Использование SharedPreferences для сохранения настроек

Для большинства приложений требуется сохранить настройки, управляющие поведением программы. Android предоставляет несколько альтернативных способов сохранения данных приложения, каждый из которыхоптимизи­рован для определенных нужд.

**Общие настройки** — простой и легкий механизм, основанный на парах «ключ — значение» и предназначенный для сохранения примитивных данных приложения, чаще всего пользовательских настроек.

* [Введение](http://suvitruf.ru/2012/09/17/1916/android-ispolzovanie-sharedpreferences-dlya-soxraneniya-nastroek/#intro)

[Создание и сохранение настроек](http://suvitruf.ru/2012/09/17/1916/android-ispolzovanie-sharedpreferences-dlya-soxraneniya-nastroek/#save_and_change_pref)

[Получение общих настроек](http://suvitruf.ru/2012/09/17/1916/android-ispolzovanie-sharedpreferences-dlya-soxraneniya-nastroek/#get_pref)

[Очистка/удаление значения](http://suvitruf.ru/2012/09/17/1916/android-ispolzovanie-sharedpreferences-dlya-soxraneniya-nastroek/#delete_pref)

[Preferences Framework](http://suvitruf.ru/2012/09/17/1916/android-ispolzovanie-sharedpreferences-dlya-soxraneniya-nastroek/#preferences_framework)

[PreferenceScreen](http://suvitruf.ru/2012/09/17/1916/android-ispolzovanie-sharedpreferences-dlya-soxraneniya-nastroek/#preferences_screen)

[settings.xml](http://suvitruf.ru/2012/09/17/1916/android-ispolzovanie-sharedpreferences-dlya-soxraneniya-nastroek/#settings_xml)

[Определение параметров в string.xml и array.xml](http://suvitruf.ru/2012/09/17/1916/android-ispolzovanie-sharedpreferences-dlya-soxraneniya-nastroek/#string_xml_array_xml)

[PreferenceActivity](http://suvitruf.ru/2012/09/17/1916/android-ispolzovanie-sharedpreferences-dlya-soxraneniya-nastroek/#preference_activity)

[Отслеживание изменений в общих настройках](http://suvitruf.ru/2012/09/17/1916/android-ispolzovanie-sharedpreferences-dlya-soxraneniya-nastroek/#onSharedPreferenceChangeListener)

[This function is not relevant for a modern fragment-based PreferenceActivity](http://suvitruf.ru/2012/09/17/1916/android-ispolzovanie-sharedpreferences-dlya-soxraneniya-nastroek/#This_function_is_not_relevant_for_a_modern_fragmentbased_PreferenceActivity)

Введение

Класс **SharedPreferences** позволяет создавать в приложении именованные ассоциативные массивы типа «ключ — значение», которые могут быть использованы различными компонентами приложения (работая при этом  
в контексте одного и того же приложения).

Общие настройки поддерживают базовые типы boolean, string, float, long и integer, что делает их идеальным средством для быстрого сохранения значений по умолчанию, переменных экземпляра класса, текущего состояния UI и пользовательских настроек.

Создание и сохранение настроек

Чтобы создать или изменить Общие настройки, нужно вызвать метод  
getSharedPreferences в контексте приложения, передавая имя общих настроек, которые вы хотите изменить. Настройки недоступны для других приложений.

[view source](http://suvitruf.ru/2012/09/17/1916/android-ispolzovanie-sharedpreferences-dlya-soxraneniya-nastroek/#viewSource)[print](http://suvitruf.ru/2012/09/17/1916/android-ispolzovanie-sharedpreferences-dlya-soxraneniya-nastroek/#printSource)

public static String MY\_PREF = "MY\_PREF";

protected void savePreferences()

{

// получить доступ к объекту Editor, чтобы изменить общие настройки.

SharedPreferences.Editor editor = mySharedPreferences.edit();

// задать новые базовые типы в объекте общих настроек.

editor.putBoolean("isTrue", true);

editor.putFloat("floatNumber", 1f);

editor.putInt("intNumber", 2);

editor.putLong("longNumber", 3l);

editor.putString("textValue", "Not Empty");

editor.commit();

}

Получение общих настроек

Надо передать в getSharedPreferences имя тех общих настроек, доступ к которым хотите получить, и используйте типизированный метод get<тип>, чтобы извлечь сохраненные значения. Передаются ключ и значение по умолчанию (используется случае, если для данного ключа пока что не сохранено никакое значение).

[view source](http://suvitruf.ru/2012/09/17/1916/android-ispolzovanie-sharedpreferences-dlya-soxraneniya-nastroek/#viewSource)[print](http://suvitruf.ru/2012/09/17/1916/android-ispolzovanie-sharedpreferences-dlya-soxraneniya-nastroek/#printSource)

public static String MY\_PREF = "MY\_PREF";

public void loadPreferences()

{

int mode = Activity.MODE\_PRIVATE;

SharedPreferences mySharedPreferences = getSharedPreferences(MY\_PREF, mode);

boolean isTrue = mySharedPreferences.getBoolean("isTrue", false);

float lastFloat = mySharedPreferences.getFloat("floatNumber", 0f);

int wholeNumber = mySharedPreferences.getInt("intNumber", 1);

long aNumber = mySharedPreferences.getLong("longNumber", 0);

String stringPreference = mySharedPreferences.getString("textValue", "");

}

Очистка/удаление значения

Для очистки значений используйте методы

SharedPreferences.Editor.remove(String key)

и

SharedPreferences.Editor.clear().

Файл настроек хранится в

**/data/data/package\_name/shared\_prefs/имя\_файла\_настроек.xml**. Можно удалить:

File file= new File("/data/data/.../shared\_prefs/файл\_настроек.xml")

file.delete();

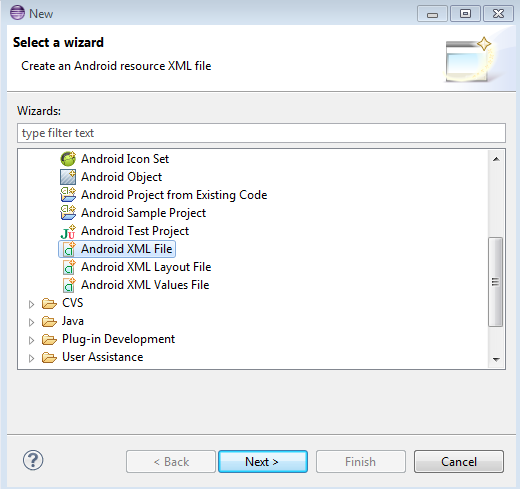
Preferences Framework

В примерах выше происходило обычное сохранение и восстанавление нужных параметров. То есть, создаётся свой xml-файл разметки и размещаются там нужные элементы управления. Но Android для этих целей предоставляет собственный Preferences Framework, с помощью которого можно создавать индивидуальный набор предпочтений и встраивать их в приложения.

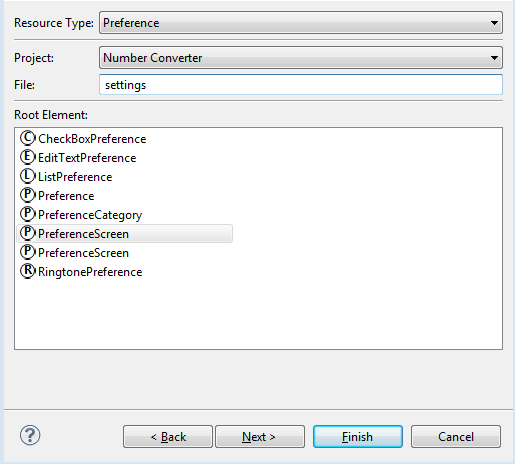
То есть, теперь нет необходимости создавать файл разметки и get/set параметров писать самим. Необходимо лишь в отдельном файле перечислить параметры, а Android уже остальное сделает. Предпочтения — это отдельная активность в приложении, вызываемая из активности. Сам Activity расширяется от класса PreferenceActivity. Предпочтения определяются в отдельном XML-файле, где корневым элементом является элемент <PreferenceScreen>, который представляет собой контейнер для предпочтений и может содержать дочерние элементы <PreferenceScreen>. Элемент <PreferenceCategory> также является контейнерным элементом и предназначен для объединения предпочтений в группы.

PreferenceScreen

Необходимо первоначально создать в папке **res/xml/** XML-файл ресурсов, как например, **settings.xml**. Выбирайте тип файла **Android XML File**:



При создании файла в диалоговом окне установите переключатель на **Preference** и убедитесь, что корневым элементом файла установлен **PreferenceScreen**:



settings.xml

Далее задаём необходимые настройки:

[view source](http://suvitruf.ru/2012/09/17/1916/android-ispolzovanie-sharedpreferences-dlya-soxraneniya-nastroek/#viewSource)[print](http://suvitruf.ru/2012/09/17/1916/android-ispolzovanie-sharedpreferences-dlya-soxraneniya-nastroek/#printSource)

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

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

<PreferenceCategory android:title="@string/interface\_settings">

<EditTextPreference

android:key="pref\_size"

android:title="@string/interface\_settings\_pref\_size"

android:summary="@string/interface\_settings\_set\_new\_fsize"

android:defaultValue="14"

android:dialogTitle="@string/interface\_settings\_enter\_fsize"/>

<ListPreference

android:key="pref\_lng"

android:title="@string/interface\_settings\_language"

android:summary="@string/interface\_settings\_set\_language"

android:defaultValue="1"

android:entries="@array/lng\_list"

android:entryValues="@array/entryvalues\_lng"

android:dialogTitle="@string/interface\_settings\_check\_language"/>

</PreferenceCategory>

</PreferenceScreen>

Мы определили две настройки: размер шрифта, язык приложения. С <EditTextPreference> всё в целом понятно, простое поле. А вот для выбора языка посложнее элемент. Нас тут интересует 2 поля: android:entries и android:entryValues. Из названия, думаю, понятно их назначение.

Определение параметров в string.xml и array.xml

Естественно, необходимо определить все переменные. В **string.xml**:

[view source](http://suvitruf.ru/2012/09/17/1916/android-ispolzovanie-sharedpreferences-dlya-soxraneniya-nastroek/#viewSource)[print](http://suvitruf.ru/2012/09/17/1916/android-ispolzovanie-sharedpreferences-dlya-soxraneniya-nastroek/#printSource)

<resources>

<string  name="pref\_lng">Стиль</string>

<string  name="pref\_size">Размер</string>

<string name="interface\_settings">Настройки интерфейса</string>

<string name="interface\_settings\_pref\_size">Размер шрифта</string>

<string name="interface\_settings\_set\_new\_fsize">Устанавливает новый размер шрифта</string>

<string name="interface\_settings\_enter\_fsize">Введите размер шрифта (от 10 до 32)</string>

<string name="interface\_settings\_language">Язык приложения</string>

<string name="interface\_settings\_set\_language">Устанавливает стиль для шрифта</string>

<string name="interface\_settings\_check\_language">Выберите язык</string>

<string name="title\_activity\_settings">Настройки</string>

</resources>

А так же в **array.xml**:

[view source](http://suvitruf.ru/2012/09/17/1916/android-ispolzovanie-sharedpreferences-dlya-soxraneniya-nastroek/#viewSource)[print](http://suvitruf.ru/2012/09/17/1916/android-ispolzovanie-sharedpreferences-dlya-soxraneniya-nastroek/#printSource)

<resources>

<string-array name="lng\_list">

<item>Язык аппарата</item>

<item>English</item>

<item>Русский</item>

</string-array>

<string-array name="entryvalues\_lng">

<item>default</item>

<item>en</item>

<item>ru</item>

</string-array>

</resources>

PreferenceActivity

Для PreferenceActivity автоматически создаётся SettingsActivity. Нам в нём слегка необходимо изменить метод onCreate.

[view source](http://suvitruf.ru/2012/09/17/1916/android-ispolzovanie-sharedpreferences-dlya-soxraneniya-nastroek/#viewSource)[print](http://suvitruf.ru/2012/09/17/1916/android-ispolzovanie-sharedpreferences-dlya-soxraneniya-nastroek/#printSource)

@Override

public void onCreate(Bundle savedInstanceState)

{

super.onCreate(savedInstanceState);

addPreferencesFromResource(R.xml.settings);

}

Метод addPreferencesFromResource() считывает установленные настройки из XML-файла, где хранятся наши ресурсы, и делает их доступными для программы. В результате мы должны увидеть настройки на экране. Если автоматически наше Activity не было зарегестрировано, то добавим в манифесте AndroidManifest.xml:

[view source](http://suvitruf.ru/2012/09/17/1916/android-ispolzovanie-sharedpreferences-dlya-soxraneniya-nastroek/#viewSource)[print](http://suvitruf.ru/2012/09/17/1916/android-ispolzovanie-sharedpreferences-dlya-soxraneniya-nastroek/#printSource)

<activity

android:name=".SettingsActivity"

android:label="@string/title\_activity\_settings" >

<meta-data

android:name="android.support.PARENT\_ACTIVITY"

android:value="android.preference.PreferenceActivity" />

</activity>

Ну и напишем код, чтобы при клике на меню, открывалось наше окно с настройками. Для этого определим метод по созданию меню и onCreateOptionsMenu и метод, отслеживающий нажатия по меню onOptionsItemSelected:

[view source](http://suvitruf.ru/2012/09/17/1916/android-ispolzovanie-sharedpreferences-dlya-soxraneniya-nastroek/#viewSource)[print](http://suvitruf.ru/2012/09/17/1916/android-ispolzovanie-sharedpreferences-dlya-soxraneniya-nastroek/#printSource)

public static final int IDM\_PREF = 101;

@Override

public boolean onCreateOptionsMenu(Menu menu )

{

menu.add(0,  IDM\_PREF, 0, "Настройки");

return super.onCreateOptionsMenu(menu);

}

@Override

public boolean onOptionsItemSelected(MenuItem item)

{

Intent intent = new Intent();

switch (item.getItemId())

{

case IDM\_PREF:

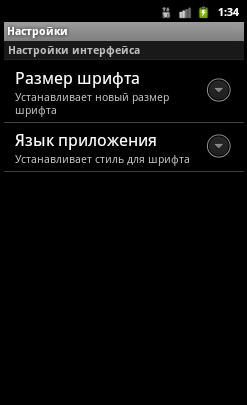
intent.setClass(this, SettingsActivity.class);

startActivity(intent);

break;

}

}



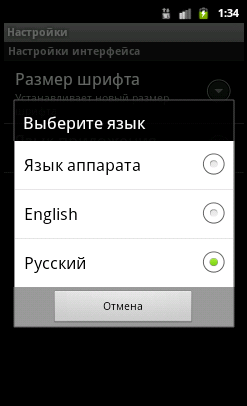
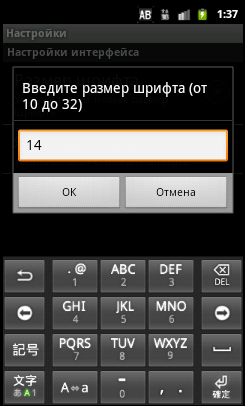
Теперь при вызове меню опций, нажимайте на пункт «Настройки» и откроется наше окно настроек.

Для сохранения предпочтений на выбор есть 4 класса:

* CheckBoxPreference
* EditTextPreference
* ListPreference
* RingtonePreference

Конкретно в данном примере используется **EditTextPreference** и **ListPreference**.

Так выглядит окно изменения для настроек:

Отслеживание изменений в общих настройках

Класс [onSharedPreferenceChangeListener](http://suvitruf.ru/android/" \l "useful_source_code_restart_OnSharedPreferenceChange" \o "Android: onSharedPreferenceChangeListener) позволяет вызвать callback-метод в момент добавления, удаления или изменения конкретной Общей настройки. Используя этот обработчик, компоненты вашего приложения могут следить за изменениями в настройках, установленных пользователем, и обновлять пользовательский интерфейс или корректировать поведение программы.

This function is not relevant for a modern fragment-based PreferenceActivity

This function is not relevant for a modern fragment-based PreferenceActivity. Inflates the given XML resource and adds the preference hierarchy to the current preference hierarchy.

Такое сообщение вы увидите в SDK по поводу многих методов PreferenceActivity, у того же addPreferencesFromResource. Это означает, что методы устарели. В сущности, ничего фатального в этом нет, просто в более поздних версиях появились более удобные методы.

С версии **3.0** появились так называемые [Fragments](http://developer.android.com/guide/components/fragments.html), с помощью которых можно работать с настройками.

Ещё раз повторю, если у вас старый вариант для работы с настройками используется, то он будет работать на всех устройствах нормально. Но если вы пишите для устройств с версией Android 3+, то лучше используйте фрагменты.