

|  |
| --- |
| МИНОБРНАУКИ РОССИИ |
| Федеральное государственное бюджетное образовательное учреждениевысшего образования"МИРЭА - Российский технологический университет"РТУ МИРЭА |

Институт информационных технологий (ИТ)

Кафедра инструментального и прикладного программного обеспечения (ИППО)

**Лабораторная работа №1**

**по дисциплине**

**«Разработка программных приложений»**

**Тема**

**«Реализация приложения»**

Студент группы: ИКБО-16-18 Осадчук Г. М. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ *(Подпись студента)*

Руководитель работы Зорина Н.В.

*(Подпись преподавателя)*

Москва 2020

**Цель работы**

Цель данной лабораторной работы - получение практических навыков разработки приложения, изучение разработки под Android на языке Java, а также компонента RecyclerView.

**Выполнения лабораторной работы**

Задание:

Реализовать приложение, состоящее из двух активити:

Первое активити - это splash screen. Полноэкранное без ActionBar, c одной картинкой (или анимацией по желанию) по центру. Активити показывается 2 секунды, потом запускает второе и первое «умирает». Ожидание необходимо реализовать через Thread.sleep.

Второе активити представляет из себя список из 1000000 (один миллион) элементов, где четные элементы имеют серый фон (#CCCCCC), а нечетные — белый (#FFFFFF). Каждый элемент содержит картинку и текст. Картинка может быть любая, и она не меняется для элементов (выбирайте маленькую). Текст каждого элемента — это его индекс текстом (пример «cто двенадцать»). Отсчет начинается с единицы. Шрифт большой.

**Решение**

Весь код можно найти по [ссылке](https://github.com/EVOLUTION666/Android-App-Laba1).

**MainActivity**

|  |
| --- |
| import androidx.appcompat.app.AppCompatActivity; import androidx.recyclerview.widget.RecyclerView;  import android.os.Bundle;  public class MainActivity extends AppCompatActivity {   @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.*activity\_main*);  RecyclerView recyclerView = findViewById(R.id.*recycler\_view*);  RecyclerAdapter adapter = new RecyclerAdapter(this);  recyclerView.setAdapter(adapter);  } } |

**SplashScreenActivity**

|  |
| --- |
| import android.app.Activity; import android.os.Bundle; import android.content.Intent; import java.util.concurrent.TimeUnit;  public class SplashScreenActivity extends Activity {   @Override  public void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.*splash\_screen*);  Thread thread = new Thread() {  @Override  public void run() {  try {  TimeUnit.*SECONDS*.sleep(2);  Intent intent = new Intent(SplashScreenActivity.this, MainActivity.class);  startActivity(intent);  finish();  } catch (InterruptedException e) {  e.printStackTrace();  }   }  };  ThreadHandler handler = ThreadHandler.*createInstance*(thread);  } } |

**RecyclerAdapter**

|  |
| --- |
| import android.content.Context; import android.view.LayoutInflater; import android.view.View; import android.view.ViewGroup; import android.widget.ImageView; import android.widget.LinearLayout; import android.widget.TextView; import androidx.annotation.NonNull; import androidx.recyclerview.widget.RecyclerView;  public class RecyclerAdapter extends RecyclerView.Adapter<RecyclerAdapter.ViewHolder>{  LayoutInflater inflater;   public class ViewHolder extends RecyclerView.ViewHolder {  final ImageView imageView;  final TextView text;  final LinearLayout layout;  public ViewHolder(@NonNull View itemView) {  super(itemView);  layout = itemView.findViewById(R.id.*layout*);  imageView = itemView.findViewById(R.id.*img*);  text = itemView.findViewById(R.id.*text*);  }  }   public RecyclerAdapter(Context context) {  this.inflater = LayoutInflater.*from*(context);  }   @NonNull  @Override  public RecyclerAdapter.ViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {  View view = inflater.inflate(R.layout.*item*, parent, false);  return new ViewHolder(view);  }   @Override  public void onBindViewHolder(@NonNull RecyclerAdapter.ViewHolder holder, int position) {  holder.text.setText(IntRetranslator.*retranslate*(position+1));  if((position+1) %2 == 0)  holder.layout.setBackgroundColor(0xCCCCCCCC);  else  holder.layout.setBackgroundColor(0xFFFFFFFF);  }   @Override  public int getItemCount() {  return 1000000;  } } |

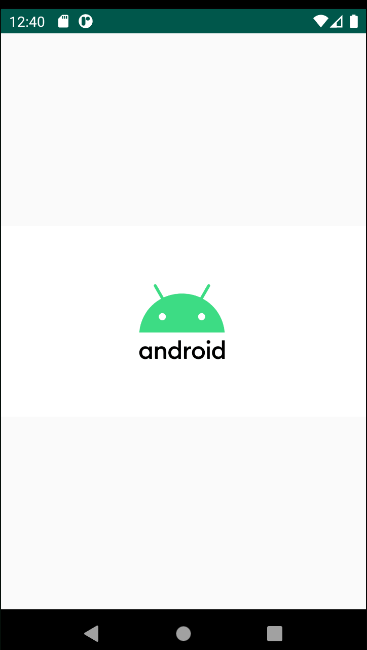
**IntRetranslator**

|  |
| --- |
| public class IntRetranslator {  static String [][] toHundred = { {"","од","дв","три","четыре","пять","шесть","семь","восемь","девять"},  {"", "десять " ,"двадцать ","тридцать ","сорок ","пятьдесят ","шестьдесят ","семьдесят ","восемьдесят ","девяносто "},  {"","сто ","двести ","триста ","четыреста ","пятьсот ","шестьсот ","семьсот ","восемьсот ","девятьсот "} };  static String[] elevenToNineteen = {"десять ","одиннадцать ","двенадцать ","тринадцать ","четырнадцать ","пятнадцать ","шестнадцать ",  "семнадцать ","восемнадцать ","девятнадцать "};  static String[][] thousandsAndMillions = {{"", "", "", ""},  {"миллиардов ", "миллионов ", "тысяч ", ""},  {"миллиард ", "миллион ", "тысяча ", ""},  {"миллиарда ", "миллиона ", "тысячи ", ""},  {"миллиардов ", "миллионов ", "тысяч ", ""}};  public static String retranslate(long number) {  String text = "";  if(number == 0)  {  return "ноль";  }  int million = (int) (number)/ 1000000 ;  int thousand = (int) (number-(million\*1000000)) / 1000;  int lasts = (int) (number % 1000);  return text + toThousands (million , 1)+toThousands(thousand , 2)+toThousands(lasts , 3);  }  private static String toThousands(int value, int index) {  int hundreds = value/100;  int decimal = (value - (hundreds\*100)) / 10;  int units = value % 10;  String text = "";  if ( decimal == 1 ) text = toHundred [2] [hundreds] + elevenToNineteen [units];  else text = toHundred [2] [hundreds] + toHundred [1][decimal] + toHundred [0] [units];   // формируем окончания в единицах  if (index == 2) {if (units == 1 && decimal != 1) text = text + "на ";  else if (units == 2 & decimal != 1) text = text + "е ";  if (units > 1 && decimal != 1) text = text + " ";}  else {if (units == 1 && decimal != 1) text = text + "ин ";  if (units == 2 & decimal != 1) {text = text + "а ";}  else if (units != 0 & decimal != 1) text = text + " ";}   // дописываем степень числа  int indexA = 0;  if (value != 0 ) {  if (units == 0 || decimal == 1 ) indexA = 1;  else if (units == 1) indexA = 2;  else if (units > 1 & units < 5) indexA = 3;  else indexA = 4;}  text = text + thousandsAndMillions [indexA][index];  return text;  }   public static void main(String[] args) {  System.out.println(IntRetranslator.retranslate(1000000));  } } |

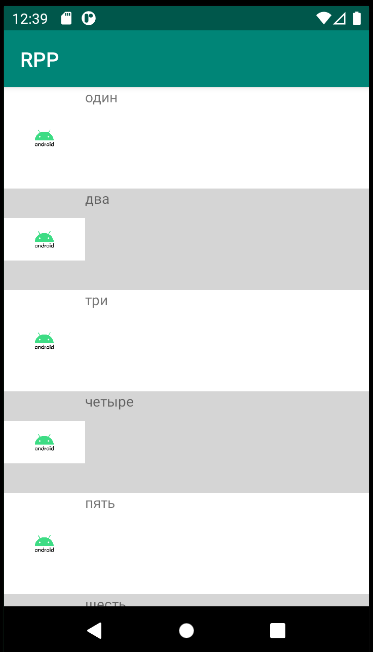
**RecyclerAdapter**

|  |
| --- |
| import android.content.Context; import android.view.LayoutInflater; import android.view.View; import android.view.ViewGroup; import android.widget.ImageView; import android.widget.LinearLayout; import android.widget.TextView; import androidx.annotation.NonNull; import androidx.recyclerview.widget.RecyclerView;  public class RecyclerAdapter extends RecyclerView.Adapter<RecyclerAdapter.ViewHolder>{  LayoutInflater inflater;   public class ViewHolder extends RecyclerView.ViewHolder {  final ImageView imageView;  final TextView text;  final LinearLayout layout;  public ViewHolder(@NonNull View itemView) {  super(itemView);  layout = itemView.findViewById(R.id.*layout*);  imageView = itemView.findViewById(R.id.*img*);  text = itemView.findViewById(R.id.*text*);  }  }   public RecyclerAdapter(Context context) {  this.inflater = LayoutInflater.*from*(context);  }   @NonNull  @Override  public RecyclerAdapter.ViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {  View view = inflater.inflate(R.layout.*item*, parent, false);  return new ViewHolder(view);  }   @Override  public void onBindViewHolder(@NonNull RecyclerAdapter.ViewHolder holder, int position) {  holder.text.setText(IntRetranslator.*retranslate*(position+1));  if((position+1) %2 == 0)  holder.layout.setBackgroundColor(0xCCCCCCCC);  else  holder.layout.setBackgroundColor(0xFFFFFFFF);  }   @Override  public int getItemCount() {  return 1000000;  } } |

**Результат работы программы:**



**Рис 1. Активити 1**



**Рис 2. Активити 2**

**Выводы по работе:**

В течение данной лабораторной работы был изучен процесс разработки под Android, а также компонент RecyclerView.

**Используемая литература:**

1. <https://developer.android.com/guide/topics/ui/layout/recyclerview?hl=ru>