**Android 5 Preferences tutorial (PreferenceScreen, PreferenceActivity, and PreferenceFragment)**

By Alvin Alexander. Last updated: July 19 2018

***Table of Contents***

* [1) The res/xml/preferences.xml file](https://alvinalexander.com/android/android-tutorial-preferencescreen-preferenceactivity-preferencefragment#the-res-xml-preferences-xml-file)
* [2) The res/values/array.xml file](https://alvinalexander.com/android/android-tutorial-preferencescreen-preferenceactivity-preferencefragment#the-res-values-array-xml-file)
* [3) Your Android Preferences activity](https://alvinalexander.com/android/android-tutorial-preferencescreen-preferenceactivity-preferencefragment#your-android-preferences-activity)
* [4) Java code to launch the PreferenceActivity](https://alvinalexander.com/android/android-tutorial-preferencescreen-preferenceactivity-preferencefragment#java-code-to-launch-the-preferenceactivity)
* [5) Saving/reading data](https://alvinalexander.com/android/android-tutorial-preferencescreen-preferenceactivity-preferencefragment#saving-reading-data)
* [Images](https://alvinalexander.com/android/android-tutorial-preferencescreen-preferenceactivity-preferencefragment#images)
* [Note: Android tablets, phablets, and landscape mode](https://alvinalexander.com/android/android-tutorial-preferencescreen-preferenceactivity-preferencefragment#note-android-tablets-phablets-and-landscape-mode)
* [Full source code for my main launcher Activity](https://alvinalexander.com/android/android-tutorial-preferencescreen-preferenceactivity-preferencefragment#full-source-code-for-my-main-launcher-activity)
* [Summary: Preferences tutorial for Android 5](https://alvinalexander.com/android/android-tutorial-preferencescreen-preferenceactivity-preferencefragment#summary-preferences-tutorial-for-android-5)

Android FAQ: How do I create a *Preferences* UI in [Android](https://www.android.com/)?

Solution: Creating a Preferences screen in Android isn’t too hard. The website rominirani.com has a very good [preferences tutorial](http://rominirani.com/android-preferences-tutorial/) that just needs a minor update for Android 7, as a piece of the code shown there has been deprecated.

Шаги для создания пользовательского интерфейса настроек Android:

1. Определить пользовательский интерфейс в XML, используя PreferenceScreenв качестве основного компонента/контейнер
2. В качестве необязательного шага заполните некоторые массивы строк Android, чтобы использовать их с компонентами предпочтений
3. Написать PreferenceActivity и PreferenceFragment в Java-код, чтобы показать/обработки пользовательского интерфейса

Я добавляю еще несколько шагов в следующем коде, но это основные шаги. (Обратите внимание, что следующий код составляет около 95% кода iRomin, и 5% обновления для Android 7.)

**1) The res/xml/preferences.xml file**

Define your Preferences UI using a PreferenceScreen as the main container, and fill it with the sub-components needed for your preferences. Put this code in a file named *res/xml/preferences.xml* to get started:

Определите ваши Preferences UI с помощью PreferenceScreen как основной контейнер, и заполнить его с суб-компонентов, необходимых для вашего предпочтения. Поместите этот код в файл в *res/xml/preferences.xml*  -файле , чтобы начать:

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

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

<EditTextPreference android:title="Your Name"

android:key="username"

android:summary="Please provide your username"></EditTextPreference>

<CheckBoxPreference android:title="Application Updates"

android:defaultValue="false"

android:summary="This option if selected will allow the application to check for latest versions."

android:key="applicationUpdates" />

<ListPreference android:title="Download Details"

android:summary="Select the kind of data that you would like to download"

android:key="downloadType"

android:defaultValue="1"

android:entries="@array/listArray"

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

</PreferenceScreen>

**2) The res/values/array.xml file**

iRomani кода необходимо следующее строковый массив значений, которые он определяет в файле с именем *res/values/array.xml*:

<resources>

<string-array name="listArray">

<item>Headings</item>

<item>Headings and Details</item>

<item>All Data</item>

</string-array>

<string-array name="listValues">

<item>1</item>

<item>2</item>

<item>3</item>

</string-array>

</resources>

(См. мой [Андроид строковый массив учебнике](https://alvinalexander.com/android/how-to-create-static-array-strings-in-android-xml-file) , если вам нужно понять больше о строковых массивов, определенных в XML.)

**3) Your Android Preferences activity**

Код здесь был обновлен с iRomani уроке, и содержит необходимые исправления для работы с Android 5, в частности, использование PreferenceFragment наряду с PreferenceActivity::

package com.alvinalexander.preferencestestapp;

import android.os.Bundle;

import android.preference.PreferenceActivity;

import android.preference.PreferenceFragment;

public class MyPreferencesActivity extends PreferenceActivity {

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

getFragmentManager().beginTransaction().replace(android.R.id.content, new MyPreferenceFragment()).commit();

}

public static class MyPreferenceFragment extends PreferenceFragment

{

@Override

public void onCreate(final Bundle savedInstanceState)

{

super.onCreate(savedInstanceState);

addPreferencesFromResource(R.xml.preferences);

}

}

}

[Фрагменты](https://developer.android.com/guide/components/fragments.html?hl=ru) обеспечивают более универсальную архитектуру для вашего приложения по сравнению с использованием отдельных операций, вне зависимости от типа создаваемой операции. Фактически, для управления отображением ваших настроек мы предлагаем вам использовать [PreferenceFragment](https://developer.android.com/reference/android/preference/PreferenceFragment.html?hl=ru) вместо [PreferenceActivity](https://developer.android.com/reference/android/preference/PreferenceActivity.html?hl=ru) при каждой возможности.

Ваша реализация [PreferenceFragment](https://developer.android.com/reference/android/preference/PreferenceFragment.html?hl=ru) может содержать просто определение метода [onCreate()](https://developer.android.com/reference/android/preference/PreferenceFragment.html?hl=ru" \l "onCreate(android.os.Bundle)) для загрузки файла предпочтений посредством [addPreferencesFromResource()](https://developer.android.com/reference/android/preference/PreferenceFragment.html?hl=ru" \l "addPreferencesFromResource(int)). Например:

public static class SettingsFragment extends PreferenceFragment {  
    @Override  
    public void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
  
        // Load the preferences from an XML resource  
        addPreferencesFromResource(R.xml.preferences);  
    }  
    ...  
}

**4) Java code to launch the PreferenceActivity**

Вы запустите PreferenceActivity так же, как любой другой Android Activity, с помощью Intent. Этот код показывает, что:

Intent i = new Intent(this, MyPreferencesActivity.class);

startActivity(i);

(Полный исходный код для моего действия "launcher" показан внизу, после изображений.)

Или вы можете добавить этот фрагмент в операцию [Activity](https://developer.android.com/reference/android/app/Activity.html?hl=ru), как вы сделали бы это для любого другого фрагмента [Fragment](https://developer.android.com/reference/android/app/Fragment.html?hl=ru). Например:

public class SettingsActivity extends Activity {  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
  
        // Display the fragment as the main content.  
        getFragmentManager().beginTransaction()  
                .replace(android.R.id.content, new SettingsFragment())  
                .commit();  
    }  
}

**5) Saving/reading data**

Как iromani обсуждает, вы можете сохранить / читать данные предпочтений в других местах вашего приложения, как это::

SharedPreferences SP = PreferenceManager.getDefaultSharedPreferences(getBaseContext());

String strUserName = SP.getString("username", "NA");

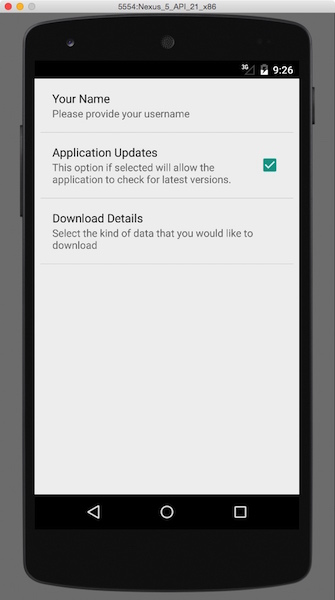
boolean bAppUpdates = SP.getBoolean("applicationUpdates",false);

String downloadType = SP.getString("downloadType","1");

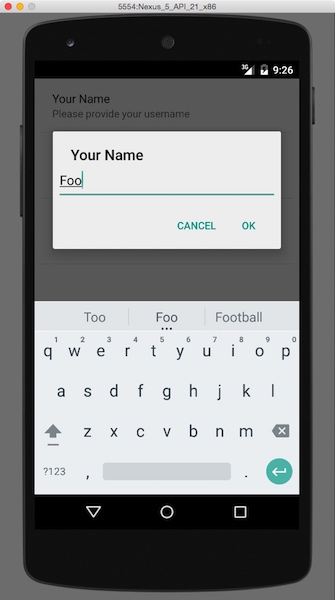
По мере программирования, это все, что вам нужно для реализации экрана настроек Android. (Для получения дополнительной информации, пожалуйста, см.  [that tutorial](http://rominirani.com/android-preferences-tutorial/).)

**Images (Изображения)**

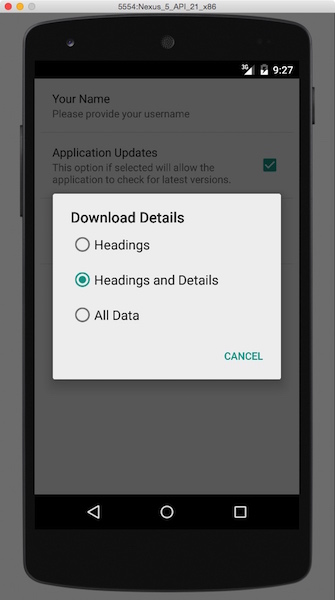
Следующие изображения показывают, как выглядит этот код при запуске в эмуляторе Android API Nexus 5 21. Во-первых, вот главный экран настроек:



Это изображение показывает, как выглядит текстовое поле” Ваше имя":



Там нет ничего, чтобы видеть с “обновления приложений” поле флажок, так что я буду пропустить это. Это последнее изображение показывает, как выглядит поле переключателя:



**Примечание: Android таблетки, phablets и ландшафтный режим**

Я еще ничего не сделал, чтобы учитывать более широкие дисплеи, такие как планшеты или использование ландшафтного режима на " phablets."Всякий раз, когда мне это нужно для моих собственных проектов, я буду обновлять этот учебник, или написать новый.

**Полный исходный код для моего основного действия запуска**

Я запускаю экран настроек с помощью кнопки / значка в меню параметров Android. Таким образом, это может помочь увидеть мою основную, пусковую активность, т. е. ту, которая запускает PreferenceActivity:

package com.alvinalexander.preferencestestapp;

import android.content.Intent;

import android.support.v7.app.ActionBarActivity;

import android.support.v4.app.Fragment;

import android.os.Bundle;

import android.view.LayoutInflater;

import android.view.Menu;

import android.view.MenuItem;

import android.view.View;

import android.view.ViewGroup;

public class MainActivity extends ActionBarActivity {

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

if (savedInstanceState == null) {

getSupportFragmentManager().beginTransaction()

.add(R.id.container, new PlaceholderFragment())

.commit();

}

}

@Override

public boolean onCreateOptionsMenu(Menu menu) {

// Inflate the menu; this adds items to the action bar if it is present.

getMenuInflater().inflate(R.menu.menu\_main, menu);

return true;

}

/\*\*

\* react to the user tapping/selecting an options menu item

\*/

@Override

public boolean onOptionsItemSelected(MenuItem item) {

switch (item.getItemId()) {

case R.id.menu\_item\_new\_thingy:

//Toast.makeText(this, "ADD!", Toast.LENGTH\_SHORT).show();

Intent i = new Intent(this, MyPreferencesActivity.class);

startActivity(i);

return true;

default:

return super.onOptionsItemSelected(item);

}

}

/\*\*

\* A placeholder fragment containing a simple view.

\*/

public static class PlaceholderFragment extends Fragment {

public PlaceholderFragment() {

}

@Override

public View onCreateView(LayoutInflater inflater, ViewGroup container,

Bundle savedInstanceState) {

View rootView = inflater.inflate(R.layout.fragment\_main, container, false);

return rootView;

}

} //end fragment

} //end activity

Этот код работает с меню, определенные в XML, который я сохранил в файл  menu/menu\_main.xml:

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

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

xmlns:myapp="http://schemas.android.com/apk/res-auto">

<item android:id="@+id/menu\_item\_new\_thingy"

android:icon="@android:drawable/ic\_menu\_add"

android:title="@string/new\_thingy"

myapp:showAsAction="ifRoom" />

</menu>

(I use a + icon for my cheesy example; you should use something else in your real code.)

**Summary: Preferences tutorial for Android 5**

In summary, if you want to put together a “preferences” screen in your Android 5 applications, I hope this code and images is helpful.