МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ

КАФЕДРА ПРОГРАМНОЇ ІНЖЕНЕРІЇ

Звіт

з практичної роботи №3

Виконав: Перевірив:

ст. гр. ПЗПІ-23-2 ст. викладач кафедри ПІ

Семьонов Олег Сокорчук Ігор Петрович

Харків 2024

**Мета заняття**

Ознайомитися з основними типами діалогових панелей в Android.

Навчитися створювати та використовувати діалогові панелі в Android-застосунках.

Засвоїти методи для налаштування діалогових панелей та їх взаємодії з користувачем.

Засвоїти концепцію черги повідомлень (Message Queue) в Android.

Вивчити роботу з обробником повідомлень (Handler) для взаємодії з потоками.

Навчитися створювати та використовувати Handler для асинхронної обробки даних.

Навчитись створювати і налаштовувати списки з прогортанням (RecyclerView Adapter)

**Хід роботи**

1.1 Створимо проект з шаблоном Empty Views Activity

1.2 Створимо діалогову панель з типом Alert Dialog (діалогове вікно з можливістю вибору між кількома варіантами або підтвердження певної дії.). Для цього створимо кнопку, яка ініціюватиме показ діалогової панелі:

<Button  
 android:id="@+id/showDialogButton"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="176dp"  
 android:text="Show Dialog"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />

Тепер ініціалізуємо подію за допомогою слухача setOnClickListener:

Button showDialogButton = findViewById(R.id.showDialogButton);  
showDialogButton.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 new AlertDialog.Builder(MainActivity.this)  
 .setTitle("Dialog")  
 .setMessage("Example of AlertDialog")  
 .setPositiveButton("Ok", new DialogInterface.OnClickListener() {  
 @Override  
 public void onClick(DialogInterface dialog, int which) {  
  
 }  
 })  
 .setNegativeButton("Cancel", new DialogInterface.OnClickListener() {  
 @Override  
 public void onClick(DialogInterface dialog, int which) {  
  
 }  
 })  
 .show();  
 }  
});

1.3 Аналогічно створимо діалогову панель DatePickerDialog (дозволяє користувачу вибрати дату). Зробимо кнопки для натиснення:

<Button  
 android:id="@+id/showDatePicker"  
 android:layout\_width="169dp"  
 android:layout\_height="63dp"  
 android:text="DatePickerDialog"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/showDialogButton"  
 app:layout\_constraintVertical\_bias="0.257" />

Та ініціалізуємо подію за допомогою слухача:

Button showDatePicker = findViewById(R.id.showDatePicker);  
showDatePicker.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 DatePickerDialog datePickerDialog = new DatePickerDialog(MainActivity.this,  
 new DatePickerDialog.OnDateSetListener() {  
 @Override  
 public void onDateSet(DatePicker view, int year, int month, int dayOfMonth) {  
  
 }  
 }, 2024, 11 ,1 );  
 datePickerDialog.show();  
 }  
});

1.4 Створимо кастомний діалог, який і налаштуємо  
Для цього зробимо окрему розмітку і застилізуємо її (див. ДОДАТОК А)

Аналогічно створимо кнопку для виклику діалогу:

<Button  
 android:id="@+id/showCustomDialog"  
 android:layout\_width="180dp"  
 android:layout\_height="63dp"  
 android:layout\_marginTop="88dp"  
 android:text="Custom Dialog"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/showDatePicker" />

І прив’яжимо обробник подій:

Button showCustomDialog = findViewById(R.id.showCustomDialog);  
showCustomDialog.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 LayoutInflater inflater = getLayoutInflater();  
 View dialogView = inflater.inflate(R.layout.custom\_dialog, null);  
 AlertDialog.Builder builder = new AlertDialog.Builder(MainActivity.this);  
 builder.setView(dialogView)  
 .setPositiveButton("OK", new DialogInterface.OnClickListener() {  
 @Override  
 public void onClick(DialogInterface dialog, int which) {  
  
 }  
 })  
 .setNegativeButton("Cancel", new DialogInterface.OnClickListener() {  
 @Override  
 public void onClick(DialogInterface dialog, int which) {  
  
 }  
 });  
 builder.create().show();  
 }  
});

З першою підчастиною практичного заняття закінчено.

2.1 Створимо проект з шаблоном Empty Views Activity

2.2 Створимо простий приклад з Handler

Додамо кнопку і текстове поле у файлі розмітки:

<TextView  
 android:id="@+id/handlerMessageTextView1"  
 android:layout\_width="223dp"  
 android:layout\_height="47dp"  
 android:gravity="center"  
 android:text="Message 1 will appear here"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.023" />  
  
<Button  
 android:id="@+id/startHandlerButton1"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="80dp"  
 android:text="Start Handler 1"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.498"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />

Cтворимо об’єкт Handler:

Handler handler1 = new Handler(Looper.getMainLooper());  
Button startHandlerButton = findViewById(R.id.startHandlerButton1);  
startHandlerButton.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 handler1.postDelayed(new Runnable() {  
 @Override  
 public void run() {  
 TextView textView = findViewById(R.id.handlerMessageTextView1);  
 textView.setText("Handler executed after delay");  
 }  
 }, 2000);  
 }  
});

При натисканні на кнопку текст в текстовому полі оновиться через 2 секунди.

2.3 Взаємодія між потоками за допомогою Handler

Створимо нове текстове поле для другого прикладу:

<TextView  
 android:id="@+id/handlerMessageTextView2"  
 android:layout\_width="223dp"  
 android:layout\_height="47dp"  
 android:layout\_marginTop="24dp"  
 android:gravity="center"  
 android:text="Message 2 will appear here"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/startHandlerButton1" />

І додамо Handler який оновлює поле з фонового потоку:

Handler handler2 = new Handler(Looper.getMainLooper());  
new Thread(new Runnable() {  
 @Override  
 public void run() {  
 try {  
 Thread.sleep(3000);  
 } catch (InterruptedException e){  
 e.printStackTrace();  
 }  
 handler2.post(new Runnable() {  
 @Override  
 public void run() {  
 TextView textView = findViewById(R.id.handlerMessageTextView2);  
 textView.setText("Updated from background thread");  
 }  
 });  
 }  
}).start();

2.4 Відправка повідомлень через Handler

Створимо ще одне текстове поле (аналогічне першому) та реалузємо відправку через Handler:

Handler handler3 = new Handler(Looper.getMainLooper()){  
 @Override  
 public void handleMessage(Message msg){  
 TextView textView = findViewById(R.id.handlerMessageTextView3);  
 textView.setText("Message received: " + msg.what);  
 }  
};  
new Thread(new Runnable() {  
 @Override  
 public void run() {  
 try {  
 Thread.sleep(2000);  
 }  
 catch (InterruptedException e){  
 e.printStackTrace();  
 }  
 Message msg = handler3.obtainMessage();  
 msg.what = 1;  
 handler3.sendMessage(msg);  
 }  
}).start();

Отже, ми дізнались як можна використовувати Handler для взаємодії між потоками

3.1 Створимо проект з шаблоном Empty Views Activity

3.2 Створимо адаптер для нашого списку (Клас, який відповідає за постачання даних до RecyclerView та створення необхідних представлень (views) для кожного елемента даних.), який використовує допоміжний клас ViewHolder (Паттерн проектування, який використовується для зберігання посилань на елементи, що відображаються на екрані, що дозволяє уникнути повторних викликів методу findViewById()):

public class MyAdapter extends RecyclerView.Adapter<MyAdapter.ViewHolder> {  
 private String[] mData;  
 private int[] mImageData;  
  
 public MyAdapter(String[] data, int[] imageData){  
 mData = data;  
 mImageData = imageData;  
 }  
  
 public static class ViewHolder extends RecyclerView.ViewHolder {  
 public TextView textView;  
 public ImageView imageView;  
  
 public ViewHolder(View v) {  
 super(v);  
 textView = v.findViewById(R.id.textView);  
 imageView = v.findViewById(R.id.imageView);  
 }  
 }  
 @NonNull  
 @Override  
 public MyAdapter.ViewHolder onCreateViewHolder(ViewGroup parent, int viewType) {  
 View v = LayoutInflater.from(parent.getContext())  
 .inflate(R.layout.list\_item, parent, false);  
 return new ViewHolder(v);  
 }  
  
 @Override  
 public void onBindViewHolder(ViewHolder holder, int position) {  
 holder.textView.setText(mData[position]);  
 holder.imageView.setImageResource(mImageData[position]);  
 holder.itemView.setOnClickListener(v -> {  
 Toast.makeText(v.getContext(),"Clicked: "  
 + mData[position],Toast.LENGTH\_SHORT).show();  
 });  
 }  
  
 @Override  
 public int getItemCount() {  
 return mData.length;  
 }  
}

3.3 Створимо розмітку елементу списка:

<?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">  
  
 <TextView  
 android:id="@+id/textView"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="192dp"  
 android:text="TextView"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.498"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
 <ImageView  
 android:id="@+id/imageView"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="52dp"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/textView"  
 app:srcCompat="@drawable/ic\_launcher\_background" />  
</androidx.constraintlayout.widget.ConstraintLayout>

3.4 Створимо масиви з текста та картинок у основному класі для імітації елементів у списку:

String[] myDataset = {"Item1", "Item2", "Item3","Item4","Item5"};  
int[] myImage = {R.drawable.ic\_launcher\_background, R.drawable.ic\_launcher\_foreground,  
 R.drawable.ic\_launcher\_background, R.drawable.ic\_launcher\_foreground,  
 R.drawable.ic\_launcher\_background};  
RecyclerView recyclerView = findViewById(R.id.recyclerView);  
recyclerView.setLayoutManager(new LinearLayoutManager(this));  
MyAdapter adapter = new MyAdapter(myDataset,myImage);  
recyclerView.setAdapter(adapter);

Висновки:  
Під час практичної роботи було ознайомлено з діалоговими панелями, обробкою потоків та списками прогортування

Відеозвіт. URL: <https://youtu.be/IipvSY9n0SI>

Використані джерела:

1. Методичні вказівки до практичних занять: <https://dl.nure.ua/mod/page/view.php?id=599587>

2. Dialogs : <https://developer.android.com/develop/ui/views/components/dialogs?hl=ru>

3. Handler: <https://developer.android.com/reference/android/os/Handler>

4. RecyclerView: <https://developer.android.com/reference/androidx/recyclerview/widget/RecyclerView>

ДОДАТОК А (код першого підзавдання)

<?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:id="@+id/main"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".MainActivity">  
  
 <Button  
 android:id="@+id/showDialogButton"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="176dp"  
 android:text="Show Dialog"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
 <Button  
 android:id="@+id/showDatePicker"  
 android:layout\_width="169dp"  
 android:layout\_height="63dp"  
 android:text="DatePickerDialog"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/showDialogButton"  
 app:layout\_constraintVertical\_bias="0.257" />  
  
 <Button  
 android:id="@+id/showCustomDialog"  
 android:layout\_width="180dp"  
 android:layout\_height="63dp"  
 android:layout\_marginTop="88dp"  
 android:text="Custom Dialog"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/showDatePicker" />  
</androidx.constraintlayout.widget.ConstraintLayout>

<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:orientation="vertical"  
 android:padding="16dp">  
  
 <TextView  
 android:id="@+id/customDialogText"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:textSize="20sp"  
 android:layout\_gravity="center"  
 android:text="Individual dialog panel" />  
  
 <EditText  
 android:id="@+id/customDialogInput"  
 android:layout\_width="match\_parent"  
 android:layout\_height="50dp"  
 android:hint="Type text"  
 android:inputType="text"  
 android:text="" />  
</LinearLayout>

package com.nuresemonovoleh.android\_pzpi\_23\_2\_semonov\_oleh\_practtask3\_1;  
  
import android.app.DatePickerDialog;  
import android.content.DialogInterface;  
import android.os.Bundle;  
import android.view.LayoutInflater;  
import android.view.View;  
import android.widget.Button;  
import android.widget.DatePicker;  
  
import androidx.activity.EdgeToEdge;  
import androidx.appcompat.app.AlertDialog;  
import androidx.appcompat.app.AppCompatActivity;  
import androidx.core.graphics.Insets;  
import androidx.core.view.ViewCompat;  
import androidx.core.view.WindowInsetsCompat;  
  
public class MainActivity extends AppCompatActivity {  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 EdgeToEdge.enable(this);  
 setContentView(R.layout.activity\_main);  
 ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main), (v, insets) -> {  
 Insets systemBars = insets.getInsets(WindowInsetsCompat.Type.systemBars());  
 v.setPadding(systemBars.left, systemBars.top, systemBars.right, systemBars.bottom);  
 return insets;  
 });  
  
 Button showDialogButton = findViewById(R.id.showDialogButton);  
 showDialogButton.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 new AlertDialog.Builder(MainActivity.this)  
 .setTitle("Dialog")  
 .setMessage("Example of AlertDialog")  
 .setPositiveButton("Ok", new DialogInterface.OnClickListener() {  
 @Override  
 public void onClick(DialogInterface dialog, int which) {  
  
 }  
 })  
 .setNegativeButton("Cancel", new DialogInterface.OnClickListener() {  
 @Override  
 public void onClick(DialogInterface dialog, int which) {  
  
 }  
 })  
 .show();  
 }  
 });  
  
  
 Button showDatePicker = findViewById(R.id.showDatePicker);  
 showDatePicker.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 DatePickerDialog datePickerDialog = new DatePickerDialog(MainActivity.this,  
 new DatePickerDialog.OnDateSetListener() {  
 @Override  
 public void onDateSet(DatePicker view, int year, int month, int dayOfMonth) {  
  
 }  
 }, 2024, 11 ,1 );  
 datePickerDialog.show();  
 }  
 });  
  
 Button showCustomDialog = findViewById(R.id.showCustomDialog);  
 showCustomDialog.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 LayoutInflater inflater = getLayoutInflater();  
 View dialogView = inflater.inflate(R.layout.custom\_dialog, null);  
 AlertDialog.Builder builder = new AlertDialog.Builder(MainActivity.this);  
 builder.setView(dialogView)  
 .setPositiveButton("OK", new DialogInterface.OnClickListener() {  
 @Override  
 public void onClick(DialogInterface dialog, int which) {  
  
 }  
 })  
 .setNegativeButton("Cancel", new DialogInterface.OnClickListener() {  
 @Override  
 public void onClick(DialogInterface dialog, int which) {  
  
 }  
 });  
 builder.create().show();  
 }  
 });  
  
 }  
  
  
  
}

ДОДАТОК В (Код другого підзавдання)

<?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:id="@+id/main"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".MainActivity">  
  
 <TextView  
 android:id="@+id/handlerMessageTextView1"  
 android:layout\_width="223dp"  
 android:layout\_height="47dp"  
 android:gravity="center"  
 android:text="Message 1 will appear here"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.023" />  
  
 <Button  
 android:id="@+id/startHandlerButton1"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="80dp"  
 android:text="Start Handler 1"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.498"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
 <TextView  
 android:id="@+id/handlerMessageTextView2"  
 android:layout\_width="223dp"  
 android:layout\_height="47dp"  
 android:layout\_marginTop="24dp"  
 android:gravity="center"  
 android:text="Message 2 will appear here"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/startHandlerButton1" />  
  
 <TextView  
 android:id="@+id/handlerMessageTextView3"  
 android:layout\_width="223dp"  
 android:layout\_height="47dp"  
 android:layout\_marginTop="48dp"  
 android:gravity="center"  
 android:text="Message 3 will appear here"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/handlerMessageTextView2" />  
  
 <TextView  
 android:id="@+id/textView3"  
 android:layout\_width="228dp"  
 android:layout\_height="66dp"  
 android:layout\_marginTop="36dp"  
 android:text="Handler Thread"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.497"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/handlerMessageTextView3" />  
  
</androidx.constraintlayout.widget.ConstraintLayout>

package com.nuresemonovoleh.android\_pzpi\_23\_2\_semonov\_oleh\_practtask3\_2;  
  
import android.os.Bundle;  
import android.os.Handler;  
import android.os.HandlerThread;  
import android.os.Looper;  
import android.os.Message;  
import android.view.View;  
import android.widget.Button;  
import android.widget.TextView;  
  
import androidx.activity.EdgeToEdge;  
import androidx.appcompat.app.AppCompatActivity;  
import androidx.core.graphics.Insets;  
import androidx.core.view.ViewCompat;  
import androidx.core.view.WindowInsetsCompat;  
  
public class MainActivity extends AppCompatActivity {  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 EdgeToEdge.enable(this);  
 setContentView(R.layout.activity\_main);  
 ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main), (v, insets) -> {  
 Insets systemBars = insets.getInsets(WindowInsetsCompat.Type.systemBars());  
 v.setPadding(systemBars.left, systemBars.top, systemBars.right, systemBars.bottom);  
 return insets;  
 });  
  
 Handler handler1 = new Handler(Looper.getMainLooper());  
 Button startHandlerButton = findViewById(R.id.startHandlerButton1);  
 startHandlerButton.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 handler1.postDelayed(new Runnable() {  
 @Override  
 public void run() {  
 TextView textView = findViewById(R.id.handlerMessageTextView1);  
 textView.setText("Handler executed after delay");  
 }  
 }, 2000);  
 }  
 });  
  
  
  
 Handler handler2 = new Handler(Looper.getMainLooper());  
 new Thread(new Runnable() {  
 @Override  
 public void run() {  
 try {  
 Thread.sleep(3000);  
 } catch (InterruptedException e){  
 e.printStackTrace();  
 }  
 handler2.post(new Runnable() {  
 @Override  
 public void run() {  
 TextView textView = findViewById(R.id.handlerMessageTextView2);  
 textView.setText("Updated from background thread");  
 }  
 });  
 }  
 }).start();  
  
  
  
  
 Handler handler3 = new Handler(Looper.getMainLooper()){  
 @Override  
 public void handleMessage(Message msg){  
 TextView textView = findViewById(R.id.handlerMessageTextView3);  
 textView.setText("Message received: " + msg.what);  
 }  
 };  
 new Thread(new Runnable() {  
 @Override  
 public void run() {  
 try {  
 Thread.sleep(2000);  
 }  
 catch (InterruptedException e){  
 e.printStackTrace();  
 }  
 Message msg = handler3.obtainMessage();  
 msg.what = 1;  
 handler3.sendMessage(msg);  
 }  
 }).start();  
  
  
 }  
  
  
}

ДОДАТОК С (код третього підзавдання)

<?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"  
 tools:context=".MainActivity">  
  
 <androidx.recyclerview.widget.RecyclerView  
 android:id="@+id/recyclerView"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"/>  
  
</LinearLayout>

<?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">  
  
 <TextView  
 android:id="@+id/textView"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="192dp"  
 android:text="TextView"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.498"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
 <ImageView  
 android:id="@+id/imageView"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="52dp"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/textView"  
 app:srcCompat="@drawable/ic\_launcher\_background" />  
</androidx.constraintlayout.widget.ConstraintLayout>

package com.nuresemonovoleh.android\_pzpi\_23\_2\_semonov\_oleh\_practtask3\_3;  
  
import android.view.LayoutInflater;  
import android.view.View;  
import android.view.ViewGroup;  
import android.widget.ImageView;  
import android.widget.TextView;  
import android.widget.Toast;  
  
import androidx.annotation.NonNull;  
import androidx.recyclerview.widget.RecyclerView;  
  
public class MyAdapter extends RecyclerView.Adapter<MyAdapter.ViewHolder> {  
 private String[] mData;  
 private int[] mImageData;  
  
 public MyAdapter(String[] data, int[] imageData){  
 mData = data;  
 mImageData = imageData;  
 }  
  
 public static class ViewHolder extends RecyclerView.ViewHolder {  
 public TextView textView;  
 public ImageView imageView;  
  
 public ViewHolder(View v) {  
 super(v);  
 textView = v.findViewById(R.id.textView);  
 imageView = v.findViewById(R.id.imageView);  
 }  
 }  
 @NonNull  
 @Override  
 public MyAdapter.ViewHolder onCreateViewHolder(ViewGroup parent, int viewType) {  
 View v = LayoutInflater.from(parent.getContext())  
 .inflate(R.layout.list\_item, parent, false);  
 return new ViewHolder(v);  
 }  
  
 @Override  
 public void onBindViewHolder(ViewHolder holder, int position) {  
 holder.textView.setText(mData[position]);  
 holder.imageView.setImageResource(mImageData[position]);  
 holder.itemView.setOnClickListener(v -> {  
 Toast.makeText(v.getContext(),"Clicked: "  
 + mData[position],Toast.LENGTH\_SHORT).show();  
 });  
 }  
  
 @Override  
 public int getItemCount() {  
 return mData.length;  
 }  
}

package com.nuresemonovoleh.android\_pzpi\_23\_2\_semonov\_oleh\_practtask3\_3;  
  
import android.os.Bundle;  
import android.view.LayoutInflater;  
import android.view.View;  
import android.view.ViewGroup;  
import android.widget.TextView;  
  
import androidx.activity.EdgeToEdge;  
import androidx.annotation.NonNull;  
import androidx.appcompat.app.AppCompatActivity;  
import androidx.core.graphics.Insets;  
import androidx.core.view.ViewCompat;  
import androidx.core.view.WindowInsetsCompat;  
import androidx.recyclerview.widget.LinearLayoutManager;  
import androidx.recyclerview.widget.RecyclerView;  
  
public class MainActivity extends AppCompatActivity {  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 EdgeToEdge.enable(this);  
 setContentView(R.layout.activity\_main);  
 ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main), (v, insets) -> {  
 Insets systemBars = insets.getInsets(WindowInsetsCompat.Type.systemBars());  
 v.setPadding(systemBars.left, systemBars.top, systemBars.right, systemBars.bottom);  
 return insets;  
 });  
  
 String[] myDataset = {"Item1", "Item2", "Item3","Item4","Item5"};  
 int[] myImage = {R.drawable.ic\_launcher\_background, R.drawable.ic\_launcher\_foreground,  
 R.drawable.ic\_launcher\_background, R.drawable.ic\_launcher\_foreground,  
 R.drawable.ic\_launcher\_background};  
 RecyclerView recyclerView = findViewById(R.id.recyclerView);  
 recyclerView.setLayoutManager(new LinearLayoutManager(this));  
 MyAdapter adapter = new MyAdapter(myDataset,myImage);  
 recyclerView.setAdapter(adapter);  
 }  
  
  
  
  
}