МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ государственное БЮДЖЕТНОЕ

образовательное учреждение

высшего образования

«НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Кафедра автоматизированных систем управления

Лабораторная работа №5

по дисциплине: «Современные технологии разработки программного обеспечения»

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Выполнил:** |  | **Проверил:** |  | |
| Студент | *Чибисов И.А* | Преподаватель | *Эстрайх И.В.*  *ст. преподаватель* | |
| Факультет | *АВТФ* |  |  | |
| Направление (специальность) подготовки | *09.04.03 – Прикладная информатика* | Балл: \_\_\_\_\_\_\_\_\_ | |  |
| Группа | *АПМ-22* | Оценка *\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_* | | |
| Шифр | *010594214* |  |  | |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | | |
| подпись | | подпись | | |
| Дата сдачи: | «\_\_»\_\_\_\_\_\_\_\_\_\_20\_\_г. | Дата защиты: | «\_\_»\_\_\_\_\_\_\_\_\_\_20\_\_г | |

Новосибирск

2023

Цель работы: научиться создавать всплывающие уведомления для привлечения внимания пользователя во время работы приложения.

Задание на лабораторную работу:

1. Разработайте приложения на Java и Kotlin, создающие всплывающее текстовое уведомление, затем добавьте в них картинки.
2. Разработайте приложение на Java, создающее всплывающее уведомление с использованием вашего файла разметки.
3. Включите в отчет цель работы, коды программ, скриншоты с результатами и вывод.

Ход работы:

1. Сделаем проект на java и на Kotlin с уведомлением, первым сделаем java.

Создадим метод по вызову Тоста. Вот такой результат получим. Как на рисунке 1.

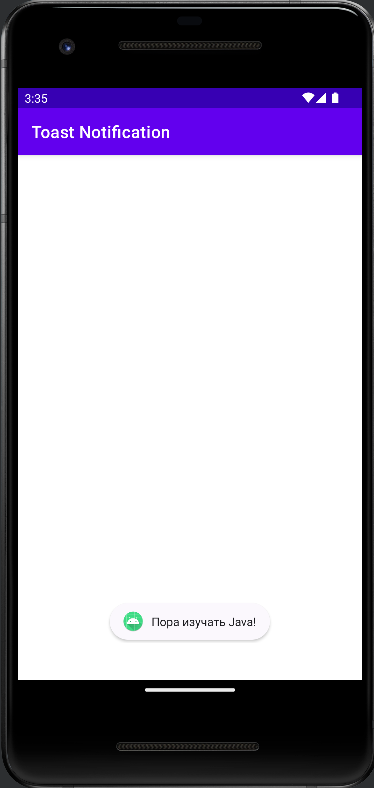


Рисунок 1 – Простой тост Java

Дополним метод, чтобы выводить и картинку, и текст. На рисунке 2 демонстрация.

//*TODO Метод для создание Тоста*

public void showToast1(View view) {  
 Toast toast = new Toast(getApplicationContext())**;** toast.setGravity(Gravity.*CENTER***,** 0**,** 0)**;** toast.setDuration(Toast.*LENGTH\_LONG*)**;** // создаем новый ImageView  
 ImageView imageView = new ImageView(getApplicationContext())**;** imageView.setImageResource(R.drawable.*image*)**;**

// создаем новый TextView для отображения текста  
 TextView textView = new TextView(getApplicationContext())**;** textView.setText("Java Учи, а лучше Go")**;** textView.setTextSize(TypedValue.*COMPLEX\_UNIT\_SP***,** 36)**;**

// создаем новый LinearLayout и добавляем туда ImageView и TextView  
 LinearLayout linearLayout = new LinearLayout(getApplicationContext())**;** linearLayout.setOrientation(LinearLayout.*VERTICAL*)**;** linearLayout.addView(imageView)**;** linearLayout.addView(textView)**;** // установка макета для LinearLayout  
 LinearLayout.LayoutParams layoutParams = new LinearLayout.LayoutParams(ViewGroup.LayoutParams.*MATCH\_PARENT***,** ViewGroup.LayoutParams.*WRAP\_CONTENT*)**;** linearLayout.setLayoutParams(layoutParams)**;** // установка LinearLayout в качестве представления для Toast  
 toast.setView(linearLayout)**;** toast.show()**;**}



Рисунок 2 – Java Toast с рисунком и текстом

Сделаем похожее приложение, но на kotlin.

Переписал метод который использовал для java, но изменил картирку и текст.

fun showToast1(view: View) {  
 val toast = Toast.makeText(*applicationContext***,** ""**,** Toast.*LENGTH\_LONG*)  
 toast.setGravity(Gravity.*CENTER***,** 0**,** 0)  
 // создаем новый ImageView  
 val imageView = ImageView(*applicationContext*)  
 imageView.setImageResource(R.drawable.*uganda*)  
// создаем новый TextView для отображения текста  
 val textView = TextView(*applicationContext*)  
 textView.*text* = "Kotlin топ, а а php класс!!!"  
 textView.setTextSize(TypedValue.*COMPLEX\_UNIT\_SP***,** 36F)  
// создаем новый LinearLayout и добавляем туда ImageView и TextView  
 val linearLayout = LinearLayout(*applicationContext*)  
 linearLayout.*orientation* = LinearLayout.*VERTICAL* linearLayout.addView(imageView)  
 linearLayout.addView(textView)  
  
// установка макета для LinearLayout  
 val layoutParams = LinearLayout.LayoutParams(ViewGroup.LayoutParams.*MATCH\_PARENT***,** ViewGroup.LayoutParams.*WRAP\_CONTENT*)  
 linearLayout.*layoutParams* = layoutParams  
// установка LinearLayout в качестве представления для Toast  
 toast.*view* = linearLayout  
 toast.show()  
 }

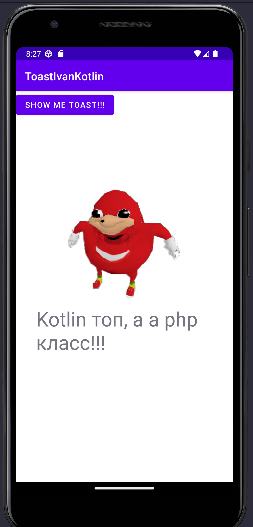


Рисунок 3 – Toast с рисунком и текстом kotlin

2 Дополним наш код Java и сделаем метод по вызову уведомления из custom\_layout. Метод представлен ниже, как рисунок.

//*TODO Метод для создание Тоста из xml*public void showToast2(View view) {  
 Toast notification = new Toast(getApplicationContext())**;** // Задаем продолжительность отображения уведомления  
 notification.setDuration(Toast.*LENGTH\_LONG*)**;** // Задаем макет для отображения уведомления  
 View layout = getLayoutInflater().inflate(R.layout.*custom\_layout***,** (ViewGroup) findViewById(R.id.*toast\_layout*))**;** notification.setView(layout)**;** // Отображаем уведомление  
 notification.show()**;**}



Рисунок 4 – Java Toast с готовым макетом

Вывод: Мы научились создавать всплывающие уведомления для привлечения внимания пользователя во время работы приложения, в kotlin и java.