**FORMATO DE TAREAS**

|  |  |
| --- | --- |
| Nombres y Apellidos del Estudiante: | Michelle Zhanay |
| Nombre de la Asignatura: | Desarrollo de Aplicación Web |
| Periodo Académico: | II Periodo Académico 2020 |
| Carrera Tecnológica: | Tecnología Superior En Desarrollo De Software |
| Ciclo de estudio: | 4° ciclo |
| Fecha de entrega: | 07/09/20 |

**Tema: Realice la siguiente consulta.**

**Consultar sobre sharedPreferences en Android**

**SHAREDPREFERENCES:**

“En Android hay varias maneras de almacenar datos: Puedes hacerlo a través de una base de datos local y/o remota, a través de las Shared Preferences o combinar ambas formas.

Normalmente se usan para almacenar información personal, opciones de presentación, etc. La limitación la ponemos nosotros, ya que está en la imaginación del propio diseñador de la aplicación.

Lo interesante de todo esto es que ese valor seguirá siendo recordado a pesar de cerrar nuestra aplicación o de incluso reiniciar o apagar nuestro móvil.

La gestión de estas preferencias se centraliza en la clase SharedPreferences, que representa una colección de preferencias. A la hora de instanciarlas, estas tienen un identificador, que es la clave o nombre que le hayamos puesto nosotros, y un modo de acceso:

MODE\_PRIVATE: Nuestra aplicación tiene acceso a estas preferencias.

MODE\_WORLD\_READABLE: Todas las aplicaciones tienen acceso en modo lectura a estas preferencias, pero únicamente nuestra aplicación puede modificarlas.

MODE\_WORLD\_WRITABLE: Todas las aplicaciones pueden leer y escribir sobre estas preferencias.

Para hacer una llamada a nuestras preferencias, debes incluir el siguiente código:

SharedPreferences prefs = getSharedPreferences("MyPreferences", Context.MODE\_PRIVATE);

Una vez referenciada nuestra colección de preferencias, ahora podemos obtener sus valores mediante el método get y escribir mediante el método put.

Por ejemplo, para obtener el valor de una preferencia llamada “url” de tipo String deberíamos poner lo siguiente:

String url = prefs.getString("url", "www.google.com");

Con esto le estamos diciendo que obtenga en valor de la preferencia “url” o, si estuviese vacío, obtuviese el valor por defecto, que en este caso es “www.google.com”.

También podríamos haber hecho lo siguiente:

String url = prefs.getString("url", "");

En el cual el valor por defecto estaría vacío.

Ten en cuenta que también admiten otros métodos para el resto de tipos de datos básicos, como son:

getBoolean()

getInt()

getFloat()

getLong()

Escribiendo valores en las preferencias

Para insertar un valor o actualizarlo, hay que hacerlo utilizando el objeto SharedPreferences.Editor.

Una vez referenciado el editor, se ha de realizar un put sobre el tipo de dato básico correspondiente y validamos mediante el método commit(). Siguiendo el ejemplo de la URL:

SharedPreferences prefs = getSharedPreferences("MyPreferences", Context.MODE\_PRIVATE);

SharedPreferences.Editor editor = prefs.edit();

editor.putString("url", "www.saulcintero.com");

editor.commit();

Y de esta manera hemos creado el valor, si no existía, o actualizado su contenido a “www.saulcintero.com” dentro de la preferencia “url”.

Ahora imagina que en vez de tipos de valores individuales, necesitas crear un array de objetos en tus Shared Preference. ¿Qué harías?

Bueno, en respuesta a eso encontré en Stack Overflow un mensaje que explicaba cómo hacerlo sin dependencia de una API en una versión en concreto. Aquí te dejo los dos métodos que hacen falta para realizar esto:

[Salvar el array]

public boolean saveArray(String[] array, String arrayName, Context mContext) {

SharedPreferences prefs = mContext.getSharedPreferences("preferencename", 0);

SharedPreferences.Editor editor = prefs.edit();

editor.putInt(arrayName +"\_size", array.length);

for(int i=0;i<array.length;i++)

editor.putString(arrayName + "\_" + i, array[i]);

return editor.commit();

}

[Recuperar el array]

public String[] loadArray(String arrayName, Context mContext)

{

SharedPreferences prefs = mContext.getSharedPreferences("preferencename", 0);

int size = prefs.getInt(arrayName + "\_size", 0);

String array[] = new String[size];

for(int i=0;i<size;i++)

array[i] = prefs.getString(arrayName + "\_" + i, null);

return array;

}

Puedes comprobarlo dentro del siguiente path:

/data/data/nombre.del.paquete/shared\_prefs/nombre.de.la.colección.xml”

(Cintero, 2013)

# Bibliografía

Cintero, S. (27 de Agosto de 2013). *Saúl Cintero*. Obtenido de Saúl Cintero: http://www.saulcintero.com/preferencias-en-android-shared-preferences/