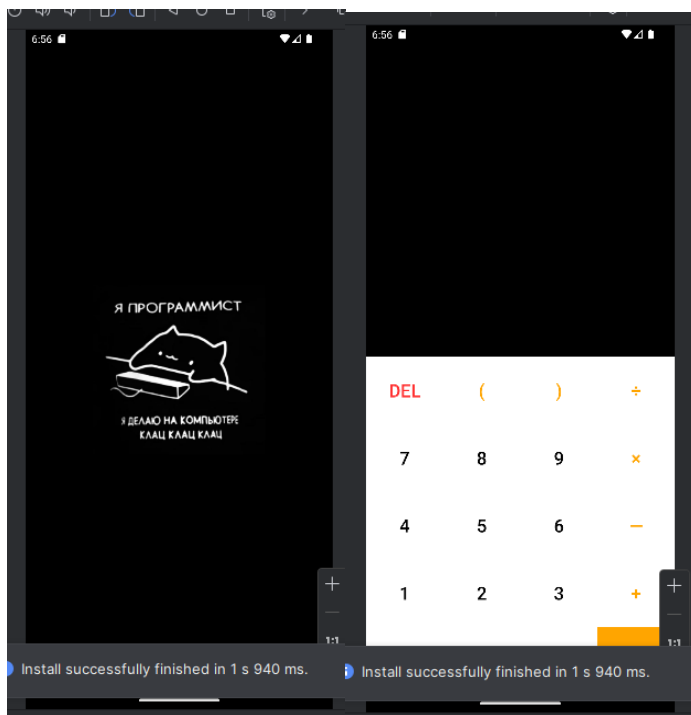


Приложение калькулятор

Задание

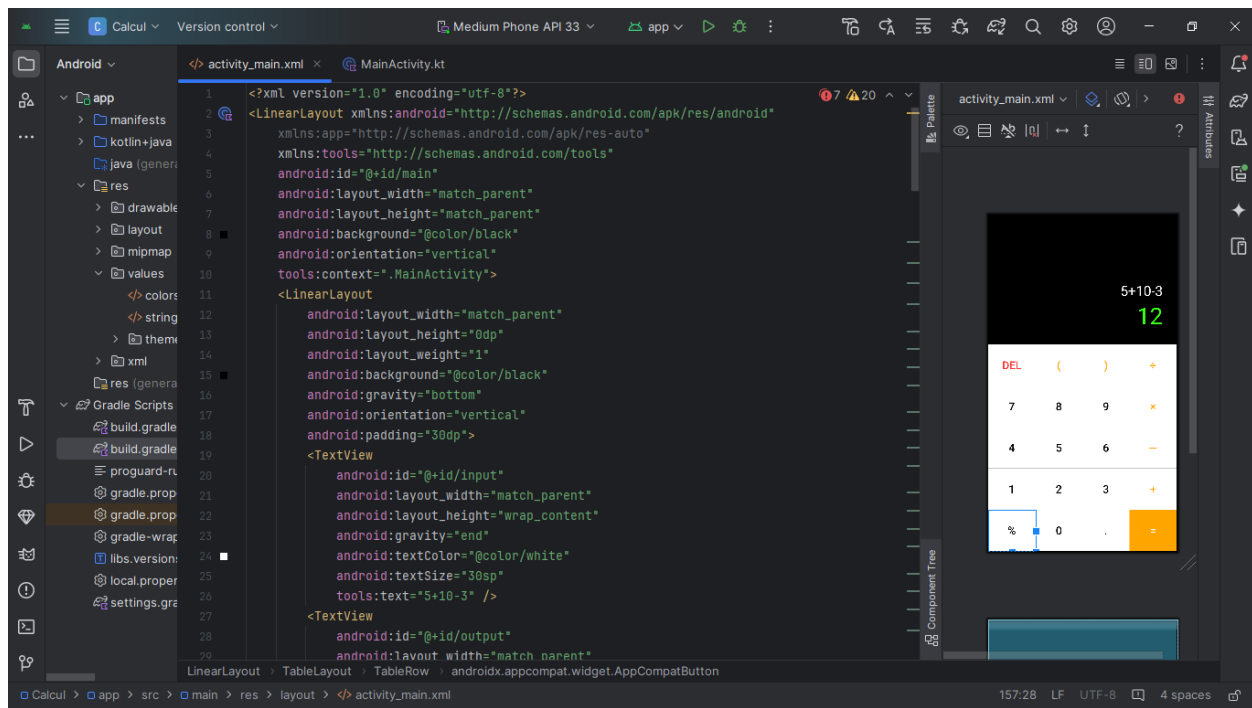


Ход работы:

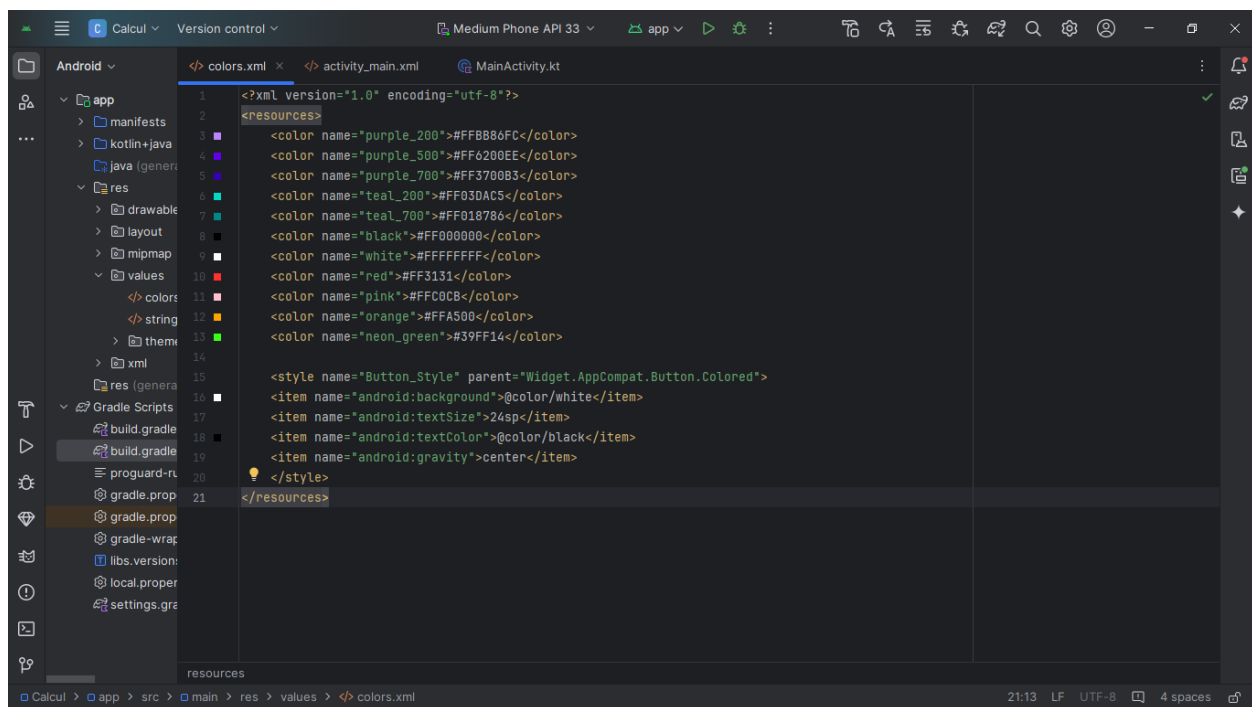
1. По образцу пишу программу

```
13  ss MainActivity : AppCompatActivity() {
14  private lateinit var binding: ActivityMainBinding
15
16  override fun onCreate(savedInstanceState: Bundle?) {
17      super.onCreate(savedInstanceState)
18      enableEdgeToEdge()
19
20      binding = ActivityMainBinding.inflate(layoutInflater)
21      setContentView(binding.root)
22
23      ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main)) { v, insets -> {
24          val systemBars = insets.getInsets(WindowInsetsCompat.Type.systemBars())
25          v.setPadding(systemBars.left, systemBars.top, systemBars.right, systemBars.bottom)
26      }}
27
28      binding.buttonClear.setOnClickListener {
29          binding.input.text = ""
30          binding.output.text = ""
31      }
32
33      binding.buttonBracketLeft.setOnClickListener { addToInputText("(") }
34      binding.buttonBracketRight.setOnClickListener { addToInputText(")") }
35      binding.button0.setOnClickListener { addToInputText("0") }
36      binding.button1.setOnClickListener { addToInputText("1") }
37      binding.button2.setOnClickListener { addToInputText("2") }
38      binding.button3.setOnClickListener { addToInputText("3") }
39      binding.button4.setOnClickListener { addToInputText("4") }
40      binding.button5.setOnClickListener { addToInputText("5") }
41      binding.button6.setOnClickListener { addToInputText("6") }
42      binding.button7.setOnClickListener { addToInputText("7") }
43      binding.button8.setOnClickListener { addToInputText("8") }
44      binding.button9.setOnClickListener { addToInputText("9") }
45      binding.buttonAdd.setOnClickListener { addToInputText("+") }
46      binding.buttonSubtraction.setOnClickListener { addToInputText("-") }
47      binding.buttonMultiply.setOnClickListener { addToInputText("*") }
48      binding.buttonDivision.setOnClickListener { addToInputText("/") }
49      binding.buttonPercent.setOnClickListener { addToInputText("%") }
50      binding.buttonEquals.setOnClickListener { showResult() }
51
52  .vate fun addToInputText(value: String) {
53      binding.input.append(value)
54  }
55
56  .vate fun getInputExpression(): String {
57      return binding.input.text.toString()
58  }
59
60  .vate fun showResult() {
61      try {
62          val expression = getInputExpression().replace(oldValue: "%", newValue: "")
63          val result = ExpressionBuilder(expression).build().evaluate()
64          binding.output.text = DecimalFormat(pattern: "0.#####").format(result)
65          binding.output.setTextColor(ContextCompat.getColor(context, this, R.color.black))
66      } catch (e: Exception) {
67          Log.e("MainActivity", "Ошибка: ${e.message}", e)
68          binding.output.text = "Ошибка"
69          binding.output.setTextColor(ContextCompat.getColor(context, this, R.color.red))
70      }
71  }
```

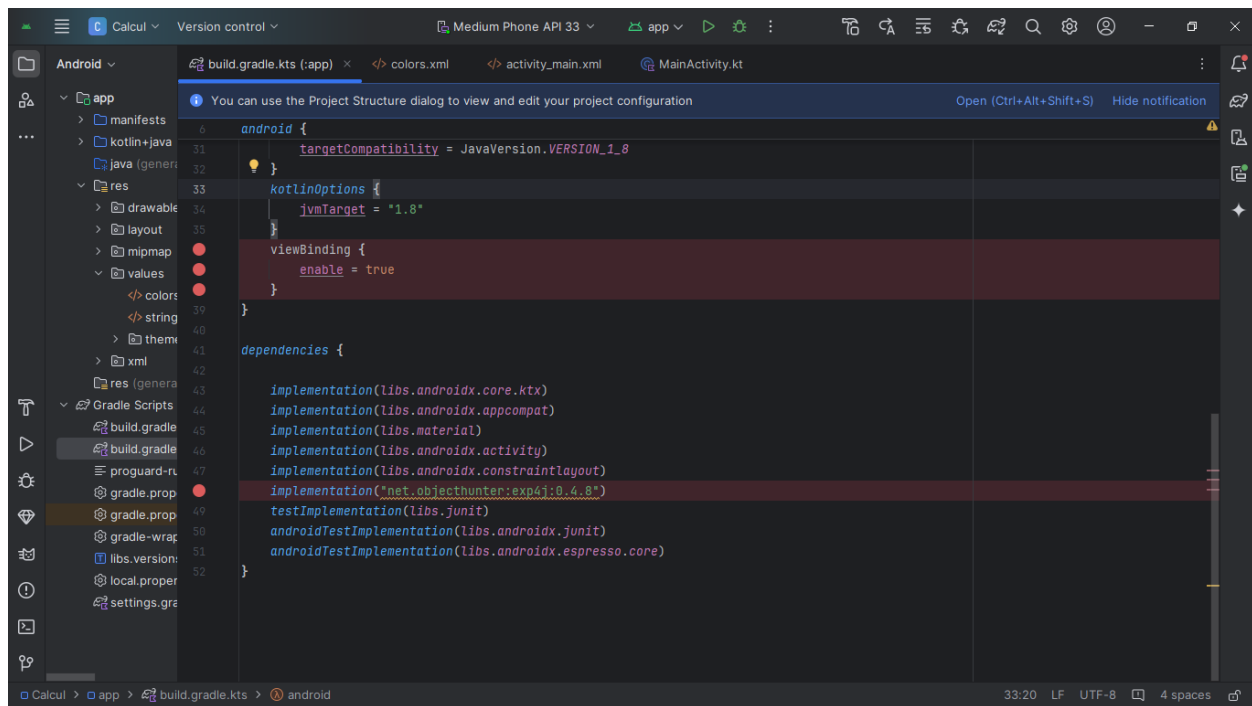
// MainActivity.kt + задание с binding



// activity_main.xml

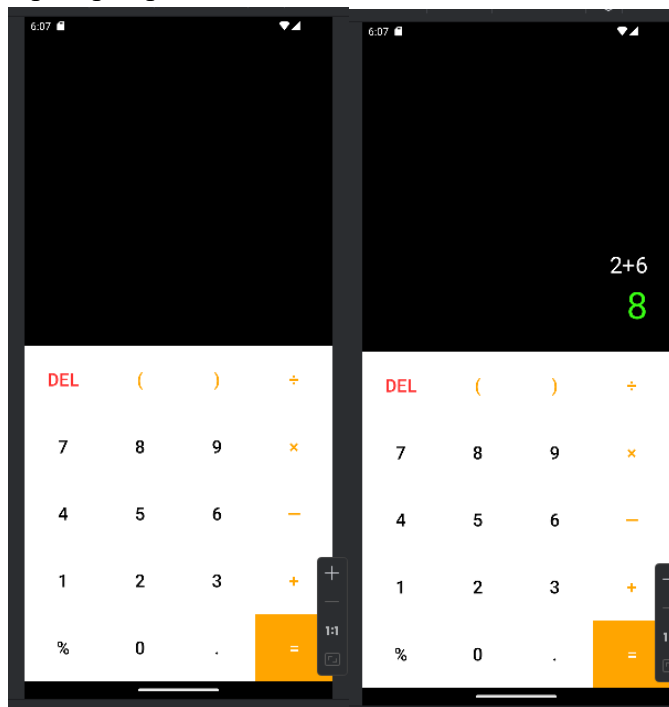


// color.xml

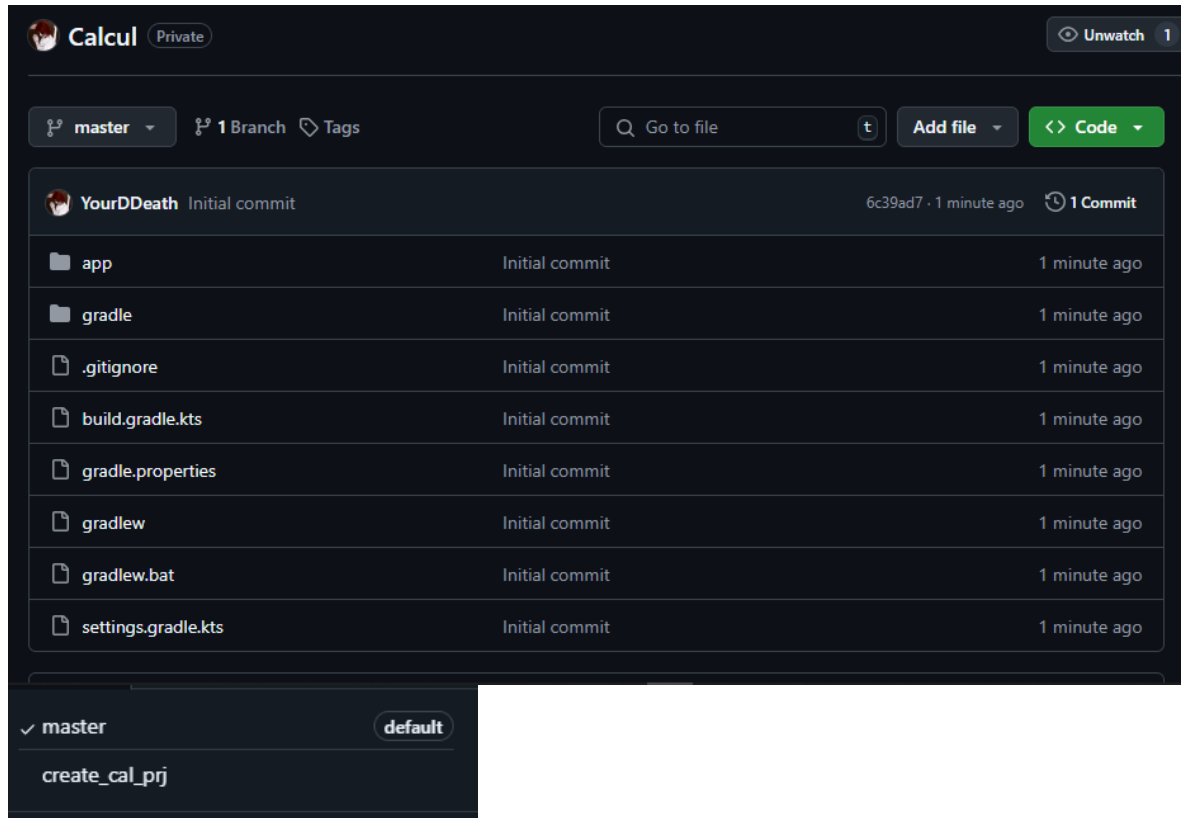


// Добавление библиотек

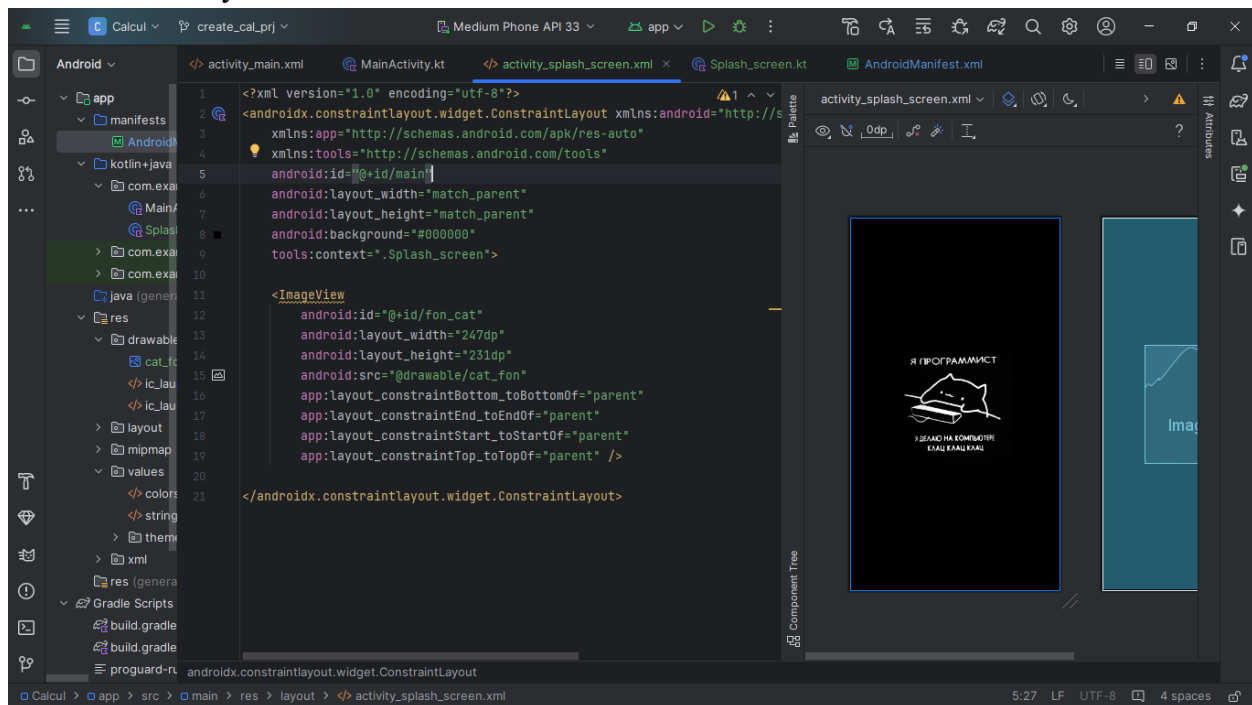
2. Проверка работы



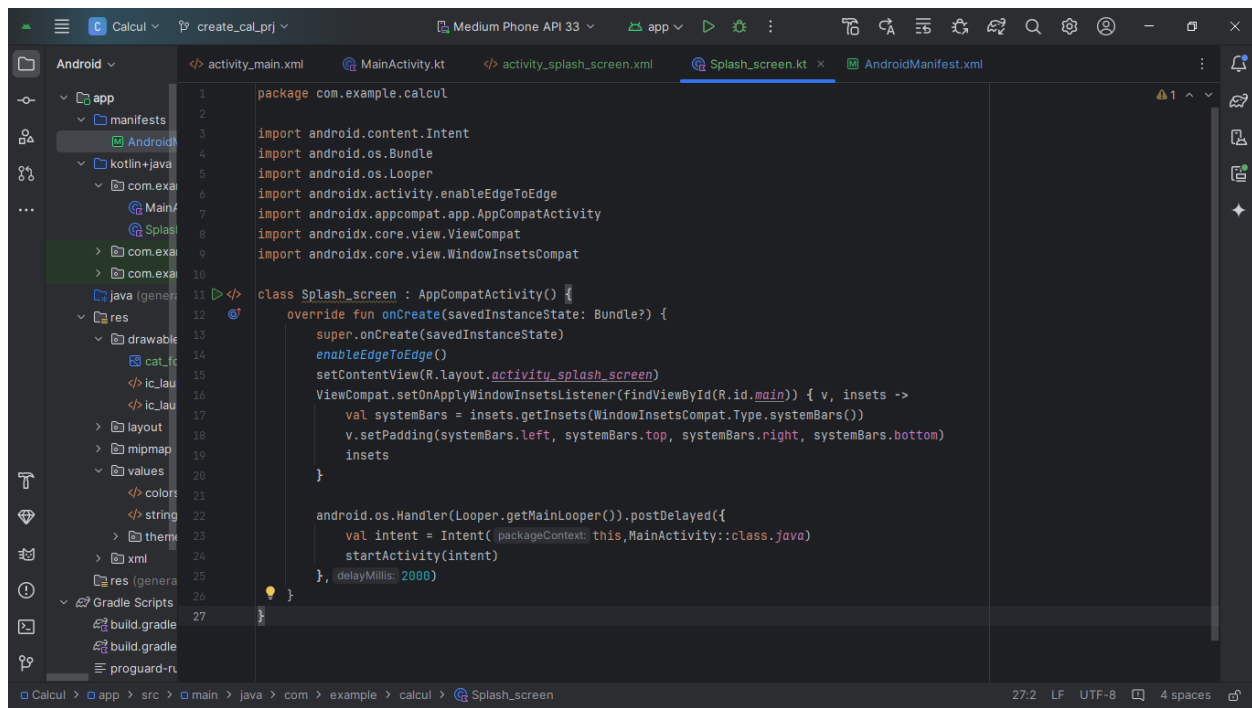
3. Создаю репозиторий в Гите и настраиваю ветки



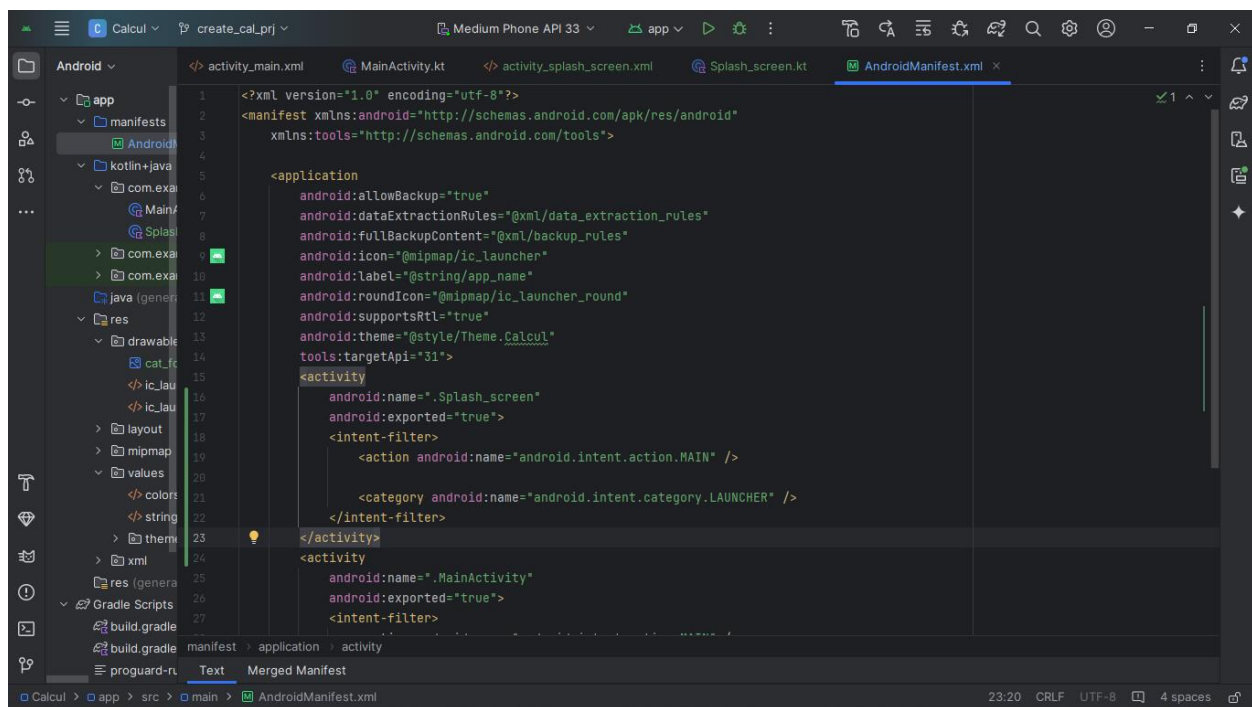
4. Создаю заставку



// xml

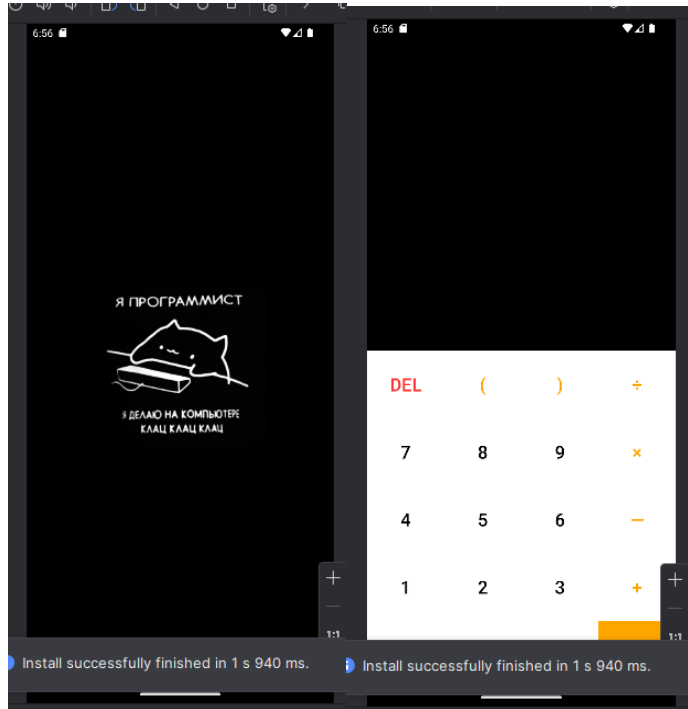


// kt



// Манифест

5. Проверяю работу заставки



6. После этого возвращаюсь на главную ветку, чтобы убедиться, что там этих изменений нет

