## Всплывающие уведомления

### Цель работы

Познакомится с сервисами Android Studio и научиться создавать приложения принимающие и отправляющие сообщения.

### Задания для выполнения

1. Создайте кнопку, которая будет выводить всплывающее сообщение на экран при нажатии с помощью объекта **Toast.**

### Методические указания

*Создание Toast-сообщения*

Всплывающее уведомление (Toast Notification) является сообщением, которое появляется на поверхности окна приложения, заполняя необходимое ему количество пространства, требуемого для сообщения. При этом текущая деятельность приложения остаётся работоспособной для пользователя. В течение нескольких секунд сообщение плавно закрывается. Всплывающее уведомление также может быть создано службой, работающей в фоновом режиме. Как правило, всплывающее уведомление используется для показа коротких текстовых сообщений.

Для создания всплывающего уведомления необходимо инициализировать объект Toast при помощи метода Toast.makeText(), а затем вызвать метод show() для отображения сообщения на экране:

Toast toast = Toast.makeText(getApplicationContext(),

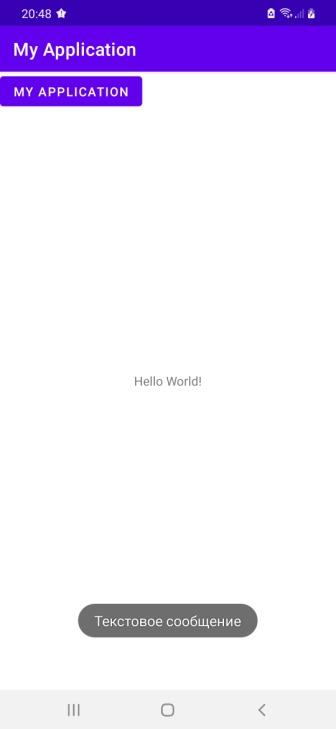
"Текстовое сообщение", Toast.LENGTH\_SHORT);

toast.show();

У метода makeText() есть три параметра:

* Контекст приложения
* Текстовое сообщение
* Продолжительность времени показа уведомления. Можно использовать только две константы

Уведомления выводятся на 3 с половиной секунды или на 2 секунды.



По умолчанию стандартное всплывающее уведомление появляется в нижней части экрана. Изменить место появления уведомления можно с помощью метода setGravity(int, int, int). Метод принимает три параметра:

* стандартная константа для размещения объекта в пределах большего контейнера (например, GRAVITY.CENTER, GRAVITY.TOP и др.);
* смещение по оси X
* смещение по оси Y

Например, если вы хотите, чтобы уведомление появилось в центре экрана, то используйте следующий код (до вызова метода show()):

toast.setGravity(Gravity.CENTER, 0, 0);

Для вывода в левом верхнем углу.

toast.setGravity(Gravity.TOP or Gravity.LEFT, 0, 0)

Если нужно сместить уведомление направо, то просто увеличьте значение второго параметра. Для смещения вниз нужно увеличить значение последнего параметра. Соответственно, для смещения вверх и влево используйте отрицательные значения.

### Контрольные вопросы

1. Чем сервисы отличаются от Activity?

Обычно Activity обрабатывает пользовательский интерфейс (UI) и взаимодействие с пользователем, в то время как сервис обрабатывает задачу на основе пользовательского ввода.

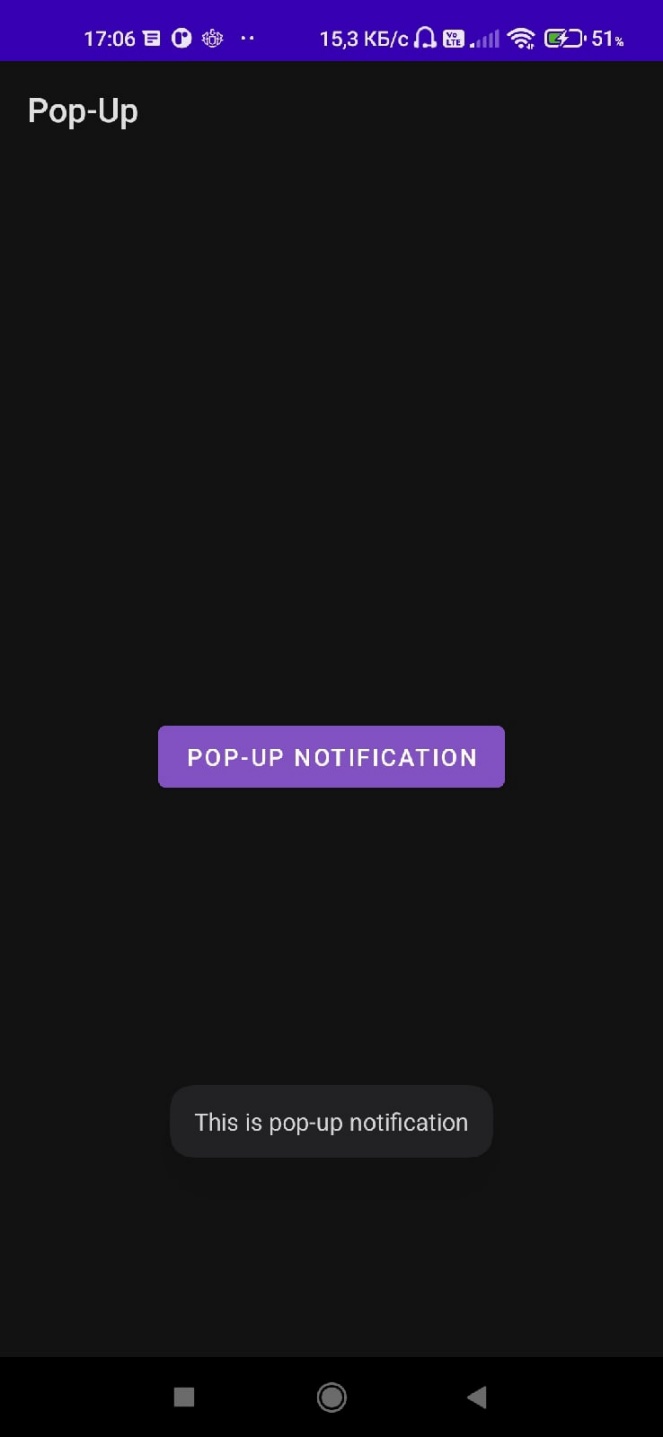
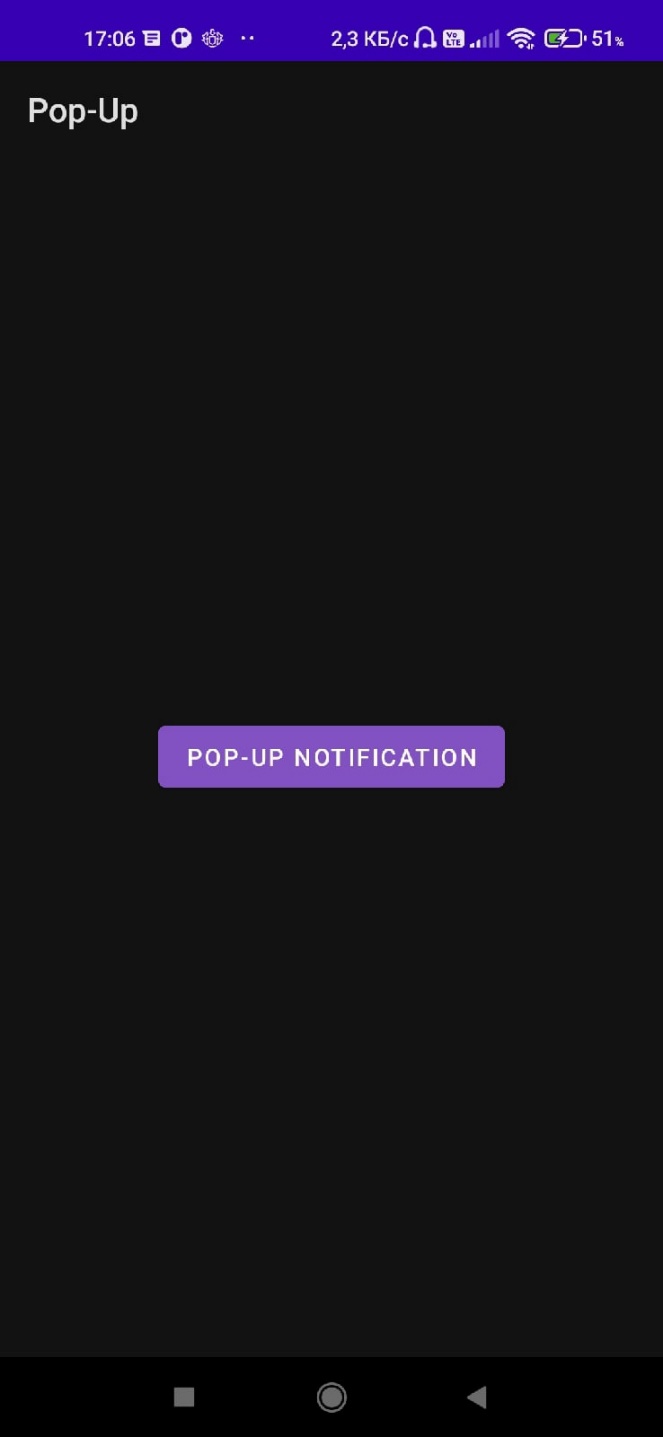
1. Возможно ли в Toast Notification разместить LinearLayout?

Да, так как Сам по себе стандартный внешний вид Toast состоит из контейнера LinearLayout, то в него так же как и в представлении активити можно разместить еще один контейнер.

1. Сколько времени отображается всплывшее сообщение созданное с помощью объекта Toast?

Уведомления выводятся на 3 с половиной секунды или на 2 секунды в зависимости от заданной длительности.

Выполнение задания:



MainActivity:

package *com.example.pop\_up*;  
  
import *androidx.appcompat.app.AppCompatActivity*;  
  
import *android.os.Bundle*;  
import *android.view.View*;  
import *android.widget.Toast*;  
  
public class *MainActivity* extends *AppCompatActivity* implements *View*.*OnClickListener* {  
  
 *@Override* protected void onCreate(*Bundle savedInstanceState*) {  
 super.onCreate(*savedInstanceState*);  
 setContentView(*R*.*layout*.activity\_main);  
 findViewById(*R*.*id*.btnPopUp).setOnClickListener(this);  
 }  
  
 *@Override* public void onClick(*View view*) {  
 *Toast*.*makeText*(this, "This is pop-up notification", *Toast*.LENGTH\_SHORT).show();  
 }  
}

<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:gravity="center"  
 android:orientation="vertical"  
 tools:context=".MainActivity">  
  
 <Button  
 android:id="@+id/btnPopUp"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="@string/pop\_up\_notification" />  
  
</LinearLayout>