

## Enunciat

Aprofitant el projecte de l'última pràctica, amplia'l i fes que quan fem clic a una opció de l'spinner, mostri la informació de l'element seleccionat i la posició on es troba a l'spinner com a text a un altre activity

A aquesta nova Activity afegeix-li també un botó per tornar enrere.

## Entrega

Document .pdf amb captures dels canvis que has hagut de fer i el codi de les Activities i el xml creats. Inclou el link del projecte compartit al DRIVE i el link de l' apk compilada.

Link al Github del projecte.

[https://github.com/cristianjimenezhernandezdev/Android\\_Programacio\\_DAM2b.git](https://github.com/cristianjimenezhernandezdev/Android_Programacio_DAM2b.git)

Link al drive

<https://drive.google.com/file/d/1UUDkIRwaJUWjYnyCRKVkTQbqxG1Du3pb/view?usp=sharing>

1- Com fem per fer accessible l'objecte Prefs per tothom?

2- Quina és la instrucció que fem servir per accedir a les SharedPreferences?

3- Quina és la instrucció que fem servir per eliminar les preferències guardades?

4- Per què hem fet una classe del tipus Application? És imprescindible en aquest cas?

Respostes.

1

Fem accessible l'objecte Prefs per tothom creant-lo a dins d'un companion object dins la classe UserVipApplication. D'aquesta manera es comporta com una instància única i global que qualsevol Activity pot consultar sense haver de crear-la cada vegada. Això garanteix que les preferències s'inicialitzen un sol cop quan arrenca l'aplicació.

2

La instrucció que fem servir per accedir a les SharedPreferences és val storage = context.getSharedPreferences(SHARED\_NAME, 0) tot i que en el projecte s'accedeix indirectament fent servir Prefs, que internament usa aquesta instrucció per llegir i guardar dades.

3

La instrucció per eliminar totes les preferències guardades és prefs.wipe() o, si ho féssim manualment, edit().clear().apply(). Aquest codi esborra totes les claus i valors que hi ha i deixa les Shared Preferences en buides.

4

Fem una classe del tipus Application perquè necessitem inicialitzar l'objecte Prefs abans que qualsevol Activity s'executi. Això permet tenir les preferències actives des del principi i així no cal d'inicialitzar-les manualment a cada pantalla. En aquest cas no és estrictament imprescindible, però sí ajuda a manejar l'aplicació de manera més correcte.

Annex de codi:

Codi

```
package com.example.preferences

import android.content.Intent
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.view.View
import android.widget.AdapterView
import android.widget.ArrayAdapter
import android.widget.Toast
import com.example.preferences.PreferencesApp.Companion.prefs
import com.example.preferences.databinding.ActivityMainBinding

class MainActivity : AppCompatActivity(), AdapterView.OnItemSelectedListener {

    private lateinit var binding: ActivityMainBinding
    private var pos = 0
    private var textSeleccionat = ""

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        binding = ActivityMainBinding.inflate(layoutInflater)
        setContentView(binding.root)

        initUI()
        initSpinner()
        checkUserValues()

    }

    fun checkUserValues() {
        if (prefs.getName().isNotEmpty()) {
            goToDetail()
        }
    }
}
```

```
        }  
    }  
  
    //Configuració del Spinner  
    //Creem un array listOf amb els valors del Spinner  
    private fun initSpinner() {  
        val interessos = listOf(  
            "Formal",  
            "Informal",  
        )  
  
        val adapterInteressos = ArrayAdapter(  
            this,  
            android.R.layout.simple_spinner_item,  
            interessos  
        )  
        adapterInteressos.setDropDownViewResource(  
            android.R.layout.simple_spinner_dropdown_item  
        )  
  
        binding.interessosSpinner.adapter = adapterInteressos  
        binding.interessosSpinner.onItemSelectedListener = this  
  
        val esVipGuardat = prefs.getVip()  
        val index = if (esVipGuardat) 0 else 1 // 0 = Formal, 1 = Informal  
  
        binding.interessosSpinner.setSelection(index)  
        pos = index  
        textSeleccionat = interessos[index]  
    }  
  
    fun initUI() {  
        binding.btnContinuar.setOnClickListener { accesToDetail() }  
    }  
  
    // guardar a SharedPreferences  
    fun accesToDetail() { //comprovo que els camps no estiguin buits  
        if (binding.etName.text.toString().isNotEmpty() &&  
            binding.etCognom.text.toString().isNotEmpty() &&  
            binding.etLink.text.toString().isNotEmpty())  
        {  
            prefs.saveName(binding.etName.text.toString())  
        }  
    }
```

```
prefs.saveCognom(binding.etCognom.text.toString())
prefs.saveLink(binding.etLink.text.toString())
prefs.saveVIP(textSeleccionat == "Formal")
goToDetail()
} else {//si algun dels camps està buit, mostrem un missatge amb un toast
    Toast.makeText(
        this,
        "Les dades no poden ser buides",
        Toast.LENGTH_SHORT
    ).show()
}
}

// Quan es selecciona un element del Spinner es fa el itemselected
override fun onItemSelected(
    parent: AdapterView<*>?,
    view: View?,
    position: Int,
    id: Long
) {
    if (parent?.id == R.id.interessos_spinner) {
        pos = position
        textSeleccionat = parent.getItemAtPosition(position).toString()
    }
}

override fun onNothingSelected(parent: AdapterView<*>?) {

}

// 
fun goToDetail() {
    if (prefs.getVip()) {
        // modo formal va a MainActivity3
        startActivity(Intent(this, MainActivity3::class.java))
    } else {
        // modo normal va a MainActivity2 amb les dades del Spinner
        val intent = Intent(this, MainActivity2::class.java)
        intent.putExtra("posicio", pos)
        intent.putExtra("text", textSeleccionat)
        startActivity(intent)
    }
}
}
```

```
package com.example.preferences

import android.app.Application

//Aquesta aplicació s'executarà abans que l'activity Mai un
class PreferencesApp:Application() {

    //crearem una instància de prefs perquè tothom pugui accedir
    companion object{
        lateinit var prefs: Prefs
    }
    //Reimplementem l'oncreate
    override fun onCreate() {
        super.onCreate()

        prefs = Prefs(applicationContext)

    }
}
```

```
package com.example.preferences

import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import androidx.core.content.ContextCompat
import com.example.preferences.PreferencesApp.Companion.prefs
import com.example.preferences.databinding.ActivityMain3Binding

class MainActivity2 : AppCompatActivity() {

    // Utilitzem el binding generat pel layout 'activity_main3.xml' (ActivityMain3Binding)
    // tot i que estem dins de la classe MainActivity2. amb això reutilitzem.
    private lateinit var binding: ActivityMain3Binding

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        binding = ActivityMain3Binding.inflate(layoutInflater)
        val view = binding.root
        setContentView(view)
        initUI()
    }

    fun initUI(){
        // Esborra totes les dades guardades (wipe)
```

```
// i torna a la pantalla anterior.  
binding.btnBack.setOnClickListener{  
    prefs.wipe()  
    onBackPressed()  
}  
  
// Recuperem les dades a SharedPreferences  
// a través del 'prefs'.  
val userName = prefs.getName()  
val userCognom = prefs.getCognom()  
val userLink = prefs.getLink()  
  
// Actualitzem la interfície amb les dades recuperades.  
  
binding.tvName.text = "Informal $userName"  
binding.tvCognom.text = "Cognoms Informal: $userCognom"  
binding.tvLink.text = "Link Informal: $userLink"  
  
// Canviem el color de fons si és formal reutilitzo la funció del vip de l'exemple  
if(prefs.getVip())  
{  
    setVIPColorBackground()  
}  
}  
  
// Canvia el color de fons  
fun setVIPColorBackground()  
{  
    binding.container.setBackgroundColor(ContextCompat.getColor(this,R.color.purple_200))  
}  
}  
  
<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
    xmlns:tools="http://schemas.android.com/tools"  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"  
    tools:context=".MainActivity3"  
    android:id="@+id/container"  
    android:orientation="vertical"  
    android:padding="16dp">  
  
<TextView
```

```
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/tvName"
    android:textSize="20sp"
    tools:text="Benvingut Nom" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/tvCognom"
    android:textSize="20sp"
    tools:text="Cognoms: [Cognom]" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/tvLink"
    android:textSize="20sp"
    tools:text="Link: [Link Github]" />

<Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="32dp"
    android:text="Tancar Sessió"
    android:id="@+id	btnBack" />

</LinearLayout>
```

```
package com.example.preferences

import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import androidx.core.content.ContextCompat
import com.example.preferences.PreferencesApp.Companion.prefs
import com.example.preferences.databinding.ActivityMain3Binding

class MainActivity3 : AppCompatActivity() {
    private lateinit var binding: ActivityMain3Binding

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        binding = ActivityMain3Binding.inflate(layoutInflater)
        val view = binding.root
```

```
setContentView(view)
initUI()
}

fun initUI(){
    // En clicar enrere, forcem que es borri tot (wipe)

    binding.btnExit.setOnClickListener{
        prefs.wipe()
        onBackPressed()
    }

    // Recuperem les dades directament del SharedPreferences
    // fent servir el prefs.
    val userName = prefs.getName()
    val userCognom = prefs.getCognom()
    val userLink = prefs.getLink()

    // Actualitzem la UI amb les dades internes.
    binding.tvName.text = "Benvingut $userName"
    binding.tvCognom.text = "Cognoms: $userCognom"
    binding.tvLink.text = "Link: $userLink"

    if(prefs.getVip())
    {
        colorFons()
    }
}

// Canviem el color de fons si és formal reutilitzo la funcio del vip de l'exemple
fun colorFons()
{
    binding.container.setBackgroundColor(ContextCompat.getColor(this,R.color.teal_200))
}
}

package com.example.preferences

import android.content.Context

class Prefs(val context: Context) {
    //Necessitem el context de l'aplicació però el context només el tenen les classes propies
```

```
d'android
//Creem una constant amb el nom de la base de dades
val SHARED_NAME ="Mydtb"
val SHARED_USER_NAME ="nom"
val SHARED_USER_COGNOM ="Cognom"
val SHARED_USER_LINK ="Link"
val SHARED_VIP="Vip"

//Instrucció per recuperar les preferències
val storage = context.getSharedPreferences(SHARED_NAME,0)

//Funcions per guardar
fun saveCognom(name:String){
    //Guardem el cognom
    storage.edit().putString(SHARED_USER_COGNOM,name).apply()
}

fun saveLink(name:String){
    //Guardem el link
    storage.edit().putString(SHARED_USER_LINK,name).apply()
}

fun saveName(name:String){
    //Guardem el nom
    storage.edit().putString(SHARED_USER_NAME,name).apply()
}

//Funció per guardar el vip (en el nostre cas formal)
fun saveVIP(vip:Boolean){
    //Guardem a la nostra SharedPreference
    storage.edit().putBoolean(SHARED_VIP,vip).apply()
}

//Funcions per recuperar dades

fun getName():String{
    //Posem el !! perquè sinó detecta que podria ser null. També podríem canviar el retorn a String?
    return storage.getString(SHARED_USER_NAME,"")!!
}

fun getCognom():String{
    return storage.getString(SHARED_USER_COGNOM,"")!!
}

fun getLink():String{
    return storage.getString(SHARED_USER_LINK,"")!!
}

//Funció per recuperar el vip (en el nostre cas modo formal)
```

```
fun getVip():Boolean{
    return storage.getBoolean(SHARED_VIP,false)
}

//Netejar les dades guardades
fun wipe(){
    storage.edit().clear().apply()
}

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity3"
    android:id="@+id/container"
    android:orientation="vertical"
    android:padding="24dp"
    android:background="#F7F7F7">

    <!-- Titol formal -->
    <TextView
        android:id="@+id/tvTitolFormal"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Dades de l'Usuari"
        android:textSize="26sp"
        android:textStyle="bold"
        android:textColor="#2B2B2B"
        android:gravity="center"
        android:paddingBottom="12dp" />

    <!-- linia -->
    <View
        android:layout_width="match_parent"
        android:layout_height="1dp"
        android:background="#CCCCCC"
        android:layout_marginBottom="20dp"/>

    <!-- part de dades-->
    <TextView
        android:id="@+id/tvName"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
```

```
    android:textSize="20sp"
    android:textColor="#333333"
    android:textStyle="bold"
    tools:text="Nom: Cristian" />

<TextView
    android:id="@+id/tvCognom"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textSize="18sp"
    android:layout_marginTop="6dp"
    android:textColor="#555555"
    tools:text="Cognoms: Jiménez" />

<TextView
    android:id="@+id/tvLink"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textSize="18sp"
    android:layout_marginTop="6dp"
    android:textColor="#555555"
    tools:text="GitHub: github.com/cris" />

<!-- Segona part -->
<View
    android:layout_width="match_parent"
    android:layout_height="1dp"
    android:background="#DDDDDD"
    android:layout_marginTop="24dp"
    android:layout_marginBottom="20dp"/>

<!-- Botó formal -->
<Button
    android:id="@+id	btnBack"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Tancar Sessió"
    android:textAllCaps="false"
    android:padding="12dp"
    android:textSize="18sp"
    android:textColor="#FFFFFF"
    android:background="#304FFE"
    android:layout_marginTop="16dp"
    android:layout_marginBottom="12dp" />

</LinearLayout>
```