5COSC023W - MOBILE APPLICATION DEVELOPMENT
Lecture 5: Android Shared Preferences

Dr Dimitris C. Dracopoulos

Saving Data in an Android Application

- Use `onSaveInstanceState()` for configuration changes or system destroying and re-creating the activity.
- Saving Key-Value Sets (small amounts)
- Saving in Files
- Saving in SQL databases (large amounts of structured data)

SharedPreferences (Saving Key-Value Sets)

To create a new shared preference file or access an existing one, call one of the following methods to get a SharedPreferences object:

- getSharedPreferences(): if you need multiple shared preferences files (the name of the preference file is the first argument) - can be called from any Context in the app

  ```
  sharedPref: SharedPreferences =
          getSharedPreferences("preference_filename",
                                          Context.MODE_PRIVATE);
  ```

- getPreferences(): call from an activity to use only one shared preference file associated with the activity

  ```
  sharedPref = getPreferences(Context.MODE_PRIVATE);
  ```

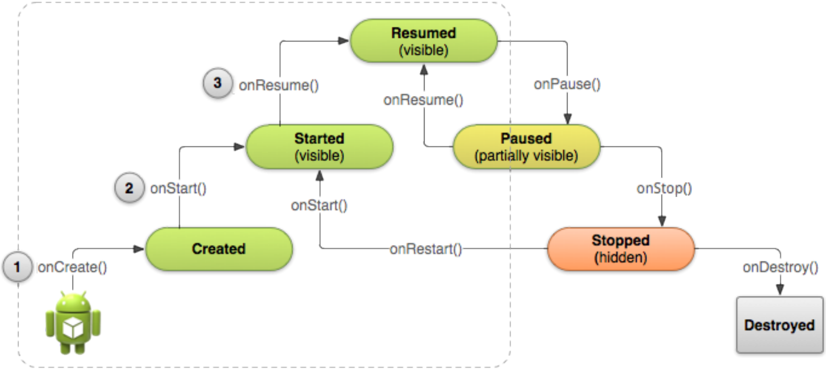
  Usage of MODE_WORLD_READABLE or MODE_WORLD_WRITEABLE imply that any other app can access your data (if it knows the filename)

Saving Key-Value Sets (Writing to Shared Preferences)

1. Create a `SharedPreferences.Editor` by calling `edit()` on `SharedPreferences`.
2. Write the keys and values with `putInt()`, `putString()`, etc.
3. Call `apply()` or `commit()`.

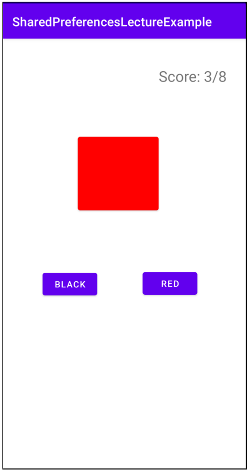
```
sharedPref: SharedPreferences  = getActivity().getPreferences(
                                               Context.MODE_PRIVATE);
editor: SharedPreferences.Editor = sharedPref.edit(); // step 1
editor.putInt("key_name", newHighScore); // step 2
editor.apply();  // step 3
```

The Activity Lifecycle (cont'ed)



An Example Application for SharedPreferences

An application which the user can guess the displayed colour. The score is persisted even the application is killed and restarted (even if the device reboots).



An Example Application for SharedPreferences (cont'ed)

The layout file `activity_main.xml`:

```xml
<?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:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <Button
        android:id="@+id/button"
        android:layout_width="130dp"
        android:layout_height="130dp"
        android:layout_marginStart="118dp"
        android:layout_marginTop="152dp"
        android:layout_marginEnd="141dp"
        android:text=""
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="1.0"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
```

An Example Application for SharedPreferences (cont'ed)

```
<Button
    android:id="@+id/button2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="60dp"
    android:layout_marginBottom="276dp"
    android:text=""
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toStartOf="@+id/button3"
    app:layout_constraintHorizontal_bias="0.059"
    app:layout_constraintStart_toStartOf="parent" />

<Button
    android:id="@+id/button3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginEnd="79dp"
    android:layout_marginBottom="277dp"
    android:text=""
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent" />
```

An Example Application for SharedPreferences (cont'ed)

```
    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="44dp"
        android:text="Score: 0/0"
        android:textSize="24sp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.889"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

An Example Application for SharedPreferences (cont'ed)

The activity code `MainActivity.kt`:

```
package uk.ac.westminster.sharedpreferenceslectureexample

import android.content.SharedPreferences
import android.graphics.Color
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.widget.Button
import android.widget.TextView
import java.util.*

class MainActivity : AppCompatActivity() {
    lateinit var prefs: SharedPreferences

    var colours = listOf(Color.BLACK, Color.BLUE, Color.RED, Color.GREEN, Color.WHITE, Color.YELLOW)
    var colours_str = listOf("BLACK", "BLUE", "RED", "GREEN", "WHITE", "YELLOW")
    var generator = Random()

    lateinit var bt: Button
    lateinit var bt2: Button
    lateinit var bt3: Button
    lateinit var tv: TextView

    var correct = 0  // number of correct answers
    var total = 0   // number of colours presented to the user
    var r = 0

    // this corresponds to either the left (if 0) or the right button (if 1)
    var correct_button = 0
```

An Example Application for SharedPreferences (cont'ed)

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

    // inflate the 3 buttons and the textview  from XML
    bt = findViewById(R.id.button)
    bt2 = findViewById(R.id.button2)
    bt3 = findViewById(R.id.button3)

    tv = findViewById<TextView>(R.id.textView)

    // create the shared preferences object
    prefs = getSharedPreferences("uk.ac.westminster.sharedpreferenceslectureexample", MODE_PRIVATE)
    // restore the data
    total = prefs.getInt("total", 0)
    correct = prefs.getInt("correct", correct)

    nextGame()

    // associate the 2 buttons with listeners
    bt2.setOnClickListener{
        ++total
        if (correct_button == 0)   // left button contains the correct answerr
            ++correct

        tv.setText("Score: $correct/$total" )

        // present the user with a new colour
        nextGame()
    }
```

An Example Application for SharedPreferences (cont'ed)

```
    bt3.setOnClickListener {
        ++total
        if (correct_button == 1)  // right buttons contains the correct answer
            ++correct

        tv.setText("Score: $correct/$total")

        // present the user with a new colour
        nextGame()
    }
}

/* what happens when the activity goes away
*  save tha data that I am interested in restoring later on*/
override fun onPause() {
    super.onPause()

    /* give me the editor associated with the sharedpreferences
       object created in the onCreate() method */
    var editor = prefs.edit()
    // start saving the data - in this case I just save the score
    editor.putInt("total", total)
    editor.putInt("correct", correct)

    // persist the data
    editor.apply()
}
```

An Example Application for SharedPreferences (cont'ed)

```
    fun nextGame() {
        tv.setText("Score: $correct/$total" )

        // generate a random number in the range 0->5
        r = generator.nextInt(colours.size)

        // choose the corresponding colour from the colours list - i.e. the r-th element in the list
        var random_colour_chosen = colours[r]

        // set the colour of the top button to the random colour chosen
        bt.setBackgroundColor(random_colour_chosen)

        // choose which button (left or right) contains the correct answer
        correct_button = generator.nextInt(2)

        // choose a second random colour to be displayed as well
        var second_colour = colours_str.random()
        while (second_colour == colours_str[r])
            second_colour = colours_str.random()

        // if the correct answer corresponds to the left button
        if (correct_button == 0 ) {
            bt2.setText("" + colours_str[r])
            bt3.setText("" + second_colour)
        }
        else {   // the right button contains the correct answer
            bt3.setText("" + colours_str[r])
            bt2.setText("" + second_colour)
        }
    }
}
```