



**Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)**

ФАКУЛЬТЕТ _____ «Информатика и системы управления»

КАФЕДРА _____ «Теоретическая информатика и компьютерные технологии»

Рубежный контроль № 1

по курсу «Разработка мобильных приложений»

Студент группы ИУ9-71Б Яровикова А. С.

Преподаватель Посевин Д. П.

Москва 2023

1 Цель

Реализовать мобильное приложение выполняющее построение простейших геометрических фигур.

2 Задание

Через форму должен вводиться порядковый номер календарного года, состоящий из четырех чисел. Под формой должен возвращаться номер года. Если год является високосным, то номер года выводится зеленым цветом, в противном случае красным.

3 Реализация

Исходный код представлен в листинге 1.

Листинг 1: main.dart

```
1 import 'package:flutter/material.dart';
2
3 void main() {
4   runApp(MyApp());
5 }
6
7 class MyApp extends StatelessWidget {
8   @override
9   Widget build(BuildContext context) {
10    return MaterialApp(
11      debugShowCheckedModeBanner: false,
12      home: Scaffold(
13        appBar: AppBar(
14          title: Text('RK1 - Leap Year Checker'),
15        ),
16        body: Center(
17          child: LeapYearPainter(),
18        ),
19      ),
20    );
21  }
22 }
23
24 class LeapYearPainter extends StatefulWidget {
```

```

25  @override
26  _LeapYearPainterState createState() => _LeapYearPainterState();
27  }
28
29  class _LeapYearPainterState extends State<LeapYearPainter> {
30    final TextEditingController _yearController = TextEditingController();
31    int _inputYear = 0; // default
32
33    @override
34    Widget build(BuildContext context) {
35      return Column(
36        mainAxisAlignment: MainAxisAlignment.center,
37        children: <Widget>[
38          TextFormField(
39            controller: _yearController,
40            keyboardType: TextInputType.number,
41            decoration: InputDecoration(labelText: 'Enter a year'),
42            onChanged: (value) {
43              setState(() {
44                _inputYear = int.tryParse(value) ?? 0; // Use default
45                value if parsing fails
46              });
47            },
48            SizedBox(height: 40),
49            CustomPaint(
50              size: Size(100, 100),
51              painter: LeapYearCustomPainter(_inputYear),
52            ),
53          ],
54        );
55    }
56  }
57
58  class LeapYearCustomPainter extends CustomPainter {
59    final int year;
60
61    LeapYearCustomPainter(this.year);
62
63    @override
64    void paint(Canvas canvas, Size size) {
65      Paint paint = Paint();
66
67      if (year % 4 == 0 && (year % 100 != 0 || year % 400 == 0)) {
68        paint.color = Colors.green; // for leap years
69        TextPainter textPainter = TextPainter(

```

```

70         text: TextSpan(
71             text: year.toString(),
72             style: TextStyle(color: Colors.green, fontSize: 40),
73         ),
74         textDirection: TextDirection.ltr,
75         textAlign: TextAlign.center,
76     );
77
78     textPainter.layout(minWidth: 0, maxWidth: size.width);
79     textPainter.paint(canvas, Offset((size.width - textPainter.width)
80 / 2, (size.height - textPainter.height) / 2));
81 } else {
82     paint.color = Colors.red; // for non-leap years
83     TextPainter textPainter = TextPainter(
84         text: TextSpan(
85             text: year.toString(),
86             style: TextStyle(color: Colors.red, fontSize: 40),
87         ),
88         textDirection: TextDirection.ltr,
89         textAlign: TextAlign.center,
90     );
91
92     textPainter.layout(minWidth: 0, maxWidth: size.width);
93     textPainter.paint(canvas, Offset((size.width - textPainter.width)
94 / 2, (size.height - textPainter.height) / 2));
95 }
96 }
97
98 @override
99 bool shouldRepaint(LeapYearCustomPainter old) => false;
100 }

```

4 Результаты

Результат представлен на рисунках 1 - 2.

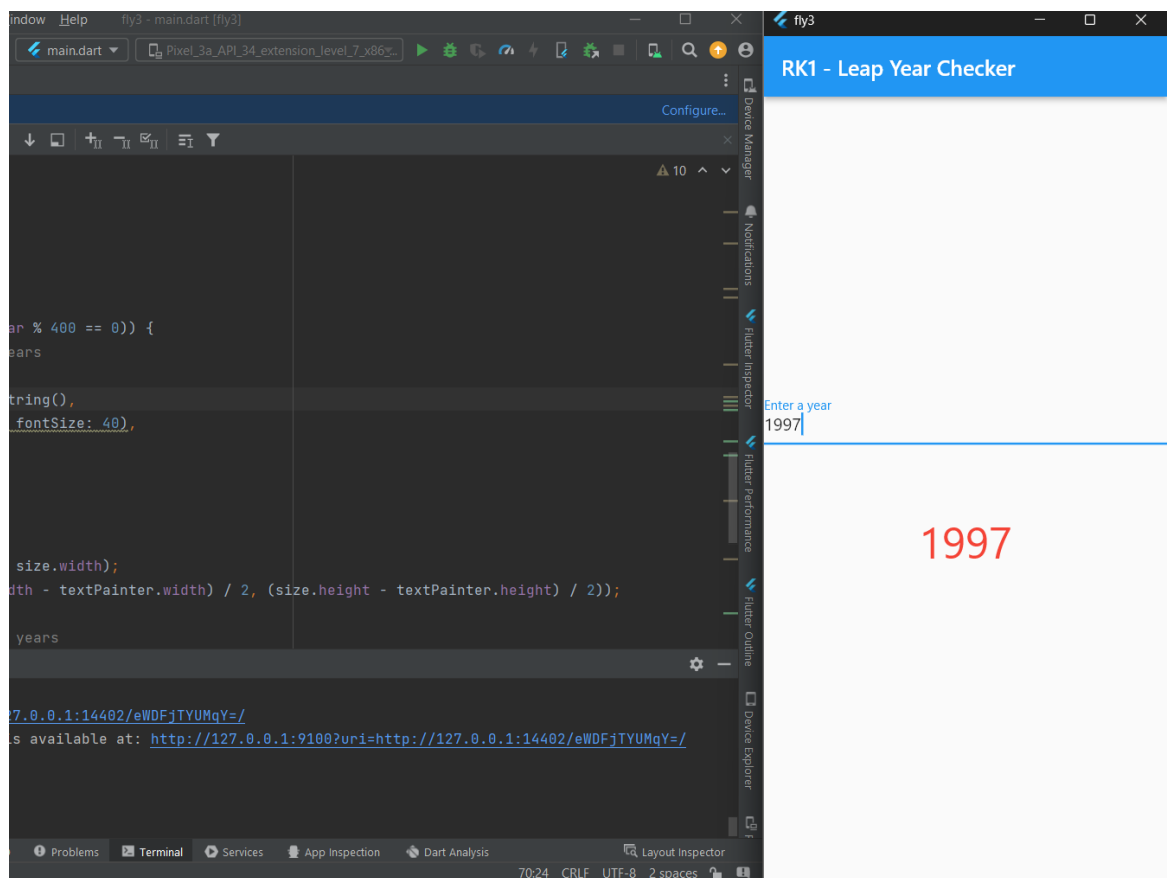


Рис. 1 — Невисокосный год

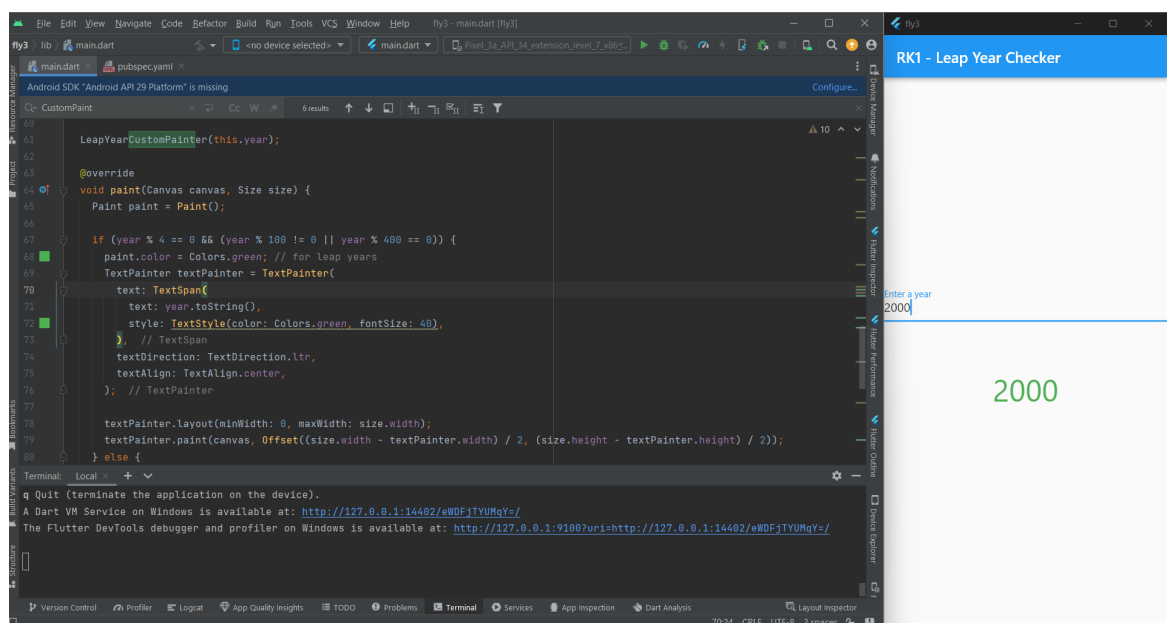


Рис. 2 — Високосный год

5 Выводы

В результате работы было создано приложение, проверяющее год на високосность.