Міністерство освіти і науки України

Національний технічний університет України

«Київський політехнічний інститут ім. І. Сікорського»

Кафедра інженерії програмного забезпечення в енергетиці

Практична робота № 1

Варіант 5

з курсу: «Кросплатформна розробка мобільних застосунків»

Виконав:

студент 4-го курсу,

групи ТВ-11

Барабаш Маріна Володимирівна

Посилання на GitHub репозиторій: https://github.com/Aylosteraa/PW\_TV11\_Barabash\_Marina\_Volodymyrivna

Перевірив:

Недашківський О.Л.

Київ 2024/2025

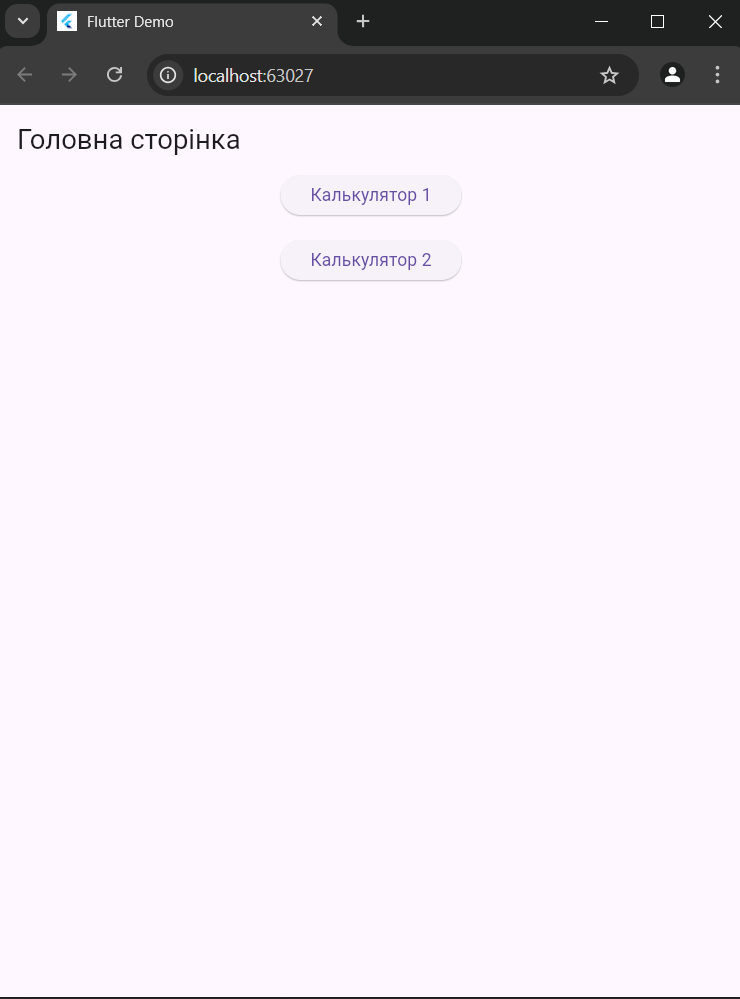
Лабораторна робота №1

**Завдання:**

Написати веб калькулятор для розрахунку складу сухої та горючої маси палива та нижчої теплоти згоряння для робочої, сухої та горючої маси за заданим кладом компонентів палива, що задаються у вигляді значень окремих компонентів типу: H P , %; C P , %; S P , %; N P , %; O P , %; W P , %; A P , %

**Хід виконання:**

Реалізуємо дизайн сторінки, на якій будуть наявні поля, де будуть здійснюватись переходи між двома калькуляторами.



**Код програми:**

import 'package:flutter/material.dart';

import 'calculators/calculator1.dart';

import 'calculators/calculator2.dart';

void main() {

  runApp(MyApp());

}

class MyApp extends StatelessWidget {

  @override

  Widget build(BuildContext context) {

    return MaterialApp(

      debugShowCheckedModeBanner: false,

      title: 'Flutter Demo',

      theme: ThemeData(

        primarySwatch: Colors.blue,

      ),

      home: HomeScreen(),

    );

  }

}

class HomeScreen extends StatelessWidget {

  @override

  Widget build(BuildContext context) {

    return Scaffold(

      appBar: AppBar(

        title: Text('Головна сторінка'),

      ),

      body: Center(

        child: Column(

          // mainAxisAlignment: MainAxisAlignment.center,

          children: [

            ElevatedButton(

              onPressed: () {

                Navigator.push(

                  context,

                  MaterialPageRoute(builder: (context) => CalculatorScreen1()),

                );

              },

              child: Text('Калькулятор 1'),

            ),

            SizedBox(height: 20),

            ElevatedButton(

              onPressed: () {

                Navigator.push(

                  context,

                  MaterialPageRoute(builder: (context) => CalculatorScreen2()),

                );

              },

              child: Text('Калькулятор 2'),

            ),

          ],

        ),

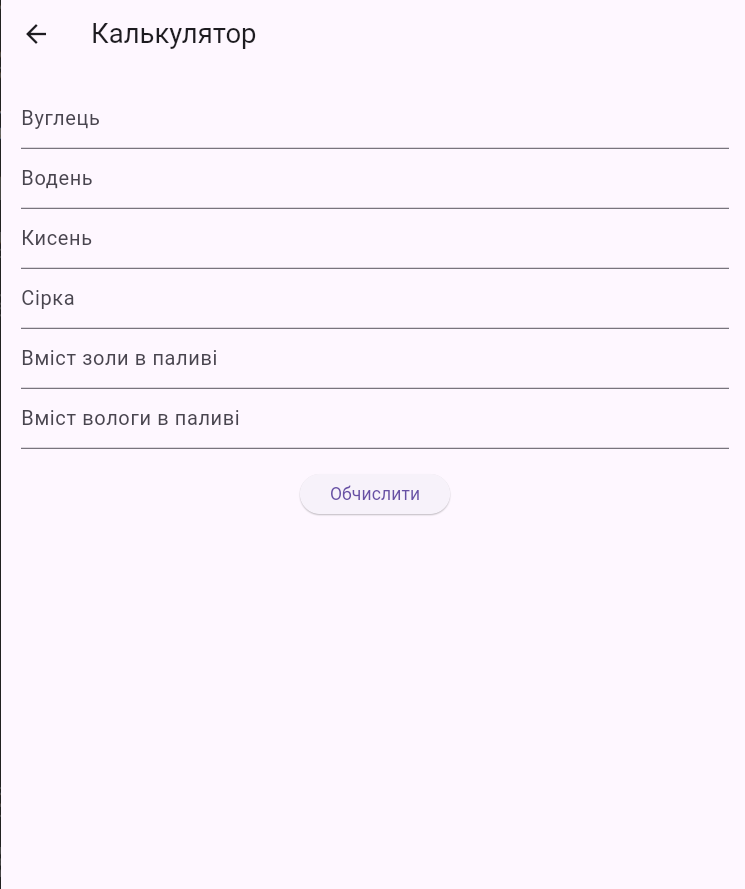
      ),

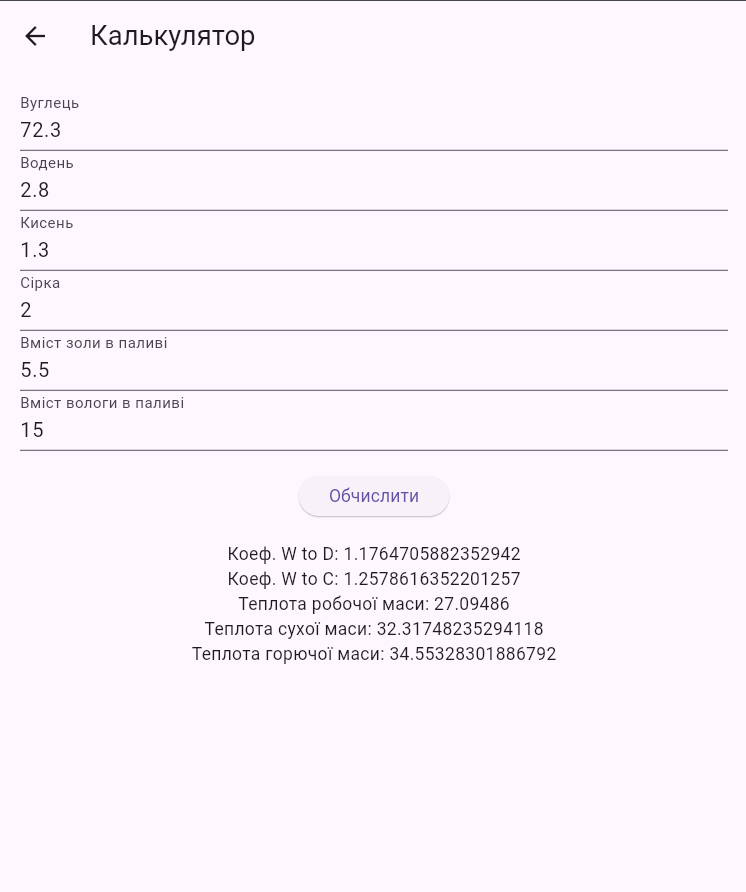
    );

  }

}

Перехід за кнопкою Калькулятор 1 на головній сторінці відкривається сторінка першого калькулятора.





**Код програми:**

import 'package:flutter/material.dart';

class CalculatorScreen1 extends StatefulWidget {

  @override

  \_CalculatorScreenState1 createState() => \_CalculatorScreenState1();

}

class \_CalculatorScreenState1 extends State<CalculatorScreen1> {

  final TextEditingController carbonController = TextEditingController();

  final TextEditingController hydrogenController = TextEditingController();

  final TextEditingController oxygenController = TextEditingController();

  final TextEditingController sulfurController = TextEditingController();

  final TextEditingController ashController = TextEditingController();

  final TextEditingController moistureController = TextEditingController();

  double coefficientWtoD = 0.0;

  double coefficientWtoC = 0.0;

  double heatWorkingMass = 0.0;

  double heatDryMass = 0.0;

  double heatCombustibleMass = 0.0;

  bool calculated = false;

  void calculate() {

    setState(() {

      double carbon = double.tryParse(carbonController.text) ?? 0.0;

      double hydrogen = double.tryParse(hydrogenController.text) ?? 0.0;

      double oxygen = double.tryParse(oxygenController.text) ?? 0.0;

      double sulfur = double.tryParse(sulfurController.text) ?? 0.0;

      double ash = double.tryParse(ashController.text) ?? 0.0;

      double moisture = double.tryParse(moistureController.text) ?? 0.0;

      coefficientWtoD = 100 / (100 - moisture);

      coefficientWtoC = 100 / (100 - moisture - ash);

      heatWorkingMass =

          (339 \* carbon + 1030 \* hydrogen - 108.8 \* (oxygen - sulfur) - 25 \* moisture) / 1000;

      heatDryMass = (heatWorkingMass + 0.025 \* moisture) \* 100 / (100 - moisture);

      heatCombustibleMass = (heatWorkingMass + 0.025 \* moisture) \* 100 / (100 - moisture - ash);

      calculated = true;

    });

  }

  @override

  Widget build(BuildContext context) {

    return Scaffold(

      appBar: AppBar(title: Text('Калькулятор')),

      body: Padding(

        padding: const EdgeInsets.all(16.0),

        child: Column(

          children: [

            TextField(controller: carbonController, decoration: InputDecoration(labelText: 'Вуглець')),

            TextField(controller: hydrogenController, decoration: InputDecoration(labelText: 'Водень')),

            TextField(controller: oxygenController, decoration: InputDecoration(labelText: 'Кисень')),

            TextField(controller: sulfurController, decoration: InputDecoration(labelText: 'Сірка')),

            TextField(controller: ashController, decoration: InputDecoration(labelText: 'Вміст золи в паливі')),

            TextField(controller: moistureController, decoration: InputDecoration(labelText: 'Вміст вологи в паливі')),

            SizedBox(height: 20),

            ElevatedButton(onPressed: calculate, child: Text('Обчислити')),

            SizedBox(height: 20),

            if (calculated) ...[

              Text('Коеф. W to D: $coefficientWtoD'),

              Text('Коеф. W to C: $coefficientWtoC'),

              Text('Теплота робочої маси: $heatWorkingMass'),

              Text('Теплота сухої маси: $heatDryMass'),

              Text('Теплота горючої маси: $heatCombustibleMass'),

            ]

          ],

        ),

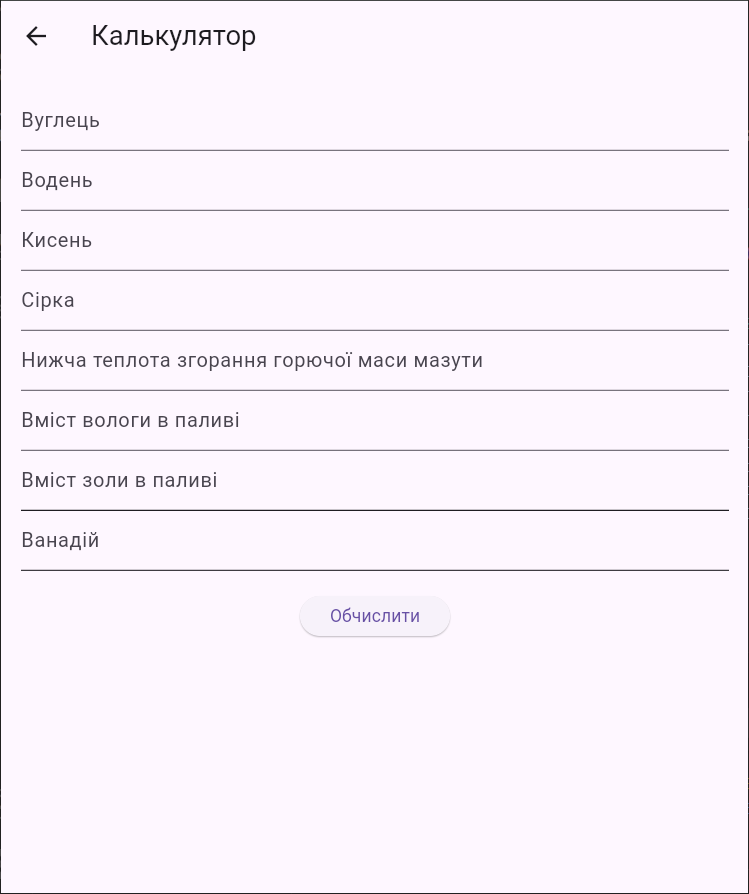
      ),

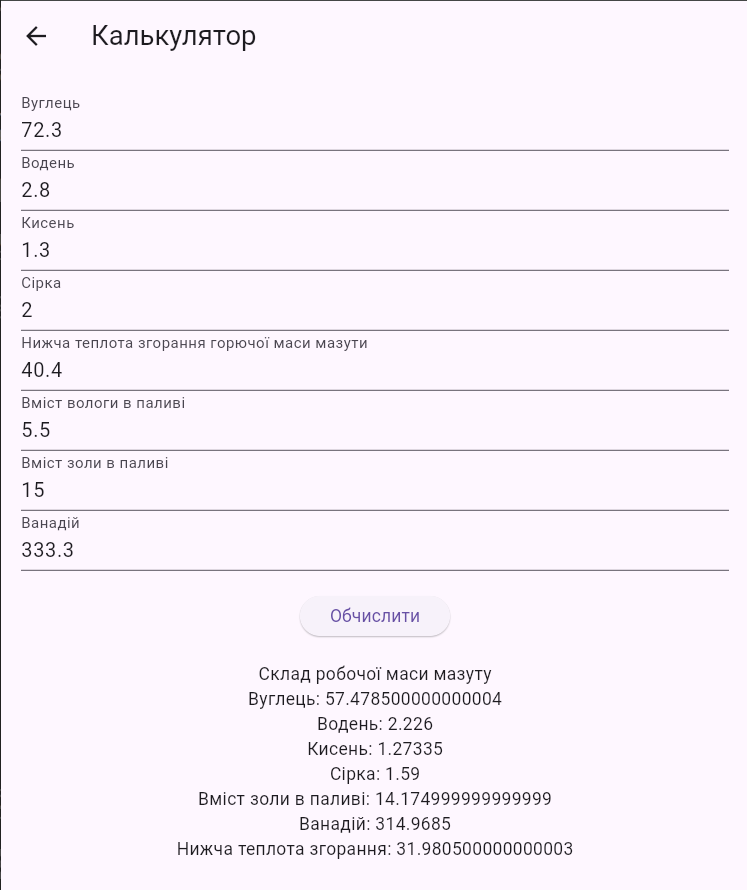
    );

  }

}

Перехід за кнопкою Калькулятор 2 на головній сторінці відкривається сторінка другого калькулятора.





**Код програми:**

import 'package:flutter/material.dart';

class CalculatorScreen2 extends StatefulWidget {

  @override

  \_CalculatorScreenState2 createState() => \_CalculatorScreenState2();

}

class \_CalculatorScreenState2 extends State<CalculatorScreen2> {

  final TextEditingController carbonController = TextEditingController();

  final TextEditingController hydrogenController = TextEditingController();

  final TextEditingController oxygenController = TextEditingController();

  final TextEditingController sulfurController = TextEditingController();

  final TextEditingController oilHeatController = TextEditingController();

  final TextEditingController fuelMoistureController = TextEditingController();

  final TextEditingController ashController = TextEditingController();

  final TextEditingController vanadiumController = TextEditingController();

  double carbonWM = 0.0;

  double hydrogenWM = 0.0;

  double oxygenWM = 0.0;

  double sulfurWM = 0.0;

  double ashWM = 0.0;

  double vanadiumWM = 0.0;

  double lowerHeatResult = 0.0;

  bool calculated = false;

  void calculate() {

    setState(() {

      double carbon = double.tryParse(carbonController.text) ?? 0.0;

      double hydrogen = double.tryParse(hydrogenController.text) ?? 0.0;

      double oxygen = double.tryParse(oxygenController.text) ?? 0.0;

      double sulfur = double.tryParse(sulfurController.text) ?? 0.0;

      double oilHeat = double.tryParse(oilHeatController.text) ?? 0.0;

      double fuelMoisture = double.tryParse(fuelMoistureController.text) ?? 0.0;

      double ash = double.tryParse(ashController.text) ?? 0.0;

      double vanadium = double.tryParse(vanadiumController.text) ?? 0.0;

      double factor1 = (100 - fuelMoisture - ash) / 100;

      double factor2 = (100 - fuelMoisture / 10 - ash / 10) / 100;

      double factor3 = (100 - fuelMoisture) / 100;

      carbonWM = carbon \* factor1;

      hydrogenWM = hydrogen \* factor1;

      oxygenWM = oxygen \* factor2;

      sulfurWM = sulfur \* factor1;

      ashWM = ash \* factor3;

      vanadiumWM = vanadium \* factor3;

      lowerHeatResult = oilHeat \* factor1 - 0.025 \* fuelMoisture;

      calculated = true;

    });

  }

  @override

  Widget build(BuildContext context) {

    return Scaffold(

      appBar: AppBar(title: Text('Калькулятор')),

      body: Padding(

        padding: const EdgeInsets.all(16.0),

        child: Column(

          children: [

            TextField(controller: carbonController, decoration: InputDecoration(labelText: 'Вуглець')),

            TextField(controller: hydrogenController, decoration: InputDecoration(labelText: 'Водень')),

            TextField(controller: oxygenController, decoration: InputDecoration(labelText: 'Кисень')),

            TextField(controller: sulfurController, decoration: InputDecoration(labelText: 'Сірка')),

            TextField(controller: oilHeatController, decoration: InputDecoration(labelText: 'Нижча теплота згорання горючої маси мазути')),

            TextField(controller: fuelMoistureController, decoration: InputDecoration(labelText: 'Вміст вологи в паливі')),

            TextField(controller: ashController, decoration: InputDecoration(labelText: 'Вміст золи в паливі')),

            TextField(controller: vanadiumController, decoration: InputDecoration(labelText: 'Ванадій')),

            SizedBox(height: 20),

            ElevatedButton(onPressed: calculate, child: Text('Обчислити')),

            SizedBox(height: 20),

            if (calculated) ...[

              Text('Склад робочої маси мазуту'),

              Text('Вуглець: $carbonWM'),

              Text('Водень: $hydrogenWM'),

              Text('Кисень: $oxygenWM'),

              Text('Сірка: $sulfurWM'),

              Text('Вміст золи в паливі: $ashWM'),

              Text('Ванадій: $vanadiumWM'),

              Text('Нижча теплота згорання: $lowerHeatResult'),

            ]

          ],

        ),

      ),

    );

  }

}

**Висновок:**

У результаті виконання практичної роботи №1 було реалізовано калькулятор, функція якого складається з розрахунку теплоти згоряння палива за допомогою мови програмування Dart. Дані вводились у поля введення та результати виводились на тій же сторінці.

Програма була реалізована на фреймворку Flutter, яка є схожою до мови програмування Kotlin з застосуванням jetpack compose.