

Міністерство освіти і науки України
Національний технічний університет України «Київський
політехнічний інститут імені Ігоря
Сікорського» Факультет інформатики та
обчислювальної техніки Кафедра інформатики
та програмної інженерії

Звіт
до лабораторної роботи № 3 з дисципліни
«Розробка мобільних застосунків під Android»

Виконала ІК-24 Юхимець Л.

Перевірів Орленко С.П.

Київ 2025

Лабораторна робота № 3

Мета роботи

Дослідити способи збереження даних у мобільних застосунках з використанням файлової системи або бази даних та отримати практичні навички щодо роботи з різними типами сховищ інформації на платформі Android

Завдання

Необхідно розробити програму під платформу Android, яка доповнює функціональність лабораторної роботи №2 реалізуючи збереження та відображення даних за допомогою сховища

Під час натискання на кнопку «ОК» здійснюється запис результату взаємодії з інтерфейсом у сховище це може бути як файлова система так і база даних SQLite а також користувач отримує повідомлення про успішне збереження інформації

Інтерфейс програми необхідно доповнити кнопкою «Відкрити» натискання на яку здійснює перехід до іншої активності де відображається вміст даних що зберігаються у сховищі Якщо дані відсутні відображається відповідне повідомлення про відсутність даних

За бажанням можна додатково реалізувати оновлення або видалення даних зі сховища

Варіант 17

17.	Вікно містить поле введення питання/завдання, дві групи опцій (складність – три рівні, тип – теоретичне або практичне), тобто радіобатони) та кнопку «ОК». Вивести інформацію щодо вибору при натисканні на кнопку «ОК» у деяке текстове поле.
-----	--

InputFragment.kt

```
package com.example.lab3

import android.content.Context
import android.content.Intent
import android.os.Bundle
import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup
import android.widget.*
```

```

import androidx.fragment.app.Fragment
import java.io.FileOutputStream

class InputFragment : Fragment() {

    interface OnInputConfirmedListener {
        fun onInputConfirmed(resultText: String)
    }

    private var listener: OnInputConfirmedListener? = null

    override fun onAttach(context: Context) {
        super.onAttach(context)
        if (context is OnInputConfirmedListener) {
            listener = context
        } else {
            throw RuntimeException("$context must implement OnInputConfirmedListener")
        }
    }

    override fun onCreateView(inflater: LayoutInflater, container: ViewGroup?,
        savedInstanceState: Bundle?): View? {
        val view = inflater.inflate(R.layout.fragment_input, container, false)

        val inputField = view.findViewById<EditText>(R.id.etQuestion)
        val typeGroup = view.findViewById<RadioGroup>(R.id.rgType)
        val levelGroup = view.findViewById<RadioGroup>(R.id.rgDifficulty)
        val okButton = view.findViewById<Button>(R.id.btnOk)
        val openButton = view.findViewById<Button>(R.id.btnOpen)

        okButton.setOnClickListener {
            val inputText = inputField.text.toString().trim()
            val selectedTypeId = typeGroup.checkedRadioButtonId
            val selectedLevelId = levelGroup.checkedRadioButtonId

            if (inputText.isEmpty() || selectedTypeId == -1 || selectedLevelId == -1) {
                Toast.makeText(requireContext(), "Fill in all fields",
                    Toast.LENGTH_SHORT).show()
                return@setOnClickListener
            }

            val type = view.findViewById<RadioButton>(selectedTypeId)?.text
            val level = view.findViewById<RadioButton>(selectedLevelId)?.text

```

```

        val result = "Question: $inputText\nType: $type\nDifficulty: $level"

        try {
            requireContext().openFileOutput("saved_data.txt",
            Context.MODE_APPEND).use {
                it.write((result + "\n\n").toByteArray())
            }
            Toast.makeText(requireContext(), "Data saved",
            Toast.LENGTH_SHORT).show()
        } catch (e: Exception) {
            Toast.makeText(requireContext(), "Save error",
            Toast.LENGTH_SHORT).show()
        }

        listener?.onInputConfirmed(result)
    }

    openButton.setOnClickListener {
        val intent = Intent(requireContext(), StorageActivity::class.java)
        startActivity(intent)
    }

    return view
}
}

```

MainActivity.kt

```

package com.example.lab3

import android.content.Intent
import android.os.Bundle
import android.widget.Toast
import androidx.appcompat.app.AppCompatActivity
import androidx.fragment.app.commit

class MainActivity : AppCompatActivity(),
    InputFragment.OnInputConfirmedListener,
    ResultFragment.OnCancelListener {

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        if (savedInstanceState == null) {
            supportFragmentManager.commit {

```

```

        replace(R.id.fragment_container, InputFragment())
    }
}

override fun onInputConfirmed(resultText: String) {

    supportFragmentManager.commit {
        replace(R.id.fragment_container,
ResultFragment.newInstance(resultText))
        addToBackStack(null)
    }

}

override fun onCancel() {
    supportFragmentManager.popBackStack()
}
}

```

ResultFragment.kt

```

package com.example.lab3

import android.content.Context
import android.os.Bundle
import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup
import android.widget.Button
import android.widget.TextView
import androidx.fragment.app.Fragment
import com.example.lab3.R

class ResultFragment : Fragment() {

    interface OnCancelListener {
        fun onCancel()
    }

    private var cancelListener: OnCancelListener? = null

    override fun onAttach(context: Context) {
        super.onAttach(context)
        if (context is OnCancelListener) {
            cancelListener = context
        } else {
            throw RuntimeException("$context must implement OnCancelListener")
        }
    }

    override fun onCreateView(
        inflater: LayoutInflater,
        container: ViewGroup?,
        savedInstanceState: Bundle?
    ): View? {

```

```

        val view = inflater.inflate(R.layout.fragment_result, container, false)

        val resultTextView = view.findViewById<TextView>(R.id.result_text)
        val cancelButton = view.findViewById<Button>(R.id.cancel_button)

        val resultText = arguments?.getString("result_text")
        resultTextView.text = resultText

        cancelButton.setOnClickListener {
            cancelButton?.onCancel()
        }

        return view
    }

    companion object {
        fun newInstance(result: String): ResultFragment {
            val fragment = ResultFragment()
            val bundle = Bundle()
            bundle.putString("result_text", result)
            fragment.arguments = bundle
            return fragment
        }
    }
}

```

StorageActivity.kt

```

package com.example.lab3

import android.annotation.SuppressLint
import android.os.Bundle
import android.widget.Button
import android.widget.TextView
import androidx.appcompat.app.AppCompatActivity
import java.io.BufferedReader
import java.io.InputStreamReader

class StorageActivity : AppCompatActivity() {

    @SuppressLint("SetTextI18n", "MissingInflatedId")
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_storage)

        val storageText = findViewById<TextView>(R.id.tvStorage)

        try {
            val fileInput = openFileInput("saved_data.txt")
            val content = fileInput.bufferedReader().use { it.readText() }

            if (content.isBlank()) {
                storageText.text = "There are no data"
            } else {

```

```

        storageText.text = content
    }
} catch (e: Exception) {
    storageText.text = "File not found or empty"
}

val btnClear = findViewById<Button>(R.id.btnClear)
btnClear.setOnClickListener {
    try {
        openFileOutput("saved_data.txt", MODE_PRIVATE).use {
            it.write("").toByteArray()
        }
        storageText.text = "There are no data"
    } catch (e: Exception) {
        storageText.text = "Failed to clear file"
    }
}
}
}
}

```

ViewDataActivity.kt

```

package com.example.lab3

import android.annotation.SuppressLint
import android.os.Bundle
import android.widget.TextView
import androidx.appcompat.app.AppCompatActivity
import java.io.BufferedReader
import java.io.InputStreamReader

class ViewDataActivity : AppCompatActivity() {
    @SuppressLint("SetTextI18n")
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_view_data)

        val tvData = findViewById<TextView>(R.id.tvData)
        val filename = "question_data.txt"

        try {
            val inputStream = openFileInput(filename)
            val reader = BufferedReader(InputStreamReader(inputStream))
            val content = reader.readText()
            tvData.text = content
            reader.close()
        } catch (e: Exception) {
            tvData.text = "No data saved"
        }
    }
}

```

activity_main.xml

```
<FrameLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/fragment_container"
    android:layout_width="match_parent"
    android:layout_height="match_parent" />
```

activity_storage.xml

```
<?xml version="1.0" encoding="utf-8"?>
<ScrollView
xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <LinearLayout
        android:orientation="vertical"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:padding="16dp">

        <TextView
            android:id="@+id/tvStorage"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:padding="16dp"
            android:textSize="16sp"
            android:textColor="@android:color/black"
            android:text="Loading..." />

        <Button
            android:id="@+id/btnClear"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Clear"
            android:layout_gravity="center_horizontal"
            android:layout_marginTop="16dp"
            android:minHeight="48dp"
            android:paddingHorizontal="24dp"
            android:paddingVertical="12dp" />

    </LinearLayout>
</ScrollView>
```

activity_view_data.xml


```

<?xml version="1.0" encoding="utf-8"?>
<ScrollView
xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:padding="16dp">

    <TextView
        android:id="@+id/tvData"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:textSize="16sp"
        android:textColor="@android:color/black" />
</ScrollView>

```

fragment_input.xml

```

<?xml version="1.0" encoding="utf-8"?>
<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <LinearLayout
        android:orientation="vertical"
        android:padding="16dp"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:gravity="center_horizontal">

        <EditText
            android:id="@+id/etQuestion"
            android:hint="@string/hint_question"
            android:layout_width="250dp"
            android:layout_height="wrap_content"
            android:inputType="text"
            android:autofillHints="text"
            android:padding="12dp"
            android:minHeight="48dp"
            android:textColor="@android:color/black"
            android:background="@android:drawable/edit_text" />

        <TextView
            android:text="@string/select_type"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginTop="16dp"
            android:textSize="16sp"
            android:textColor="@android:color/black" />

        <RadioGroup
            android:id="@+id/rgType"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="vertical"

```

```
        android:paddingTop="8dp">

        <RadioButton
            android:id="@+id/rbTheory"
            android:text="@string/type_theory"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:minHeight="48dp"
            android:padding="8dp"
            android:textColor="@android:color/black" />

        <RadioButton
            android:id="@+id/rbPractice"
            android:text="@string/type_practice"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:minHeight="48dp"
            android:padding="8dp"
            android:textColor="@android:color/black" />
    </RadioGroup>

    <TextView
        android:text="@string/select_difficulty"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:paddingTop="16dp"
        android:textSize="16sp"
        android:textColor="@android:color/black" />

    <RadioGroup
        android:id="@+id/rgDifficulty"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical"
        android:paddingTop="8dp">

        <RadioButton
            android:id="@+id/rbEasy"
            android:text="@string/difficulty_easy"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:minHeight="48dp"
            android:padding="8dp"
            android:textColor="@android:color/black" />

        <RadioButton
            android:id="@+id/rbMedium"
            android:text="@string/difficulty_medium"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:minHeight="48dp"
            android:padding="8dp"
            android:textColor="@android:color/black" />

        <RadioButton
            android:id="@+id/rbHard"
            android:text="@string/difficulty_hard"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
```

```

        android:minHeight="48dp"
        android:padding="8dp"
        android:textColor="@android:color/black" />
</RadioGroup>

<Button
    android:id="@+id/btnOk"
    android:text="@string/ok"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="24dp"
    android:minHeight="48dp"
    android:paddingHorizontal="24dp"
    android:paddingVertical="12dp" />

<Button
    android:id="@+id/btnOpen"
    android:text="@string/open"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="12dp"
    android:minHeight="48dp"
    android:paddingHorizontal="24dp"
    android:paddingVertical="12dp" />

</LinearLayout>
</ScrollView>

```

fragment_result.xml

```
<?xml version="1.0" encoding="utf-8"?>
```

```

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"

    android:orientation="vertical"

    android:layout_width="match_parent"

    android:layout_height="wrap_content"

    android:padding="16dp">

    <TextView

        android:id="@+id/result_text"

        android:layout_width="match_parent"

        android:layout_height="wrap_content"

        android:text="@string/result"

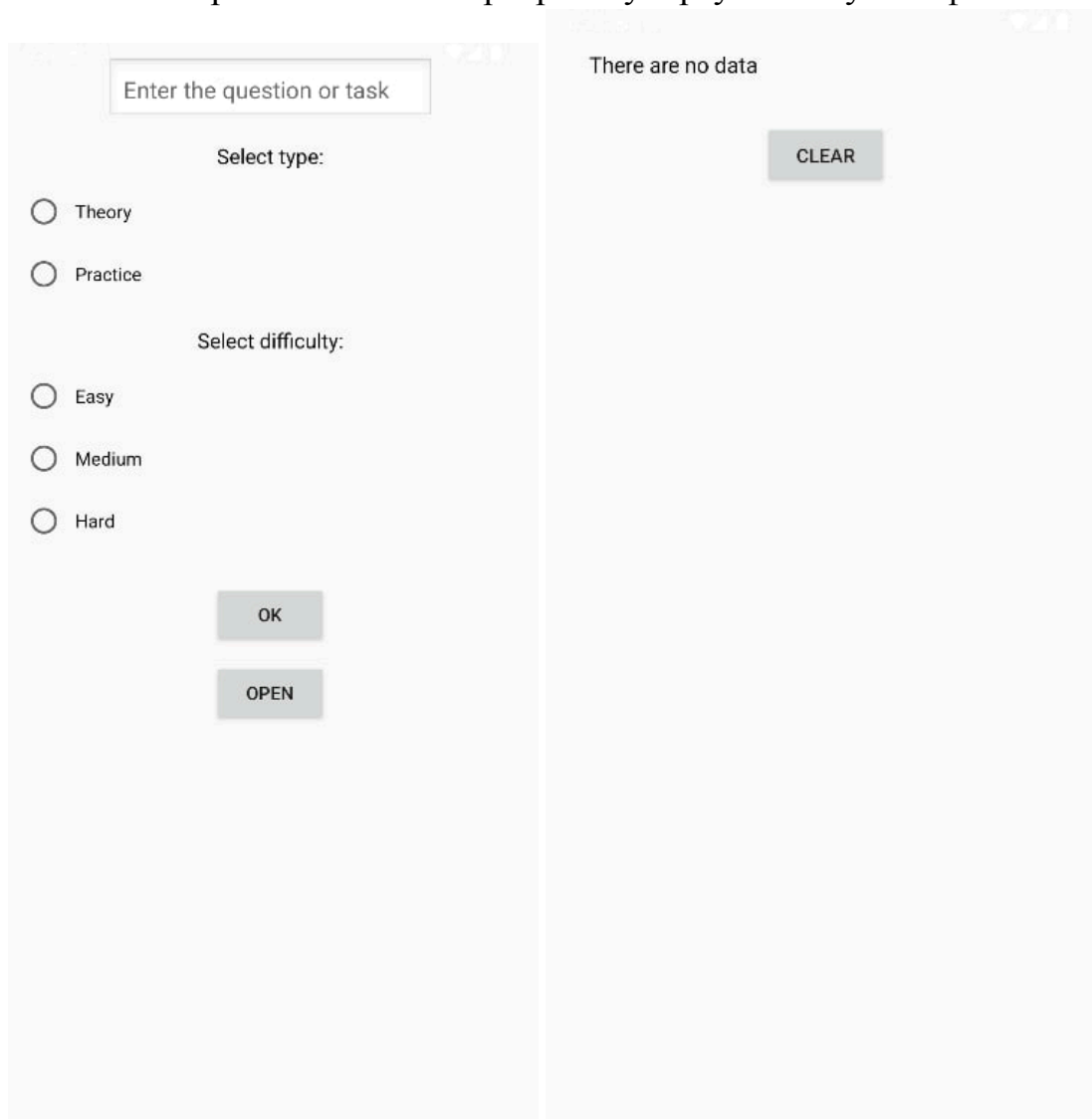
        android:textSize="18sp"

        android:padding="8dp"/>

```

```
<Button  
    android:id="@+id/cancel_button"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="@string/cancel"  
    android:layout_marginTop="16dp"/>  
</LinearLayout>
```

Скріни виконання програми у віртуальному телефоні



Lesia

Select type:

- ☒ Theory
- ☐ Practice

Select difficulty:


- ☐ Easy
- ☒ Medium
- ☐ Hard

OK

OPEN

Question: Lesia
Type: Theory
Difficulty: Medium

CANCEL

 Data saved

Question: Lesia
Type: Theory
Difficulty: Medium

CLEAR

There are no data

CLEAR