

1. Как вызвать определенное окно своего приложение? А другого?

Вызвать определенное окно своего приложения:

Intent intent = new Intent(context, NewActivity.class);  
startActivity(intent);

Вызвать определенное окно другого приложения:

Intent intent = new Intent();

intent.setComponent(new ComponentName("com.example", "com.example.MyExampleActivity"));  
startActivity(intent);

1. Что такое таск? Почему при перемещении между окнами работает кнопка “Назад”?

Task — группа из нескольких Activity. Все Activity располагаются в стеке в том порядке, в котором они были вызваны. Если пользователь нажмет кнопку "Назад", то Activity, которая лежит на вершине стека, будет завершена и удалена из стека, а пользователь увидит Activity, которая была вызвана до вызова завершенной Activity.

### Дополнительные задания

1. Создайте приложение, состоящее из четырех активностей, и реализуйте переходы между ними.
2. Реализуйте переходы между активностями используя меню приложения. Меню должно быть описано в XML файле и быть общим для всех четырех активностей.
3. (\*) Реализуйте передачу данных между активностями. В первом окне создайте текстовое поле для ввода имени и кнопку. При тапе на кнопку должно открываться второе окно, в котором отображается имя, введенное пользователем.

Для задания 2 используются MainActivity, FirstActivity, SecondActivity и ThirdActivity.

Для задания 3 используются MainActivity и NameActivity.

MainActivity.java

package com.example.intents;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.content.Intent;  
import android.os.Bundle;  
import android.view.Menu;  
import android.view.MenuItem;  
import android.view.View;  
import android.widget.Button;  
import android.widget.EditText;  
  
public class MainActivity extends AppCompatActivity {  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 EditText editText = findViewById(R.id.*editText*);  
 Button button = findViewById(R.id.*button*);  
  
 button.setOnClickListener(  
 new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 String name = editText.getText().toString();  
 Intent intent = new Intent(MainActivity.this, NameActivity.class);  
 intent.putExtra("name", name);  
 startActivity(intent);  
 }  
 }  
 );  
 }  
  
 @Override  
 public boolean onCreateOptionsMenu(Menu menu) {  
 getMenuInflater().inflate(R.menu.*main\_menu*, menu);  
 return super.onCreateOptionsMenu(menu);  
 }  
  
 @Override  
 public boolean onOptionsItemSelected(MenuItem item) {  
 switch (item.getItemId()) {  
 case R.id.*firstActivity*:  
 startActivity(new Intent(MainActivity.this, FirstActivity.class));  
 break;  
 case R.id.*secondActivity*:  
 startActivity(new Intent(MainActivity.this, SecondActivity.class));  
 break;  
 case R.id.*thirdActivity*:  
 startActivity(new Intent(MainActivity.this, ThirdActivity.class));  
 break;  
 }  
 return super.onOptionsItemSelected(item);  
 }  
}

activity\_main.xml

*<?*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=".MainActivity">  
  
 <EditText  
 android:id="@+id/editText"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:ems="10"  
 android:inputType="textPersonName"  
 android:text="Имя"  
 app:layout\_constraintBottom\_toTopOf="@+id/button"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent" />  
  
 <Button  
 android:id="@+id/button"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Вввод"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
</androidx.constraintlayout.widget.ConstraintLayout>

NameActivity.java

package com.example.intents;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.content.Intent;  
import android.os.Bundle;  
import android.widget.TextView;  
  
public class NameActivity extends AppCompatActivity {  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_name*);  
  
 TextView textView = findViewById(R.id.*textView*);  
  
 Intent intent = getIntent();  
 textView.setText(intent.getStringExtra("name"));  
 }  
}

activity\_name.xml

*<?*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=".NameActivity">  
  
 <TextView  
 android:id="@+id/textView"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="TextView"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
</androidx.constraintlayout.widget.ConstraintLayout>

FirstActivity.java

package com.example.intents;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.content.Intent;  
import android.os.Bundle;  
import android.view.Menu;  
import android.view.MenuItem;  
  
public class FirstActivity extends AppCompatActivity {  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_first*);  
 }  
  
 @Override  
 public boolean onCreateOptionsMenu(Menu menu) {  
 getMenuInflater().inflate(R.menu.*main\_menu*, menu);  
 return super.onCreateOptionsMenu(menu);  
 }  
  
 @Override  
 public boolean onOptionsItemSelected(MenuItem item) {  
 switch (item.getItemId()) {  
 case R.id.*mainActivity*:  
 startActivity(new Intent(FirstActivity.this, MainActivity.class));  
 break;  
 case R.id.*secondActivity*:  
 startActivity(new Intent(FirstActivity.this, SecondActivity.class));  
 break;  
 case R.id.*thirdActivity*:  
 startActivity(new Intent(FirstActivity.this, ThirdActivity.class));  
 break;  
 }  
 return super.onOptionsItemSelected(item);  
 }  
}

activity\_first.xml

*<?*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=".FirstActivity">  
  
 <TextView  
 android:id="@+id/textView2"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="First Activity"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
</androidx.constraintlayout.widget.ConstraintLayout>

Аналогично SecondActivity.java и activity\_second.xml, ThirdActivity.java и activity\_third.xml.

Я предусмотрел, что переход из Activity в ту же самую Activity не совсем логичен, поэтому при нажатии на соответствующий пункт меню та Activity, на которой мы находимся, не вызывается — в принципе никаких действий при нажатии не происходит, но сам пункт, конечно, показывается, чтобы показать структуру нашего приложения.







