

9. Приложение калькулятор

- Подключил binding
- Добавил необходимые цвета в colors:

```
■ <color name="black">#FF000000</color>
■ <color name="white">#FFFFFFF</color>

■ <color name="purple_200">#FFBB86FC</color>
■ <color name="purple_500">#FF6200EE</color>
■ <color name="purple_700">#FF3700B3</color>
■ <color name="teal_200">#FF03DAC5</color>
■ <color name="teal_700">#FF018786</color>
■ <color name="red">#FF3131</color>
■ <color name="pink">#FFC0CB</color>
■ <color name="orange">#FFA500</color>
■ <color name="neon_green">#39FF14</color>

■ <style name="Button_Style" parent="Widget.AppCompat.Button.Colored">
■ <item name="android:background">@color/white</item>
■ <item name="android:textSize">24sp</item>
■ <item name="android:textColor">@color/black</item>
■ <item name="android:gravity">center</item>
■ </style>
```

- Сделал макет приложения:

```
■ <LinearLayout
■   android:layout_width="match_parent"
■   android:layout_height="0dp"
■   android:layout_weight="1"
■   android:background="@color/black"
■   android:gravity="bottom"
■   android:orientation="vertical"
■   android:padding="30dp">
■   <TextView
■     android:id="@+id/input"
■     android:layout_width="match_parent"
■     android:layout_height="wrap_content"
■     android:gravity="end"
■     android:textColor="@color/white"
■     android:textSize="30sp"
■     tools:text="5+10-3" />
■   <TextView
■     android:id="@+id/output"
■     android:layout_width="match_parent"
■     android:layout_height="wrap_content"
■     android:gravity="end"
■     android:textColor="@color/neon_green"
■     android:textSize="50sp"
■     tools:text="12" />
■ </LinearLayout>
■ <TableLayout
■   android:layout_width="match_parent"
■   android:layout_height="wrap_content"
■   android:stretchColumns="*>
```



```

<androidx.appcompat.widget.AppCompatButton
    android:id="@+id/button_addition"
    style="@style/Button_Style"
    android:layout_width="wrap_content"
    android:layout_height="90dp"
    android:text="+"
    android:textColor="@color/orange" />
</TableRow>
<TableRow>
    <androidx.appcompat.widget.AppCompatButton
        android:id="@+id/button_percent"
        style="@style/Button_Style"
        android:layout_width="wrap_content"
        android:layout_height="90dp"
        android:text="%" />
    <androidx.appcompat.widget.AppCompatButton
        android:id="@+id/button_0"
        style="@style/Button_Style"
        android:layout_width="wrap_content"
        android:layout_height="90dp"
        android:text="0" />
    <androidx.appcompat.widget.AppCompatButton
        android:id="@+id/button_dot"
        style="@style/Button_Style"
        android:layout_width="wrap_content"
        android:layout_height="90dp"
        android:text="." />
    <androidx.appcompat.widget.AppCompatButton
        android:id="@+id/button_equals"
        style="@style/Button_Style"
        android:layout_width="wrap_content"
        android:layout_height="90dp"
        android:background="@color/orange"
        android:text="="
        android:textColor="@color/white" />
</TableRow>

```

- Прописал код для MainActivity:


```

// Функция для добавления текста к полю ввода
private fun addToInputText(value: String) {
    binding.input.append(value) // Добавляем переданное значение в конец поля ввода
}

// Функция для получения строки ввода
private fun getInputExpression(): String {
    return binding.input.text.toString() // Возвращаем текст из поля ввода
}

// Функция для показа результата вычислений
private fun showResult() {
    try {
        // Заменяем % на соответствующее значение для деления
        val expression = getInputExpression().replace(oldValue: "%", newValue: "/100")
        val result = ExpressionBuilder(expression).build().evaluate()
        binding.output.text =
            DecimalFormat(pattern: "0.#####").format(result).toString()
        binding.output.setTextColor(
            ContextCompat.getColor(context: this,
                R.color.neon_green))
    } catch (e: Exception) {
        binding.output.text = "Ошибка"
        binding.output.setTextColor(ContextCompat.getColor(context: this,
            R.color.red))
    }
}
}

```

- Сделал простую заставку:

Калькулятор

- Прописал код

```
android.os.Handler(Looper.getMainLooper()).postDelayed({  
    val intent = Intent( packageContext: this,MainActivity::class.java)  
    startActivity(intent)  
}, delayMillis: 10000)
```

- Непонятно какие изменения необходимо сделать в AndroidManifest, поэтому пока заставка отсутствует.

Теперь, что бы наша заставка отобразилась прежде, чем запуститься калькулятор, вы должны выполнить изменения в AndroidManifest.

- Результат:

