Android Kotlin Mobile Development

Building A Simple Calculator App In Android With Kotlin

May 15, 2024 0 views 5 min read #Android

▼ Table Of Content

Building a Simple Calculator App in Android with Kotlin

Introduction

Prerequisites
Setup Your Project
Designing the UI
Implementing Functionality in Kotlin
Running Your App
Demo Video
Download Links
Conclusion

Building a Simple Calculator App in Android with Kotlin#

Introduction#

In this tutorial, we will create a simple calculator app using Android Studio, Kotlin, and the mXparser library to handle mathematical expressions. This project is great for beginners and intermediate developers who want to enhance their skills in Android app development.

Prerequisites#

- Android Studio installed on your computer.
- Basic understanding of Kotlin and XML layout design.
- Familiarity with third-party libraries in Android.

Setup Your Project#

1. **Create a New Android Project**:
    - Open Android Studio.
    - Click on "New Project".
    - Select "Empty Activity".
    - Name your project "Calculator".

- Set "Kotlin" as the programming language.
- Finish.

2. **Add mXparser Library**:
   - Open your `build.gradle (Module: app)` file.
   - Add the following dependency to include the mXparser library:

```
implementation("org.mariuszgromada.math:MathParser.org-mXparser:4.4.2")
```

3. **Enable View Binding**:
   - Still in the `build.gradle (Module: app)` file, add the following inside the `android` block:

```
viewBinding {
    enabled = true
}
```

4. **Sync Your Project**:
   - Click "Sync Now" in the bar that appears in Android Studio to ensure all configurations are updated.

Designing the UI#

1. **Modify `activity_main.xml`**:
   - Replace the content with the following XML to create the UI for your calculator:

```
<?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:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:background="@color/window_background"
    tools:context=".MainActivity">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:gravity="center"
        android:onClick="openWebsite"
        android:padding="16dp"
        android:text="ibsanju Calculator"
```

```
            android:textColor="@color/black"
            android:textSize="24sp" />

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="0dp"
            android:layout_weight="1"
            android:gravity="bottom"
            android:padding="16dp"
            android:background="@color/io_background"
            android:orientation="vertical">

            <TextView
                android:id="@+id/input"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:gravity="end"
                android:textSize="30sp"
                android:maxLines="3"
                android:textColor="@color/text_main"
                android:fontFamily="sans-serif-light"
                tools:text="5+10-3" />

            <TextView
                android:id="@+id/output"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:gravity="end"
                android:textSize="50sp"
                android:maxLines="2"
                android:fontFamily="sans-serif"
                android:textColor="@color/green"
                tools:text="12" />

        </LinearLayout>

        <TableLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:stretchColumns="*">

            <TableRow>

                <androidx.appcompat.widget.AppCompatButton
                    android:id="@+id/button_clear"
                    android:layout_width="wrap_content"
                    android:layout_height="90dp"
                    style="@style/Button_Style"
                    android:textColor="@color/red"
                    android:text="C" />
                <androidx.appcompat.widget.AppCompatButton
```

```
                android:id="@+id/button_bracket"
                android:layout_width="wrap_content"
                android:layout_height="90dp"
                style="@style/Button_Style"
                android:textColor="@color/green"
                android:text="(" />
            <androidx.appcompat.widget.AppCompatButton
                android:id="@+id/button_bracket_r"
                android:layout_width="wrap_content"
                android:layout_height="90dp"
                style="@style/Button_Style"
                android:textColor="@color/green"
                android:text=")" />
            <androidx.appcompat.widget.AppCompatButton
                android:textColor="@color/green"
                android:id="@+id/button_division"
                android:layout_width="wrap_content"
                android:layout_height="90dp"
                style="@style/Button_Style"
                android:text="÷" />

        </TableRow>

        <TableRow>

            <androidx.appcompat.widget.AppCompatButton
                android:id="@+id/button_7"
                android:layout_width="wrap_content"
                android:layout_height="90dp"
                style="@style/Button_Style"
                android:text="7" />
            <androidx.appcompat.widget.AppCompatButton
                android:id="@+id/button_8"
                android:layout_width="wrap_content"
                android:layout_height="90dp"
                style="@style/Button_Style"
                android:text="8" />
            <androidx.appcompat.widget.AppCompatButton
                android:id="@+id/button_9"
                android:layout_width="wrap_content"
                android:layout_height="90dp"
                style="@style/Button_Style"
                android:text="9" />
            <androidx.appcompat.widget.AppCompatButton
                android:textColor="@color/green"
                android:id="@+id/button_multiply"
                android:layout_width="wrap_content"
                android:layout_height="90dp"
                style="@style/Button_Style"
                android:text="×" />

```

```
        </TableRow>
        <TableRow>

            <androidx.appcompat.widget.AppCompatButton
                android:id="@+id/button_4"
                android:layout_width="wrap_content"
                android:layout_height="90dp"
                style="@style/Button_Style"
                android:text="4" />
            <androidx.appcompat.widget.AppCompatButton
                android:id="@+id/button_5"
                android:layout_width="wrap_content"
                android:layout_height="90dp"
                style="@style/Button_Style"
                android:text="5" />
            <androidx.appcompat.widget.AppCompatButton
                android:id="@+id/button_6"
                android:layout_width="wrap_content"
                android:layout_height="90dp"
                style="@style/Button_Style"
                android:text="6" />
            <androidx.appcompat.widget.AppCompatButton
                android:textColor="@color/green"
                android:id="@+id/button_subtraction"
                android:layout_width="wrap_content"
                android:layout_height="90dp"
                style="@style/Button_Style"
                android:textSize="40sp"
                android:text="-" />

        </TableRow>
        <TableRow>

            <androidx.appcompat.widget.AppCompatButton
                android:id="@+id/button_1"
                android:layout_width="wrap_content"
                android:layout_height="90dp"
                style="@style/Button_Style"
                android:text="1" />
            <androidx.appcompat.widget.AppCompatButton
                android:id="@+id/button_2"
                android:layout_width="wrap_content"
                android:layout_height="90dp"
                style="@style/Button_Style"
                android:text="2" />
            <androidx.appcompat.widget.AppCompatButton
                android:id="@+id/button_3"
                android:layout_width="wrap_content"
                android:layout_height="90dp"
                style="@style/Button_Style"
                android:text="3" />
```

```
            <androidx.appcompat.widget.AppCompatButton
                android:textColor="@color/green"
                android:id="@+id/button_addition"
                android:layout_width="wrap_content"
                android:layout_height="90dp"
                style="@style/Button_Style"
                android:text="+" />

        </TableRow>
        <TableRow>

            <androidx.appcompat.widget.AppCompatButton
                android:id="@+id/button_croxx"
                android:layout_width="wrap_content"
                android:layout_height="90dp"
                style="@style/Button_Style"
                android:text="AC" />

            <androidx.appcompat.widget.AppCompatButton
                android:id="@+id/button_0"
                android:layout_width="wrap_content"
                android:layout_height="90dp"
                style="@style/Button_Style"
                android:text="0" />
            <androidx.appcompat.widget.AppCompatButton
                android:id="@+id/button_dot"
                android:layout_width="wrap_content"
                android:layout_height="90dp"
                style="@style/Button_Style"
                android:text="." />
            <androidx.appcompat.widget.AppCompatButton
                android:id="@+id/button_equals"
                android:layout_width="0dp"
                android:layout_height="90dp"
                android:layout_weight="1"
                style="@style/Button_Style"
                android:textColor="@color/green"
                android:text="=" />

        </TableRow>
    </TableLayout>

</LinearLayout>
```

- This layout uses `LinearLayout` and `TableLayout` to organize the buttons and display panels.

2. **Define Colors and Styles**:

- In `res/values/colors.xml`, define the necessary colors:

```xml
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <color name="purple_200">#FFBB86FC</color>
    <color name="purple_500">#FF6200EE</color>
    <color name="purple_700">#F9F9F9</color>
    <color name="teal_200">#FF03DAC5</color>
    <color name="teal_700">#FF018786</color>
    <color name="black">#FF000000</color>
    <color name="white">#FFFFFFFF</color>

    <color name="window_background">#FFFFFFFF</color>
    <color name="io_background">#F9F9F9</color>
    <color name="green">#4ea043</color>
    <color name="red">#d14f4f</color>
    <color name="text_main">#FF000000</color>
</resources>
```

Home About Contact

- In `res/values/styles.xml`, define a style for the calculator buttons:

```xml
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <style name="Button_Style" parent="Widget.AppCompat.Button.Borderless">
        <item name="android:textSize">24sp</item>
        <item name="android:textColor">@color/black</item>
        <item name="android:gravity">center</item>
        <item name="fontFamily">sans-serif-light</item>
    </style>
</resources>
```

Implementing Functionality in Kotlin#

1. **Setup the `MainActivity`:**
    - Open `MainActivity.kt`.
    - Implement view binding and initialize UI components.
    - Set up button click listeners to build the expression and calculate results.

```kotlin
package com.ibsanju.calculator

import android.content.Intent
import android.net.Uri
import android.os.Bundle
import android.view.View
import androidx.appcompat.app.AppCompatActivity
import androidx.appcompat.app.AppCompatDelegate
```

```kotlin
import androidx.core.content.ContextCompat
import com.ibsanju.calculator.databinding.ActivityMainBinding
import org.mariuszgromada.math.mxparser.Expression
import java.text.DecimalFormat

class MainActivity : AppCompatActivity() {
    private lateinit var binding: ActivityMainBinding

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        binding = ActivityMainBinding.inflate(layoutInflater)
        setContentView(binding.root)

        AppCompatDelegate.setDefaultNightMode(AppCompatDelegate.MODE_NIGHT_FOLLOW_SYSTEM)
        setupCalculatorButtons()
    }

    private fun setupCalculatorButtons() {
        // Number buttons
        listOf(
            binding.button0, binding.button1, binding.button2, binding.button3,
            binding.button4, binding.button5, binding.button6, binding.button7,
            binding.button8, binding.button9
        ).forEachIndexed { index, button ->
            button.setOnClickListener {
                binding.input.text = addToInputText(index.toString())
                showResult()
            }
        }

        // Operators and other buttons
        binding.buttonCroxx.apply {
            setOnClickListener {
                binding.input.text = ""
                binding.output.text = ""
            }
        }

        binding.buttonBracket.setOnClickListener { binding.input.text = addToInputText("(") }
        binding.buttonBracketR.setOnClickListener {
            binding.input.text = addToInputText(")"); showResult();
        }
        binding.buttonClear.setOnClickListener {
            binding.input.text = binding.input.text.dropLast(1)
            showResult()
        }
        binding.buttonDot.setOnClickListener { binding.input.text = addToInputText(".") }
        binding.buttonDivision.setOnClickListener { binding.input.text = addToInputText("÷") }
        binding.buttonMultiply.setOnClickListener { binding.input.text = addToInputText("×") }
        binding.buttonSubtraction.setOnClickListener { binding.input.text = addToInputText("-") }
        binding.buttonAddition.setOnClickListener { binding.input.text = addToInputText("+") }
```

```
            binding.buttonEquals.setOnClickListener { showResult() }
        }

        private fun addToInputText(buttonValue: String): String {
            val currentText = binding.input.text.toString()
            if (buttonValue == "(" && currentText.isNotEmpty() && currentText.last().isDigit()) {
                // If last character is a number and the new input is an opening parenthesis
                return "$currentText×("
            } else if (buttonValue.first()
                    .isDigit() && currentText.isNotEmpty() && currentText.last() == ')'
            ) {
                // If the new input starts with a digit and the last character is a closing parenthesis
                return "$currentText×$buttonValue"
            }
            return currentText + buttonValue
        }


        private fun getInputExpression(): String =
            binding.input.text.toString()
                .replace("÷", "/")
                .replace("×", "*")

        private fun showResult() {
            try {
                val expression = getInputExpression()
                val result = Expression(expression).calculate()
                if (result.isNaN()) {
                    binding.output.text = "Invalid Input"
                    binding.output.setTextColor(ContextCompat.getColor(this, R.color.red))
                } else {
                    binding.output.text = DecimalFormat("0.######").format(result).toString()
                    binding.output.setTextColor(ContextCompat.getColor(this, R.color.green))
                }
            } catch (e: Exception) {
                binding.output.text = "Error: " + e.message
                binding.output.setTextColor(ContextCompat.getColor(this, R.color.red))
            }
        }

        fun openWebsite(view: View) {
            val intent = Intent(Intent.ACTION_VIEW, Uri.parse("https://www.ibsanju.com"))
            startActivity(intent)
        }
    }

```


- Ensure that input handling for operations is correct, especially for inserting multiplication signs when needed next to parentheses.

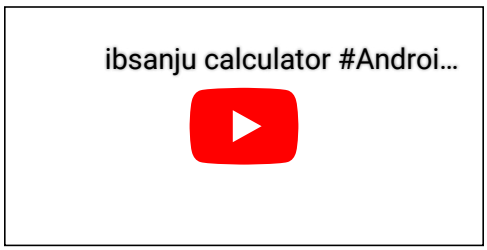
2. **Handling Mathematical Expressions**:
   - Use the mXparser library to evaluate the expression entered by the user.
   - Handle exceptions and invalid inputs gracefully.

Running Your App#

- **Build and Run**:
  - Connect an Android device or use an emulator.
  - Run the application and test the functionality of your calculator.

Demo Video#

Check out the demo of **"Calculator App"** below:



Download Links#

- **Android (apk)**: click here to download

Conclusion#

Congratulations! You've now built a fully functional calculator app for Android that handles basic arithmetic operations. As an extension, consider adding features such as complex operations, history of calculations, or even graphing capabilities.

Enter your email Submit

Home About Contact

sitemap.xml

Made with ♥ by IbsanjU