

Taller de Android

SharedPreferences
Settings

SharedPreferences

Es una API de Android, la cual nos permite persistir valores simples del tipo clave – valor.

La gestión se centraliza en la clase SharedPreferences, que representará a una colección de preferencias.

Una aplicación Android puede gestionar varias colecciones de preferencias, que se diferencian mediante un identificador único. Para obtener una referencia a una colección determinada utilizaremos el método `getSharedPreferences()` al que pasaremos el identificador de la colección y un *modo de acceso*.

SharedPreferences

El valor guardado en SharedPreferences para cada configuración puede ser uno de los siguientes tipos de datos: Boolean, Float, Int, Long, String.

-Para obtener la colección preferencias

```
SharedPreferences preferencias= getSharedPreferences("perfil",  
MODE_PRIVATE);
```

-Para recuperar un valor de la colección utilizamos:
`getString(“key”, “default_value”)`

SharedPreferences

Para insertar o modificar valores necesitamos de la clase utilitaria Editor.

```
SharedPreferences.Editor editorPreferencias=preferencias.edit();
```

Los métodos los cuales nos permiten persistir valores son :

```
putLong(String key, long value)
```

```
putInt(String key, int value)
```

```
putString(String key, String value)
```

```
putStringSet(String key, Set<String> values)
```

```
commit()
```

SharedPreferences

//Obtener preferencias

```
SharedPreferences preferencias= getSharedPreferences("perfil",  
MODE_PRIVATE);  
String texto=preferencias.getString("texto", "");
```

//Guardar en preferencias

```
SharedPreferences preferencias=getSharedPreferences("perfil",  
MODE_PRIVATE);  
SharedPreferences.Editor editorPreferencias=preferencias.edit();  
editorPreferencias.putString("texto", nombre);  
editorPreferencias.commit();
```

Settings

Las aplicaciones generalmente incluyen una configuración que permite que los usuarios modifiquen las funciones y los comportamientos de las apps.

Para esto Android nos provee un conjunto de elementos para facilitar esta tarea.

Settings - PreferenceScreen

Paso 1: Crear el xml de preferencias, en la carpeta XML.

Nuestro XML de preferencias debe tener el tag:

```
<PreferenceScreen  
xmlns:android="http://schemas.android.com/apk/res/android" >
```

Settings - PreferenceScreen

CheckBoxPreference

Representa un tipo de opción que sólo puede tomar dos valores distintos: activada o desactivada. Es el equivalente a un control de tipo *checkbox*. En este caso tan sólo tendremos que especificar los atributos: nombre interno de la opción (`android:key`), texto a mostrar (`android:title`) y descripción de la opción (`android:summary`).

```
<CheckBoxPreference android:title="Guardar partida"
```

```
    android:defaultValue="false"
```

```
    android:summary="Desea que la partida se guarde."
```

```
    android:key="guardarPartida" />
```


Settings - PreferenceScreen

EditTextPreference

Representa un tipo de opción que puede contener como valor una cadena de texto. Al pulsar sobre una opción de este tipo se mostrará un cuadro de diálogo sencillo que solicitará al usuario el texto a almacenar. Para este tipo, además de los tres atributos comunes a todas las opciones (`key`, `title` y `summary`) también tendremos que indicar el texto a mostrar en el cuadro de diálogo, mediante el atributo `android:dialogTitle`.

```
<EditTextPreference android:title="Nombre del usuario"

    android:key="usuario"

    android:summary="Nombre del usuario"/>
```

Settings - PreferenceScreen

ListPreference

Representa un tipo de opción que puede tomar como valor un elemento, y sólo uno, seleccionado por el usuario entre una lista de valores predefinida. Al pulsar sobre una opción de este tipo se mostrará la lista de valores posibles y el usuario podrá seleccionar uno de ellos.

Estas listas de valores la cual usaremos para popular la lista la definiremos en como un archivo XML dentro de la carpeta `/values/arrays`. Definiremos para ello los recursos de tipos `<string-array>` necesarios, en este caso dos, uno para la lista de valores visibles y otro para la lista de valores internos, cada uno de ellos con su ID único correspondiente

Settings - PreferenceScreen

```
<ListPreference
    android:title="Dificultad"

    android:summary="Dificultad
"

    android:key="lista"

    android:defaultValue="1"

    android:entries="@array/listArray"

    android:entryValues="@array/
listValues" />
```

```
<resources>

<string-array
    name="listArray">
    <item>Baja</item>
    <item>Media</item>
    <item>Alta</item>
</string-array>

<string-array
    name="listValues">
    <item>1</item>
    <item>2</item>
    <item>3</item>
</string-array>
</resources>
```

Settings - PreferenceActivity

Paso 2

a) Crear nuestra Activity extendiendo de PreferenceActivity.

```
public class MyPreferencesActivity extends PreferenceActivity {  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
  
        getFragmentManager().beginTransaction().replace(android.R.id.conte  
nt, new MyPreferenceFragment()).commit();  
    }  
}
```

Settings - PreferenceActivity

b) Crear el fragment extendiendo de PreferenceFragment

```
public class MyPreferenceFragment extends PreferenceFragment
{
    @Override
    public void onCreate(final Bundle savedInstanceState)
    {
        super.onCreate(savedInstanceState);
        addPreferencesFromResource(R.xml.preferences);
    }
}
```

Settings - PreferenceActivity

Paso 3: Crear el punto de acceso en el menú, en las activitys que se requiera.

@Override

```
public boolean onCreateOptionsMenu(Menu menu) {
```

```
    MenuInflater inflater = getMenuInflater();
```

```
    inflater.inflate(R.menu.preference_main, menu);
```

```
    return true;
```

```
}
```

Settings - PreferenceActivity

Agregar action a la opción de menú.

@Override

```
public boolean onOptionsItemSelected(MenuItem item) {  
  
    int id = item.getItemId();  
    if (id == R.id.action_settings) {  
        Intent i = new Intent(this, MyPreferencesActivity.class);  
        startActivity(i);  
        return true;  
    }  
    return super.onOptionsItemSelected(item);  
}
```

Settings - PreferenceActivity

```
#menu/preference_main.xml
```

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

```
<menu
```

```
xmlns:android="http://schemas.android.com/apk/res/android">
```

```
    <item android:id="@+id/action_settings"
```

```
        android:showAsAction="never"
```

```
        android:title="@string/action_settings" />
```

```
</menu>
```


Settings - PreferenceActivity

Por defecto el sistema guarda todos los cambios de los ajustes en un archivo SharedPreferences y para acceder a ellos tenemos que utilizar una instancia "PreferenceManager" de la siguiente manera:

```
SharedPreferences sp =  
PreferenceManager.getDefaultSharedPreferences(this);  
  
sp.getString("usuario", "");
```