Міністерство освіти і науки України  
Національний технічний університет України

«Київський політехнічний інститут ім. І. Сікорського»

Кафедра інженерії програмного забезпечення в енергетиці

Лабораторна робота № 2

Варіант 11(1)

з курсу: «Розробка програмного забезпечення мобільних пристроїв»

**Виконала:**  
студентка 4-го курсу,  
групи ТВ-11

Гундяк Валерія Русланівна

Посилання на GitHub репозиторій:

<https://github.com/pojuipolynu/PW_TV-11_Hundiak_Valeriia_Ruslanivna.git>

**Перевірив:**

Недашківський О.Л.

Київ 2024/2025

Лабораторна робота № 2

**Завдання:**

Написати мобільний калькулятор для розрахунку валових викидів шкідливих речовин у вигляді суспендованих твердих частинок при спалювання вугілля, мазуту та природного газу якщо розглядається:

Енергоблок з котлом, призначеним для факельного спалювання вугілля з високим вмістом летких, типу газового або довгополуменевого, з рідким шлаковидаленням. Номінальна паропродуктивність котла енергоблока становить 950 т/год, а середня фактична паропродуктивність – 760 т/год. На ньому застосовується ступенева подача повітря та рециркуляція димових газів. Пароперегрівачі котла очищуються при зупинці блока. Для уловлювання твердих частинок використовується електростатичний фільтр типу ЕГА з ефективністю золовловлення 0,985.

Установки для очищення димових газів від оксидів азоту та сірки відсутні.

За звітний період використовувалось таке паливо:

- донецьке газове вугілля марки ГР – 1.096.363 т;

- високосірчистий мазут марки 40 – 70.945 т;

- природний газ із газопроводу Уренгой-Ужгород – 84 762 тис. м3.

За даними елементного та технічного аналізу склад робочої маси вугілля наступний, %:

- вуглець (Cr) – 52,49;

- водень (Hr) – 3,50;

- кисень (Or) – 4,99;

- азот (Nr) – 0,97;

- сірка (Sr) – 2,85;

- зола (Ar) – 25,20;

- волога (Wr) – 10,00;

- леткі речовини (Vr) – 25,92.

Нижча теплота згоряння робочої маси вугілля становить 20,47 МДж/кг. Технічний аналіз уловленої золи та шлаку показав, що масовий вміст горючих речовин у леткій золі Гвин дорівнює 1,5 %, а в шлаці Гшл – 0,5 %.

За даними таблиці А.3 (додаток А) склад горючої маси мазуту настуgний, %:

- вуглець – 85,50;

- водень – 11,20;

- кисень та азот – 0,80;

- сірка – 2,50;

- нижча теплота згоряння горючої маси мазуту дорівнює 40,40 МДж/кг;

- вологість робочої маси палива – 2,00 %;

- зольність сухої маси – 0,15 %;

- вміст ванадію (V) – 333,3 мг/кг (= 2222\*0,15).

За даними таблиці А.3 (додаток А) об’ємний склад сухої маси природного газу

становить, %:

- метан (CH4) – 98,90;

- етан (C2H6) – 0,12;

- пропан (C3H8) – 0,011;

- бутан (C4H10) – 0,01;

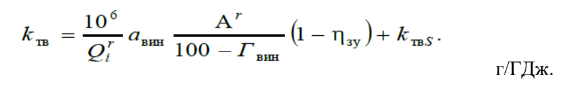
- вуглекислий газ (CO2) – 0,06;

- азот (N2) – 0,90;

- об’ємна нижча теплота згоряння газу дорівнює 33,08 МДж/м3;

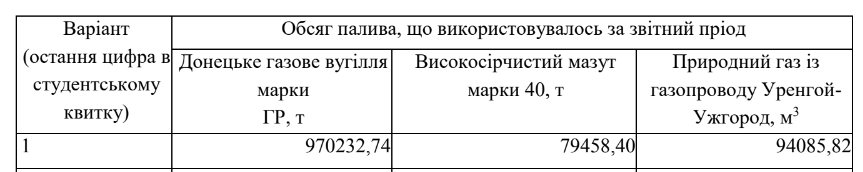
- густина – 0,723 кг/м3 при нормальних умовах.

**Хід виконання:**

Основні формули для роботи калькулятора були задані у практичному матеріалі  




Дані за варіантом:



**Код програми:**

Основний код програми має вигляд:

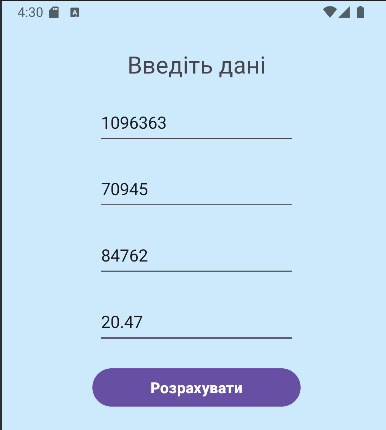
package com.example.mobilecalculator  
  
import android.os.Bundle  
import android.widget.Button  
import android.widget.EditText  
import android.widget.TextView  
import android.view.View  
import androidx.activity.enableEdgeToEdge  
import androidx.appcompat.app.AppCompatActivity  
import kotlin.math.pow  
import java.math.BigDecimal  
import java.math.RoundingMode  
  
class MainActivity : AppCompatActivity() {  
 //Декларація компонентів інтерфейсу  
 private lateinit var coalWeight: EditText  
 private lateinit var mazutWeight: EditText  
 private lateinit var gasWeight: EditText  
 private lateinit var qCoal: EditText  
 private lateinit var calculateButton: Button  
 private lateinit var buttonChangeInputs: Button  
 private lateinit var resultTextView: TextView  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 *enableEdgeToEdge*()  
 setContentView(R.layout.*activity\_main*)  
  
 // Ініціалізація комірок вводу данних  
 coalWeight = findViewById(R.id.*masacoal*)  
 mazutWeight = findViewById(R.id.*masamazut*)  
 gasWeight = findViewById(R.id.*masagas*)  
 qCoal = findViewById(R.id.*zhoraniiacoal*)  
  
 // Ініціалізація кнопок та виводу результату  
 calculateButton = findViewById(R.id.*button*)  
 buttonChangeInputs = findViewById(R.id.*button2*)  
 resultTextView = findViewById(R.id.*result*)  
  
 //Ініціалізація самого розрахунку  
 calculateButton.setOnClickListener **{** //Перевірка заповненості полів  
 if (!areInputsValid()) {  
 resultTextView.*text* = "Усі поля мають бути заповнені"  
 resultTextView.*visibility* = View.*VISIBLE* return@setOnClickListener  
 }  
 showResult(calculateResults())  
 **}** //Ініціалізація кнопки повернення назад(зміни даних)  
 buttonChangeInputs.setOnClickListener **{** toggleInputs(show = true)  
 **}** }  
  
 //Функція перевірки заповненості полів  
 private fun areInputsValid(): Boolean {  
 val coalWeightInput = coalWeight.*text*.toString()  
 val mazutWeightInput = mazutWeight.*text*.toString()  
 val gasWeightInput = gasWeight.*text*.toString()  
 val qCoalInput = qCoal.*text*.toString()  
  
 if (coalWeightInput.*isEmpty*() || mazutWeightInput.*isEmpty*() || gasWeightInput.*isEmpty*() || qCoalInput.*isEmpty*()) {  
 return false  
 }  
  
 return try {  
 coalWeightInput.*toDouble*()  
 mazutWeightInput.*toDouble*()  
 gasWeightInput.*toDouble*()  
 qCoalInput.*toDouble*()  
 true  
 } catch (e: NumberFormatException) {  
 false  
 }  
 }  
  
 //Функція розрахунку  
 private fun calculateResults(): Map<String, Any> {  
 //Отримання початкових даних  
 val gasDensity = 0.723  
 val coal = coalWeight.*text*.toString().*toDouble*()  
 val mazut = mazutWeight.*text*.toString().*toDouble*()  
 val gas = gasWeight.*text*.toString().*toDouble*() \* gasDensity  
 val q = qCoal.*text*.toString().*toDouble*()  
  
 //Розрахунок результатів на основі маси вугілля  
 val kCoal = BigDecimal((10.0.*pow*(6).toInt() / q \* 0.8 \* 25.2 / (100 - 1.5) \* (1 - 0.985))).setScale(2, RoundingMode.*HALF\_UP*).toDouble()  
 val eCoal = BigDecimal((10.0.*pow*(-6) \* kCoal \* q \* coal)).setScale(2, RoundingMode.*HALF\_UP*).toDouble()  
 //Розрахунок результатів на основі маси мазуту  
 val kmazut = BigDecimal((10.0.*pow*(6).toInt() / 39.48 \* 1 \* 0.15 / (100 - 0) \* (1 - 0.985))).setScale(2, RoundingMode.*HALF\_UP*).toDouble()  
 val emazut = BigDecimal((10.0.*pow*(-6) \* kmazut \* 39.48 \* mazut)).setScale(2, RoundingMode.*HALF\_UP*).toDouble()  
 //Розрахунок результатів на основі маси газу  
 val kGas = BigDecimal((10.0.*pow*(6).toInt() / 33.08 \* 0 \* 0 / (100 - 0) \* (1 - 0.985))).setScale(2, RoundingMode.*HALF\_UP*).toDouble()  
 val eGas = BigDecimal((10.0.*pow*(-6) \* kGas \* 33.08 \* gas)).setScale(2, RoundingMode.*HALF\_UP*).toDouble()  
 //Повернення результатів  
 return *mapOf*(  
 "kCoal" *to* kCoal,  
 "eCoal" *to* eCoal,  
 "kmazut" *to* kmazut,  
 "emazut" *to* emazut,  
 "kGas" *to* kGas,  
 "eGas" *to* eGas)  
 }  
 //Функція для виведення результату розрахунку  
 private fun showResult(results: Map<String, Any>) {  
 resultTextView.*text* = """  
 Для заданого енергоблоку і відповідним умовам роботи:  
 1. Показник емісії твердих частинок при спалюванні вугілля становитиме: ${results["kCoal"]} г/ГДж;  
 2. Валовий викид при спалюванні вугілля становитиме: ${results["eCoal"]} т.;  
 3. Показник емісії твердих частинок при спалюванні мазуту становитиме: ${results["kmazut"]} г/ГДж;  
 4. Валовий викид при спалюванні мазуту становитиме: ${results["emazut"]} т.;  
 5. Показник емісії твердих частинок при спалюванні природного газу становитиме: ${results["kGas"]}  
 г/ГДж;  
 6. Валовий викид при спалюванні природного газу становитиме: ${results["eGas"]} т..  
 """.*trimIndent*()  
  
 resultTextView.*visibility* = View.*VISIBLE* toggleInputs(show = false)  
 }  
  
 //Функція змінни видимості полів  
 private fun toggleInputs(show: Boolean) {  
 val visibility = if (show) View.*VISIBLE* else View.*GONE* coalWeight.*visibility* = visibility  
 mazutWeight.*visibility* = visibility  
 gasWeight.*visibility* = visibility  
 qCoal.*visibility* = visibility  
 calculateButton.*visibility* = visibility  
  
 buttonChangeInputs.*visibility* = if (show) View.*GONE* else View.*VISIBLE* resultTextView.*visibility* = if (show) View.*GONE* else View.*VISIBLE* }  
  
}

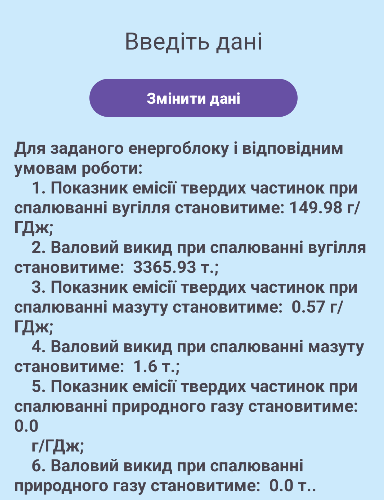
В сій частині коду відбувається отримання даних з рядків вводу інтерфейсу. Після заповнення всіх полів(відбувається перевірка чи всі поля заповнені) відбуваються обрахунки.

Обрахунки округлюються до сотих.  
  
Після виведення результатів з’являється додаткова кнопка, яка дозволяє повернутись назад.

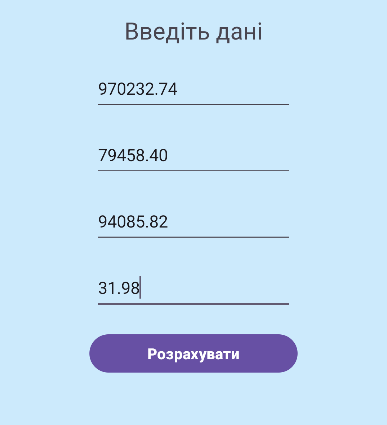
Код, що формує вигляд інтерфейсу має вигляд:  
<?xml version="1.0" encoding="utf-8"?>  
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:id="@+id/main"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:background="#cceafc"  
 tools:context=".MainActivity">  
  
 <!-- Input Section -->  
 <TextView  
 android:id="@+id/inputs"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="50dp"  
 android:text="@string/data"  
 android:textSize="25dp"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
 <EditText  
 android:id="@+id/masacoal"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="50dp"  
 android:layout\_marginTop="20dp"  
 android:ems="10"  
 android:hint="@string/coal"  
 android:inputType="text"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/inputs" />  
  
 <EditText  
 android:id="@+id/masamazut"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="50dp"  
 android:layout\_marginTop="20dp"  
 android:ems="10"  
 android:hint="@string/mazut"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/masacoal" />  
  
 <EditText  
 android:id="@+id/masagas"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="50dp"  
 android:layout\_marginTop="20dp"  
 android:ems="10"  
 android:hint="@string/gas"  
 android:inputType="text"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/masamazut" />  
  
 <EditText  
 android:id="@+id/zhoraniiacoal"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="50dp"  
 android:layout\_marginTop="20dp"  
 android:ems="10"  
 android:hint="@string/q"  
 android:inputType="text"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/masagas" />  
  
 <Button  
 android:id="@+id/button"  
 android:layout\_width="220dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="20dp"  
 android:text="@string/calculate"  
 android:textSize="16sp"  
 android:textStyle="bold"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/zhoraniiacoal" />  
  
 <Button  
 android:id="@+id/button2"  
 android:layout\_width="220dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="20dp"  
 android:text="@string/changeinputs"  
 android:textSize="16sp"  
 android:textStyle="bold"  
 android:visibility="gone"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/zhoraniiacoal" />  
  
 <TextView  
 android:id="@+id/result"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="@string/result"  
 android:visibility="gone"  
 android:textSize="18sp"  
 android:textStyle="bold"  
 android:padding="16dp"  
 android:elevation="4dp"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/button2"/>  
  
  
</androidx.constraintlayout.widget.ConstraintLayout>

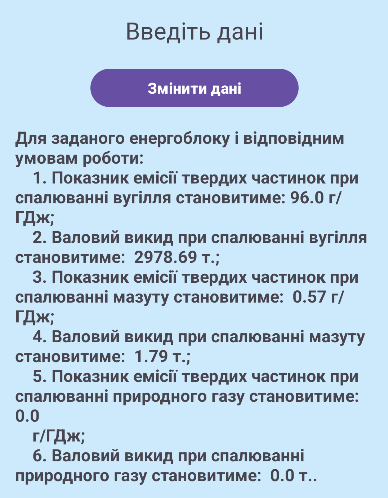
В сьому коді створюються кнопки та комірки для ведення даних.  
  
**Результат роботи програми за контрольним прикладом:**





**Результат роботи програми за варіантом (11(1)):**





**Висновок**

В результаті виконання лабораторної роботи №2 було опрацьовано алгоритм розрахунку валових викидів шкідливих речовин у вигляді суспендованих твердих частинок при спалювання вугілля, мазуту та природного газу. Також було проведено ознайомлення з Kotlin та реалізовано дані розрахунку у вигляді калькулятора Kotlin.