**Saving Key-Value Sets**

[PREVIOUS](file:///E:\android\adt-bundle-windows-x86-20131030\sdk\docs\training\basics\data-storage\index.html)[NEXT](file:///E:\android\adt-bundle-windows-x86-20131030\sdk\docs\training\basics\data-storage\files.html)

## THIS LESSON TEACHES YOU TO

1. [Get a Handle to a SharedPreferences](file:///E:\android\adt-bundle-windows-x86-20131030\sdk\docs\training\basics\data-storage\shared-preferences.html#GetSharedPreferences)
2. [Write to Shared Preferences](file:///E:\android\adt-bundle-windows-x86-20131030\sdk\docs\training\basics\data-storage\shared-preferences.html#WriteSharedPreference)
3. [Read from Shared Preferences](file:///E:\android\adt-bundle-windows-x86-20131030\sdk\docs\training\basics\data-storage\shared-preferences.html#ReadSharedPreference)

## YOU SHOULD ALSO READ

* [Using Shared Preferences](file:///E:\android\adt-bundle-windows-x86-20131030\sdk\docs\guide\topics\data\data-storage.html#pref)

每一个SharedPreference对象都指向一个包含一系列键值对的文件，该对象也提供了用于读写这些键值对的方法。每一个SharedPreference文件对象被框架管理，可以是私有的也可是是共享的。

If you have a relatively small collection of key-values that you'd like to save, you should use the[SharedPreferences](file:///E:\android\adt-bundle-windows-x86-20131030\sdk\docs\reference\android\content\SharedPreferences.html) APIs. A [SharedPreferences](file:///E:\android\adt-bundle-windows-x86-20131030\sdk\docs\reference\android\content\SharedPreferences.html) object points to a file containing key-value pairs and provides simple methods to read and write them. Each [SharedPreferences](file:///E:\android\adt-bundle-windows-x86-20131030\sdk\docs\reference\android\content\SharedPreferences.html)file is managed by the framework and can be private or shared.

**Note:** The [SharedPreferences](file:///E:\android\adt-bundle-windows-x86-20131030\sdk\docs\reference\android\content\SharedPreferences.html) APIs are only for reading and writing key-value pairs and you should not confuse them with the [Preference](file:///E:\android\adt-bundle-windows-x86-20131030\sdk\docs\reference\android\preference\Preference.html) APIs, which help you build a user interface for your app settings (although they use [SharedPreferences](file:///E:\android\adt-bundle-windows-x86-20131030\sdk\docs\reference\android\content\SharedPreferences.html) as their implementation to save the app settings). For information about using the [Preference](file:///E:\android\adt-bundle-windows-x86-20131030\sdk\docs\reference\android\preference\Preference.html) APIs, see the [Settings](file:///E:\android\adt-bundle-windows-x86-20131030\sdk\docs\guide\topics\ui\settings.html) guide.

## Get a Handle to a SharedPreferences

You can create a new shared preference file or access an existing one by calling one of two methods

* [getSharedPreferences()](file:///E:\android\adt-bundle-windows-x86-20131030\sdk\docs\reference\android\content\Context.html#getSharedPreferences(java.lang.String, int)) — Use this if you need multiple shared preference files identified by name, which you specify with the first parameter. You can call this from any [Context](file:///E:\android\adt-bundle-windows-x86-20131030\sdk\docs\reference\android\content\Context.html) in your app.
* [getPreferences()](file:///E:\android\adt-bundle-windows-x86-20131030\sdk\docs\reference\android\app\Activity.html#getPreferences(int)) — Use this from an [Activity](file:///E:\android\adt-bundle-windows-x86-20131030\sdk\docs\reference\android\app\Activity.html) if you need to use only one shared preference file for the activity. Because this retrieves a default shared preference file that belongs to the activity, you don't need to supply a name.

For example, the following code is executed inside a [Fragment](file:///E:\android\adt-bundle-windows-x86-20131030\sdk\docs\reference\android\app\Fragment.html). It accesses the shared preferences file that's identified by the resource string R.string.preference\_file\_key and opens it using the private mode so the file is accessible by only your app.

Context context = getActivity();  
SharedPreferences sharedPref = context.getSharedPreferences(  
        getString(R.string.preference\_file\_key), Context.MODE\_PRIVATE);

When naming your shared preference files, you should use a name that's uniquely identifiable to your app, such as "com.example.myapp.PREFERENCE\_FILE\_KEY"

Alternatively, if you need just one shared preference file for your activity, you can use the[getPreferences()](file:///E:\android\adt-bundle-windows-x86-20131030\sdk\docs\reference\android\app\Activity.html#getPreferences(int)) method:

SharedPreferences sharedPref = getActivity().getPreferences(Context.MODE\_PRIVATE);

**Caution:** If you create a shared preferences file with [MODE\_WORLD\_READABLE](file:///E:\android\adt-bundle-windows-x86-20131030\sdk\docs\reference\android\content\Context.html#MODE_WORLD_READABLE) or [MODE\_WORLD\_WRITEABLE](file:///E:\android\adt-bundle-windows-x86-20131030\sdk\docs\reference\android\content\Context.html#MODE_WORLD_WRITEABLE), then any other apps that know the file identifier can access your data.

## Write to Shared Preferences

To write to a shared preferences file, create a [SharedPreferences.Editor](file:///E:\android\adt-bundle-windows-x86-20131030\sdk\docs\reference\android\content\SharedPreferences.Editor.html) by calling [edit()](file:///E:\android\adt-bundle-windows-x86-20131030\sdk\docs\reference\android\content\SharedPreferences.html#edit()) on your[SharedPreferences](file:///E:\android\adt-bundle-windows-x86-20131030\sdk\docs\reference\android\content\SharedPreferences.html).

Pass the keys and values you want to write with methods such as [putInt()](file:///E:\android\adt-bundle-windows-x86-20131030\sdk\docs\reference\android\content\SharedPreferences.Editor.html#putInt(java.lang.String, int)) and [putString()](file:///E:\android\adt-bundle-windows-x86-20131030\sdk\docs\reference\android\content\SharedPreferences.Editor.html#putString(java.lang.String, java.lang.String)). Then call[commit()](file:///E:\android\adt-bundle-windows-x86-20131030\sdk\docs\reference\android\content\SharedPreferences.Editor.html#commit()) to save the changes. For example:

SharedPreferences sharedPref = getActivity().getPreferences(Context.MODE\_PRIVATE);  
SharedPreferences.Editor editor = sharedPref.edit();  
editor.putInt(getString(R.string.saved\_high\_score), newHighScore);  
editor.commit();

## Read from Shared Preferences

To retrieve values from a shared preferences file, call methods such as [getInt()](file:///E:\android\adt-bundle-windows-x86-20131030\sdk\docs\reference\android\content\SharedPreferences.html#getInt(java.lang.String, int)) and [getString()](file:///E:\android\adt-bundle-windows-x86-20131030\sdk\docs\reference\android\content\SharedPreferences.html#getString(java.lang.String, java.lang.String)), providing the key for the value you want, and optionally a default value to return if the key isn't present. For example:

SharedPreferences sharedPref = getActivity().getPreferences(Context.MODE\_PRIVATE);  
int defaultValue = getResources().getInteger(R.string.saved\_high\_score\_default);  
long highScore = sharedPref.getInt(getString(R.string.saved\_high\_score), defaultValue);