



Міністерство освіти і науки України
Національний технічний університет України
“Київський політехнічний інститут імені Ігоря Сікорського”
Факультет інформатики та обчислювальної техніки
Кафедра інформаційних систем та технологій

Лабораторна робота №3
Розробка мобільних застосунків під Android
«ДОСЛІДЖЕННЯ СПОСОБІВ ЗБЕРЕЖЕННЯ ДАНИХ»
Варіант 6

Виконала:
студентка групи ІА-21
Антропова К.Д.

Київ 2025

Мета роботи: дослідити способи збереження даних (база даних, файлова система, тощо) та отримати практичні навички щодо використання сховищ даних.

Завдання на лабораторну роботу

Написати програму під платформу Андроїд, яка доповнює програму, що розроблена за лабораторною роботою 2, роботою зі сховищами.

Тобто при натисканні на кнопку «ОК» додатково:

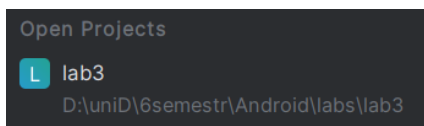
- здійснюється запис результату взаємодії з інтерфейсом до сховища (файл або базу даних);
- користувач інформується відповідним повідомленням щодо успішності запису.

Також інтерфейс необхідно доповнити кнопкою «Відкрити», натискання на яку призводить до переходу на іншу Діяльність, у якій відображається вміст даних, що зберігаються у сховищі. Якщо дані відсутні (сховище порожнє) відобразити відповідне повідомлення. За бажанням можна додатково реалізувати оновлення та видалення даних зі сховища.

| | |
|----|--|
| 6. | Вікно введення паролю містить: текстове поле, дві опції (радіо-батони) режиму введення (відображати введені символи або відображати зірочки) та кнопку «ОК». Вивести введений пароль при натисканні на кнопку «ОК» у деяке текстове поле. |
|----|--|

Хід роботи

1. Створюємо проект



2. В класі MainActivity пишемо такий код:

```
package com.example.lab3

import android.content.Intent
import android.os.Bundle
import androidx.activity.ComponentActivity
import androidx.activity.compose.setContent
import androidx.compose.material3.MaterialTheme
import androidx.compose.material3.Surface
import androidx.compose.ui.Modifier
import androidx.compose.ui.platform.ComposeView
import androidx.fragment.app.Fragment
import androidx.fragment.app.FragmentActivity
import androidx.fragment.app.commit
import java.io.File
import java.io.FileOutputStream
import java.io.IOException
import java.text.SimpleDateFormat
```

```

import java.util.Date
import java.util.Locale

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

        if (savedInstanceState == null) {
            loadFragment(InputFragment(), R.id.inputFragmentContainer)
        }

        private fun loadFragment(fragment: Fragment, containerId: Int) {
            supportFragmentManager.commit {
                replace(containerId, fragment)
            }
        }

        fun showResult(result: String) {
            val resultFragment = ResultFragment.newInstance(result)
            supportFragmentManager.commit {
                replace(R.id.resultFragmentContainer, resultFragment)
                addToBackStack(null)
            }
        }

        fun clearInput() {
            val inputFragment =
                supportFragmentManager.findFragmentById(R.id.inputFragmentContainer) as?
                InputFragment
            inputFragment?.clearInput()
        }

        fun savePassword(password: String): Boolean {
            try {
                val timestamp = SimpleDateFormat("yyyy-MM-dd HH:mm:ss",
                    Locale.getDefault())
                    .format(Date())
                val entry = "$timestamp: $password\n"

                val file = File(filesDir, "passwords.txt")
                FileOutputStream(file, true).use {
                    it.write(entry.toByteArray())
                }
                return true
            } catch (e: IOException) {
                e.printStackTrace()
                return false
            }
        }

        fun openStoredData() {
            val intent = Intent(this, StoredDataActivity::class.java)
            startActivity(intent)
        }
    }
}

```

3. В класі InputFragment пишемо такий код:

```

package com.example.lab3

```

```

import android.os.Bundle
import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup
import androidx.compose.foundation.layout.*
import androidx.compose.material3.*
import androidx.compose.runtime.*
import androidx.compose.ui.Modifier
import androidx.compose.ui.platform.ComposeView
import androidx.compose.ui.text.input.PasswordVisualTransformation
import androidx.compose.ui.text.input.VisualTransformation
import androidx.compose.ui.unit.dp
import androidx.fragment.app.Fragment
import androidx.fragment.app.commit

class InputFragment : Fragment() {
    override fun onCreateView(
        inflater: LayoutInflater,
        container: ViewGroup?,
        savedInstanceState: Bundle?
    ): View {
        return ComposeView(requireContext()).apply {
            setContent {
                MaterialTheme {
                    InputScreen()
                }
            }
        }
    }

    @Composable
    fun InputScreen() {
        var password by remember { mutableStateOf("") }
        var showPassword by remember { mutableStateOf(false) }
        var showDialog by remember { mutableStateOf(false) }
        var showSaveSuccessDialog by remember { mutableStateOf(false) }

        Column(
            modifier = Modifier
                .fillMaxSize()
                .padding(16.dp),
            verticalArrangement = Arrangement.spacedBy(16.dp)
        ) {
            TextField(
                value = password,
                onValueChange = { password = it },
                label = { Text("Введіть пароль") },
                visualTransformation = if (showPassword)
                    VisualTransformation.None
                else
                    PasswordVisualTransformation(),
                modifier = Modifier.fillMaxWidth()
            )

            Column {
                Row {
                    RadioButton(
                        selected = showPassword,
                        onClick = { showPassword = true }
                    )
                    Text("Показати символи", modifier =
Modifier.padding(start = 8.dp))
                }
                Row {

```

```

        RadioButton(
            selected = !showPassword,
            onClick = { showPassword = false }
        )
        Text("Приховати символи", modifier =
Modifier.padding(start = 8.dp))
    }

    Row(
        modifier = Modifier.fillMaxWidth(),
        horizontalArrangement = Arrangement.spacedBy(8.dp)
    ) {
        Button(
            onClick = {
                if (password.isEmpty()) {
                    showDialog = true
                } else {
                    val success = (requireActivity() as
MainActivity).savePassword(password)
                    if (success) {
                        showSaveSuccessDialog = true
                    }
                    (requireActivity() as
MainActivity).showResult("Введений пароль: $password")
                }
            },
            modifier = Modifier.weight(1f)
        ) {
            Text("OK")
        }

        Button(
            onClick = {
                (requireActivity() as
MainActivity).openStoredData()
            },
            modifier = Modifier.weight(1f)
        ) {
            Text("Відкрити")
        }
    }

    if (showDialog) {
        AlertDialog(
            onDismissRequest = { showDialog = false },
            title = { Text("Попередження") },
            text = { Text("Будь ласка, введіть пароль") },
            confirmButton = {
                TextButton(onClick = { showDialog = false }) {
                    Text("OK")
                }
            }
        )
    }

    if (showSaveSuccessDialog) {
        AlertDialog(
            onDismissRequest = { showSaveSuccessDialog = false },
            title = { Text("Успіх") },
            text = { Text("Дані успішно збережено") },
            confirmButton = {
                TextButton(onClick = { showSaveSuccessDialog =
false }) {

```

```

        Text("OK")
    }
}
)
}
}

fun clearInput() {
    parentFragmentManager.commit {
        replace(R.id.inputFragmentContainer, InputFragment())
    }
}
}
}

```

4. В класі ResultFragment пишемо такий код:

```

package com.example.lab3

import android.os.Bundle
import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup
import androidx.compose.foundation.layout.*
import androidx.compose.foundation.rememberScrollState
import androidx.compose.foundation.verticalScroll
import androidx.compose.material3.*
import androidx.compose.runtime.Composable
import androidx.compose.ui.Modifier
import androidx.compose.ui.platform.ComposeView
import androidx.compose.ui.unit.dp
import androidx.fragment.app.Fragment

class ResultFragment : Fragment() {
    companion object {
        private const val ARG_RESULT = "result"

        fun newInstance(result: String): ResultFragment {
            return ResultFragment().apply {
                arguments = Bundle().apply {
                    putString(ARG_RESULT, result)
                }
            }
        }
    }

    override fun onCreateView(
        inflater: LayoutInflater,
        container: ViewGroup?,
        savedInstanceState: Bundle?
    ): View {
        return ComposeView(requireContext()).apply {
            setContent {
                MaterialTheme {
                    ResultScreen()
                }
            }
        }
    }

    @Composable
    fun ResultScreen() {

```

```

        val result = arguments?.getString(ARG_RESULT) ?: ""

        Column(
            modifier = Modifier
                .fillMaxWidth()
                .padding(16.dp)
                .verticalScroll(rememberScrollState()),
            verticalArrangement = Arrangement.spacedBy(16.dp)
        ) {
            Text(text = result)
            Button(
                onClick = {
                    (requireActivity() as MainActivity).clearInput()
                    parentFragmentManager.popBackStack()
                },
                modifier = Modifier.fillMaxWidth()
            ) {
                Text("Cancel")
            }
        }
    }
}

```

5. В StoredDataActivity.kt пишем:

```

package com.example.lab3

import android.os.Bundle
import androidx.activity.ComponentActivity
import androidx.activity.compose.LocalOnBackPressedDispatcherOwner
import androidx.activity.compose.setContent
import androidx.compose.foundation.layout.*
import androidx.compose.foundation.rememberScrollState
import androidx.compose.foundation.verticalScroll
import androidx.compose.material3.*
import androidx.compose.runtime.Composable
import androidx.compose.ui.Alignment
import androidx.compose.ui.Modifier
import androidx.compose.ui.unit.dp
import java.io.File

class StoredDataActivity : ComponentActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContent {
            MaterialTheme {
                Surface(
                    modifier = Modifier.fillMaxSize(),
                    color = MaterialTheme.colorScheme.background
                ) {
                    StoredDataScreen(readStoredData())
                }
            }
        }
    }

    private fun readStoredData(): List<String> {
        val file = File(filesDir, "passwords.txt")
        return if (file.exists()) {
            file.readlines()
        } else {
            emptyList()
        }
    }
}

```

```

    }
}

@Composable
fun StoredDataScreen(data: List<String>) {
    val backDispatcher =
        LocalOnBackPressedDispatcherOwner.current?.onBackPressedDispatcher

    Column(
        modifier = Modifier
            .fillMaxSize()
            .padding(16.dp),
        horizontalAlignment = Alignment.CenterHorizontally
    ) {
        Text(
            text = "Збережені паролі",
            style = MaterialTheme.typography.headlineMedium,
            modifier = Modifier.padding(bottom = 16.dp)
        )

        if (data.isEmpty()) {
            Box(
                modifier = Modifier.fillMaxSize(),
                contentAlignment = Alignment.Center
            ) {
                Text(text = "Немає збережених даних", style =
MaterialTheme.typography.bodyLarge)
            }
        } else {
            Column(
                modifier = Modifier
                    .fillMaxWidth()
                    .verticalScroll(rememberScrollState()),
                verticalArrangement = Arrangement.spacedBy(8.dp)
            ) {
                data.forEach { entry ->
                    Card(
                        modifier = Modifier.fillMaxWidth()
                    ) {
                        Text(
                            text = entry,
                            modifier = Modifier.padding(16.dp)
                        )
                    }
                }
            }
        }

        Spacer(modifier = Modifier.height(16.dp))

        Button(
            onClick = {
                backDispatcher?.onBackPressed()
            },
            modifier = Modifier.fillMaxWidth()
        ) {
            Text("Назад")
        }
    }
}

```

6. В activity_main.xml також додаємо код:


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

    <FrameLayout android:id="@+id/inputFragmentContainer"
        android:layout_width="match_parent" android:layout_height="0dp"
            android:layout_weight="1" />

    <FrameLayout android:id="@+id/resultFragmentContainer"
        android:layout_width="match_parent" android:layout_height="0dp"
            android:layout_weight="1" />

</LinearLayout>
```

7. В strings.xml додаємо такий код:

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <string name="app_name">Lab3</string>
    <string name="enter_password">Введіть пароль</string>
    <string name="show_password">Показати символи</string>
    <string name="hide_password">Приховати символи</string>
    <string name="ok_button">OK</string>
    <string name="cancel_button">Cancel</string>
    <string name="open_button">Відкрити</string>
    <string name="back_button">Назад</string>
    <string name="save_success">Дані успішно збережено</string>
    <string name="no_data">Немає збережених даних</string>
    <string name="stored_passwords">Збережені паролі</string>
</resources>
```

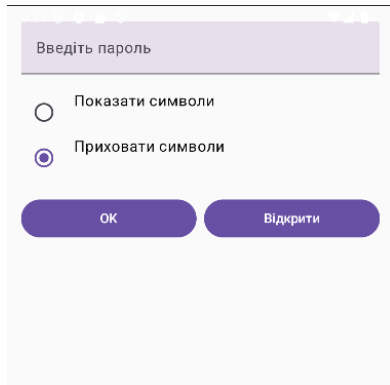
8. В AndroidManifest.xml додаємо:

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.lab3">

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.Lab3">
        <activity
            android:name=".MainActivity"
            android:exported="true">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER"
            />
            </intent-filter>
        </activity>
        <activity
            android:name=".StoredDataActivity"
            android:exported="false" />
    </application>

</manifest>
```

Заходимо в програму і бачимо вказані у завданні компоненти програми, а саме поле для введення паролю, кнопки режимів відображення, кнопка «ОК» та кнопка «Відкрити»:



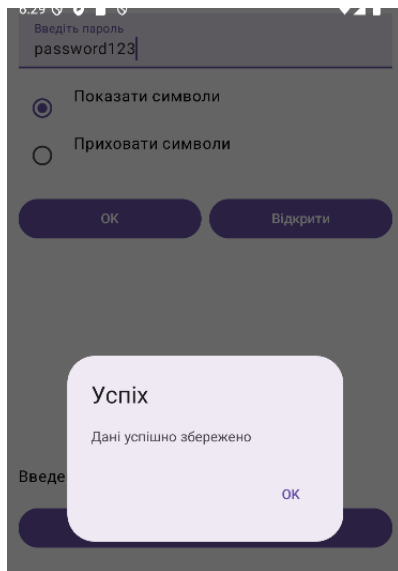
Введіть пароль

☐ Показати символи

☒ Приховати символи

ОК Відкрити

Після введення паролю при натисненні кнопки «ОК» введений пароль виводиться вікно з повідомленням про успішне збереження паролю:



Введіть пароль
password123

☒ Показати символи

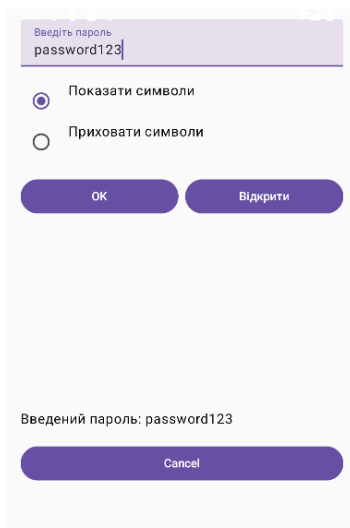
☐ Приховати символи

ОК Відкрити

Успіх
Дані успішно збережено

Введе... ОК

При натисненні «ОК» це вікно закривається, маємо виведений пароль і кнопку «Cancel»:



Введіть пароль
password123

☒ Показати символи

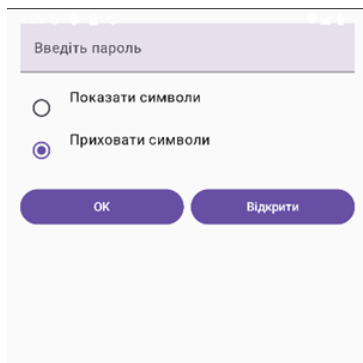
☐ Приховати символи

ОК Відкрити

Введений пароль: password123

Cancel

При натисненні кнопки «Cancel» поле введення пароллю очищається, а виведений рядок з паролем і сама кнопка зникають:



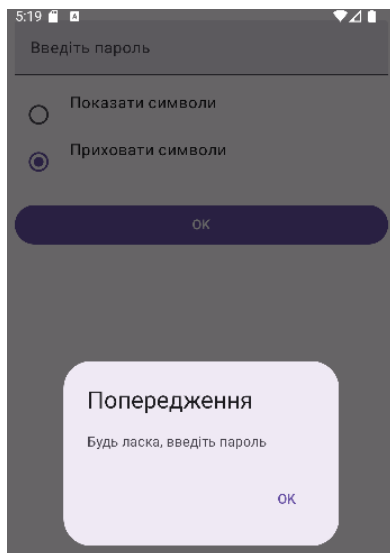
Введіть пароль

☐ Показати символи

☒ Приховати символи

ОК Відкрити

Якщо не було введено жодного символу в вікно введення пароллю, але натиснуто кнопку «ОК», відобразиться спливаюче вікно з повідомленням завершити введення всіх даних:



5:19 Введіть пароль

☐ Показати символи

☒ Приховати символи

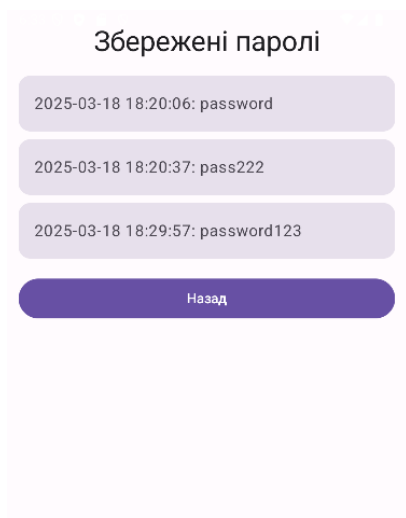
ОК

Попередження

Будь ласка, введіть пароль

ОК

При натисненні кнопки «Відкрити» відкривається вікно з збереженими паролями:



Збережені паролі

2025-03-18 18:20:06: password

2025-03-18 18:20:37: pass222

2025-03-18 18:29:57: password123

Назад

При натисненні кнопки «Назад» ми повертаємось на головний екран.

Отже, програма працює відповідно до вимог завдання.

Висновок : У ході виконання цієї лабораторної роботи я дослідила способи збереження даних (база даних, файлова система, тощо), оновила розроблений в попередній лабораторній застосунок відповідно до вимог та отримала практичні навички щодо використання сховищ даних.