МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ

КАФЕДРА ПРОГРАМНОЇ ІНЖЕНЕРІЇ

Звіт

з практичної роботи № 2 з дисципліни

Програмування під Android

Виконав: Перевірив:

ст. гр. ПЗПІ-23-4 ст. викладач кафедри ПІ

Прядко Максим Сокорчук Ігор Петрович

Харків 2024

Завдання

1. Розмітка екрана
   1. Підготовка робочого середовища
   2. Практика з Linear/Constraint-Layout
   3. Практика з атрибутами елементів
   4. Обробка подій
2. Файли розмітки Layout
   1. Створення різних Layout
   2. Практика роботи з атрибутами
3. Життєвий цикл та методи Activity
   1. Ознайомлення з життєвим циклом Activity
   2. Дослідження методів життєвого циклу
   3. Вивчення поведінки Activity при зміні орієнтації екрана
   4. Практичне завдання з ручного збереження стану
   5. Дослідження переходу між Activity
   6. Обробка завершення Activity

Хід роботи

1. Розмітка екрана

1.1 Редагування XML-файлу розмітки

Відкриємо файл res/layout/activity\_main.xml та створимо простий інтерфейс з використанням елементів, таких як TextView, Button, та ImageView:

1.2 Практика з LinearLayout

Створимо нову розмітку з використанням LinearLayout. В ній додамо декілька кнопок розташованих одне під одною.

Для розмітки задамо атрибут android:orientation="vertical", для того щоб кнопки йшли у вертикальному порядку. Для кнопок задамо властивості layout\_width та layout\_height з значеннями wrap\_content та match\_parent.

1.3 Практика з ConstraintLayout

Створимо макет з використанням різних елементів, використовуючи обмеження для позиціюванняю

1.4 Практика з атрибутами елементів

Дослідимо атрибути, як padding, margin, gravity та layout\_gravity для налаштування відступів та вирівнювання елементів, а також налаштуємо текстові поля( розмір шрифту, колір тощо)

1.5 Обробка подій кнопки

Додамо до кнопки обробник подій.

До кнопок додамо атрибути android:onClick="onChangeTextClick" та android:onClick="onShowMessageClick"

а у коді реалізуємо відповідні метои, а саме:

1. public void onChangeTextClick(View view) {  
2. textView.setText("Текст змінено!");  
3. }

1. public void onShowMessageClick(View view) {  
2. Toast.*makeText*(this, "Це повідомлення!", Toast.*LENGTH\_SHORT*).show();  
3. }

Перший метод буде замінювати текст у TextView. А другий виводитиме повідомлення.

1.6 Завдання для самостійної роботи

Створимо розмітку GridLayout та розмістимо у ній текстові поля, кнопки та зображення. Додамо кілька кнопок з різним функціоналом. Перша кнопка змінюватиме розмір тексу, а друга змінюватиме колір на синій:

public void onButton1Click(View view) {  
 if (isLargeText) {  
 textView.setTextSize(18);  
 } else {  
 textView.setTextSize(24);  
 }  
 isLargeText = !isLargeText;  
}

public void onButton2Click(View view) {  
 if (isColorChanged) {  
 textView.setTextColor(Color.*BLACK*);  
 } else {  
 textView.setTextColor(Color.*BLUE*);  
 }  
 isColorChanged = !isColorChanged;  
}

2. Файли розмітки layout

2.1 Основи роботи з XML-розміткою

Використаємо LinearLayout як основний контейнер. Також змінемо його орієнтацію на вертикальну й додамо такі елементи як TextView, Button, ImageView і змінимо їх ширину та висоту.

2.2 Створення різних Layout

Створимо макет на основі LinearLayout з використанням кнопок одної під одною та використанням атрибутів gravity i layout\_gravity:

<Button  
 android:id="@+id/button3"  
 android:layout\_width="161dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="end"  
 android:text="Button" />

Далі створимо макет на основі RelativeLayout та розмістимо декілька елементів використовуючи атрибути layout\_below.

android:layout\_below="@id/button4"

Тепер перейдемо до FrameLayout та додамо декілька елементів

Останнім пунктом було створення макету на основі GridLayout, тобто таблиці. Заповнимо її кнопками, попередньо визначимо кількість рядків і стовпців:

1. android:columnCount="3"  
2. android:rowCount="2">

2.3 Практика роботи за атрибутами

Дослідимо атрибути, такі як padding, margin, textSize, textColor, а також властивість weight у LinearLayout.

2.4 Завдання для самостійної роботи

Створимо інтерфейс користувача для застосунку калькулятора за допомогою GridLayout.

3. Життєвий цикл та методи activity

3.1 Додайте до кожного методу життєвого циклу (onCreate, onStart, onResume тощо) функцію виведення повідомлень у лог (Log.d) для відстеження послідовності викликів. Всі вони однакові, тож приведу лише один як приклад:

1. Log.d(TAG, "onStart: ");

3.2 Змініть орієнтацію пристрою (з портретної на ландшафтну) та спостерігайте за викликами методів життєвого циклу у логах.

3.3 Дослідіть, як зміна орієнтації перезапускає Activity (викликаються методи onPause(), onStop(), onDestroy(), а потім onCreate(), onStart(), onResume()).

3.4 Додайте в Activity поле для введення тексту (наприклад, EditText).

При зміні орієнтації екрана текст у полі буде очищено, оскільки Activity перезавантажується.

Було реалізовано збереження та відновлення вручну. Було створено клікер, що зберігає кількість кліків.

1. public void onIncreaseCounter(View v) {

2. TextView count = findViewById(R.id.count);

3. Integer previousCounter = Integer.parseInt(count.getText().toString());

4. Integer newValue = previousCounter + 1;

5. count.setText(newValue.toString());

6. }

Зберігається за допомогою:

1.protected void onSaveInstanceState(@NonNull Bundle outState) {

2. TextView count = findViewById(R.id.count);

3. outState.putString("counter", count.getText().toString());

4. super.onSaveInstanceState(outState);

Відновлюється за допогою:

1. protected void onRestoreInstanceState(@NonNull Bundle savedInstanceState){

2. super.onRestoreInstanceState(savedInstanceState);

3. TextView count = findViewById(R.id.count);

4. count.setText(savedInstanceState.getString("counter"));

5. }

Створіть друге Activity у вашому застосунку.

Додайте кнопку для переходу до другого Activity.

Відстежте виклики методів життєвого циклу при переході між Activity.

Додайте кнопку для завершення Activity (виклик методу finish()).

1. public void onFinishClick(View v) {

2. finish();

3. }

Висновки

Під час практичної роботи було досліджено: створення та налаштування різних розміток, використовуючи XML-файли. Виявлено, як компоненти та атрибути взаємодіють між собою.

Ознайомилися із життєвим циклом компонентів Activity у застосунках Android. Навчилися керувати станами Activity через використання ключових методів життєвого циклу.

Відеозвіт на YouTube. URL: <https://youtu.be/yX_zFyhV6fc>

Використані джерела

1. Методичні вказівки до практичних занять: Розмітка екрана в Android. URL: <https://dl.nure.ua/mod/page/view.php?id=599542> (дата звернення: 22.11.2024)
2. Методичні вказівки до практичних занять: Файли розмітки Layout. URL: <https://dl.nure.ua/mod/page/view.php?id=599543> (дата звернення: 22.11.2024)
3. Методичні вказівки до практичного заняття: Життєвий цикл та методи Activity. URL: <https://dl.nure.ua/mod/page/view.php?id=599544> (дата звернення: 2.11.2024)