[Хранение данных. Работа с файлами.](https://startandroid.ru/ru/uroki/vse-uroki-spiskom/138-urok-75-hranenie-dannyh-rabota-s-fajlami.html" \o "Урок 75. Хранение данных. Работа с файлами.)

- работаем с файлами

Работа с файлами в Android не сильно отличается от таковой в Java. В этом уроке рассмотрим, как записать/прочесть файл во внутреннюю память и на SD-карту.

Создадим проект:

**Project name**: P0751\_Files  
**Build Target**: Android 2.3.3  
**Application name**: Files  
**Package name**: ru.startandroid.develop.p0751files  
**Create Activity**: MainActivity

Заполним **strings.xml**:

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?>  <resources>      <string name="app\_name">P0751\_Files</string>      <string name="write\_file">Записать файл</string>      <string name="read\_file">Прочесть файл</string>      <string name="write\_file\_sd">Записать файл на SD</string>      <string name="read\_file\_sd">Прочесть файл с SD</string>  </resources> |

Рисуем экран **main.xml**:

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?>  <LinearLayout      xmlns:android="http://schemas.android.com/apk/res/android"      android:layout\_width="fill\_parent"      android:layout\_height="fill\_parent"      android:orientation="vertical">      <LinearLayout          android:layout\_width="match\_parent"          android:layout\_height="wrap\_content">          <Button              android:id="@+id/btnWrite"              android:layout\_width="wrap\_content"              android:layout\_height="wrap\_content"              android:text="@string/write\_file"              android:onClick="onclick">          </Button>          <Button              android:id="@+id/btnRead"              android:layout\_width="wrap\_content"              android:layout\_height="wrap\_content"              android:text="@string/read\_file"              android:onClick="onclick">          </Button>      </LinearLayout>      <LinearLayout          android:layout\_width="match\_parent"          android:layout\_height="wrap\_content">          <Button              android:id="@+id/btnWriteSD"              android:layout\_width="wrap\_content"              android:layout\_height="wrap\_content"              android:text="@string/write\_file\_sd"              android:onClick="onclick">          </Button>          <Button              android:id="@+id/btnReadSD"              android:layout\_width="wrap\_content"              android:layout\_height="wrap\_content"              android:text="@string/read\_file\_sd"              android:onClick="onclick">          </Button>      </LinearLayout>  </LinearLayout> |

4 кнопки, смысл которых понятен по тексту на них.

**MainActivity.java**:

|  |
| --- |
| package ru.startandroid.develop.p0751files;    import java.io.BufferedReader;  import java.io.BufferedWriter;  import java.io.File;  import java.io.FileNotFoundException;  import java.io.FileReader;  import java.io.FileWriter;  import java.io.IOException;  import java.io.InputStreamReader;  import java.io.OutputStreamWriter;    import android.app.Activity;  import android.os.Bundle;  import android.os.Environment;  import android.util.Log;  import android.view.View;    public class MainActivity extends Activity {      final String LOG\_TAG = "myLogs";      final String FILENAME = "file";      final String DIR\_SD = "MyFiles";    final String FILENAME\_SD = "fileSD";      /\*\* Called when the activity is first created. \*/    @Override    public void onCreate(Bundle savedInstanceState) {      super.onCreate(savedInstanceState);      setContentView(R.layout.main);    }      public void onclick(View v) {      switch (v.getId()) {      case R.id.btnWrite:        writeFile();        break;      case R.id.btnRead:        readFile();        break;      case R.id.btnWriteSD:        writeFileSD();        break;      case R.id.btnReadSD:        readFileSD();        break;      }    }      void writeFile() {      try {        // отрываем поток для записи        BufferedWriter bw = new BufferedWriter(new OutputStreamWriter(            openFileOutput(FILENAME, MODE\_PRIVATE)));        // пишем данные        bw.write("Содержимое файла");        // закрываем поток        bw.close();        Log.d(LOG\_TAG, "Файл записан");      } catch (FileNotFoundException e) {        e.printStackTrace();      } catch (IOException e) {        e.printStackTrace();      }    }      void readFile() {      try {        // открываем поток для чтения        BufferedReader br = new BufferedReader(new InputStreamReader(            openFileInput(FILENAME)));        String str = "";        // читаем содержимое        while ((str = br.readLine()) != null) {          Log.d(LOG\_TAG, str);        }      } catch (FileNotFoundException e) {        e.printStackTrace();      } catch (IOException e) {        e.printStackTrace();      }    }      void writeFileSD() {      // проверяем доступность SD      if (!Environment.getExternalStorageState().equals(          Environment.MEDIA\_MOUNTED)) {        Log.d(LOG\_TAG, "SD-карта не доступна: " + Environment.getExternalStorageState());        return;      }      // получаем путь к SD      File sdPath = Environment.getExternalStorageDirectory();      // добавляем свой каталог к пути      sdPath = new File(sdPath.getAbsolutePath() + "/" + DIR\_SD);      // создаем каталог      sdPath.mkdirs();      // формируем объект File, который содержит путь к файлу      File sdFile = new File(sdPath, FILENAME\_SD);      try {        // открываем поток для записи        BufferedWriter bw = new BufferedWriter(new FileWriter(sdFile));        // пишем данные        bw.write("Содержимое файла на SD");        // закрываем поток        bw.close();        Log.d(LOG\_TAG, "Файл записан на SD: " + sdFile.getAbsolutePath());      } catch (IOException e) {        e.printStackTrace();      }    }      void readFileSD() {      // проверяем доступность SD      if (!Environment.getExternalStorageState().equals(          Environment.MEDIA\_MOUNTED)) {        Log.d(LOG\_TAG, "SD-карта не доступна: " + Environment.getExternalStorageState());        return;      }      // получаем путь к SD      File sdPath = Environment.getExternalStorageDirectory();      // добавляем свой каталог к пути      sdPath = new File(sdPath.getAbsolutePath() + "/" + DIR\_SD);      // формируем объект File, который содержит путь к файлу      File sdFile = new File(sdPath, FILENAME\_SD);      try {        // открываем поток для чтения        BufferedReader br = new BufferedReader(new FileReader(sdFile));        String str = "";        // читаем содержимое        while ((str = br.readLine()) != null) {          Log.d(LOG\_TAG, str);        }      } catch (FileNotFoundException e) {        e.printStackTrace();      } catch (IOException e) {        e.printStackTrace();      }    }  } |

В **onclick** обрабатываем нажатия 4-х кнопок и вызываем соответствующие методы.

**writeFile** – запись файла во внутреннюю память. Используется метод [openFileOutput](http://developer.android.com/reference/android/content/Context.html" \l "openFileOutput(java.lang.String,%20int)" \t "_blank), который на вход берет имя файла и режим записи: [MODE\_PRIVATE](http://developer.android.com/reference/android/content/Context.html#MODE_PRIVATE) – файл доступен только этому приложению, [MODE\_WORLD\_READABLE](http://developer.android.com/reference/android/content/Context.html#MODE_WORLD_READABLE) – файл доступен для чтения всем, [MODE\_WORLD\_WRITEABLE](http://developer.android.com/reference/android/content/Context.html#MODE_WORLD_WRITEABLE) - файл доступен для записи всем, [MODE\_APPEND](http://developer.android.com/reference/android/content/Context.html#MODE_APPEND) – файл будет дописан, а не начат заново.

**readFile** – чтение файла из внутренней памяти. Используем метод [openFileInput](http://developer.android.com/reference/android/content/Context.html" \l "openFileInput(java.lang.String)" \t "_blank), принимающий на вход имя файла. Здесь и в методе записи внутреннего файла вы можете задать только имя файла, а каталог для ваших файлов вам уже выделен.

**writeFileSD** – запись файла на SD. Используем метод [getExternalStorageState](http://developer.android.com/reference/android/os/Environment.html" \l "getExternalStorageState()" \t "_blank) для получения состояния SD-карты. [Здесь](http://developer.android.com/reference/android/os/Environment.html#constants) можно посмотреть какие бывают состояния. Нам нужно MEDIA\_MOUNTED – когда SD-карта вставлена и готова к работе. Далее мы получаем путь к SD-карте (метод [getExternalStorageDirectory](http://developer.android.com/reference/android/os/Environment.html" \l "getExternalStorageDirectory()" \t "_blank)), добавляем свой каталог и имя файла, создаем каталог и пишем данные в файл.

**readFileSD** – чтение файла с SD. Все аналогично предыдущему методу, только файл не пишем, а читаем.

Осталось в манифест добавить разрешение на работу с файлами на SD - android.permission.WRITE\_EXTERNAL\_STORAGE.

Все сохраним и запустим. Видим экран с 4-мя кнопками:

Внутренняя память

Жмем кнопку **Записать файл**.  Видим в логе:

*Файл записан*

Проверим. Идем в **File Explorer** (Window > Show View > Other > Android > File Explorer) и открываем там папку **data/data/ru.startandroid.develop.p0751files/files** и видим там наш файл **file**.

Возвращаемся в эмулятор. Жмем **Прочесть файл** и в логе видим:

*Содержимое файла*

Это тот текст, который мы записывали в файл.

SD карта

Теперь жмем **Записать файл на SD**.

В логе видим:

*Файл записан на SD: /mnt/sdcard/MyFiles/fileSD*

Проверяем. Идем в **FileExplorer** и открываем там папку **mnt/sdcard/MyFiles/** а в ней файл **fileSD**.

Возвращаемся в эмулятор и жмем кнопку Прочесть файл с SD. В логе видим:

*Содержимое файла на SD*

Этот текст мы и записывали.

**mnt/sdcard** - обычно этот путь ведет к содержимому SD-карты. Возможно у вас он будет другой.

В общем, при работе с файлами на SD вы используете стандартные java механизмы. А при работе с внутренним хранилищем для удобства можно использовать методы-оболочки от Activity:

[openFileOutput](http://developer.android.com/reference/android/content/Context.html#openFileOutput(java.lang.String,%20int)) – открыть файл на запись

[openFileInput](http://developer.android.com/reference/android/content/Context.html#openFileInput(java.lang.String)) – открыть файл на чтение

[deleteFile](http://developer.android.com/reference/android/content/Context.html#deleteFile(java.lang.String)) – удалить файл

И есть метод [getFilesDir](http://developer.android.com/reference/android/content/Context.html" \l "getFilesDir()" \t "_blank) – возвращает объект File, соответствующий каталогу для файлов вашей программы. Используйте его, чтобы работать напрямую, без методов-оболочек.

Подробности работы в java с файловой системой я здесь описывать не буду. На нашем форуме пользователь SKR сделал отличную [памятку по работе с файлами](http://forum.startandroid.ru/viewtopic.php?f=26&t=860). Скорее всего, вы найдете там все что нужно.

Если у вас проверка SD-карты показывает, что карта недоступна (см. лог), то убедитесь в свойствах AVD, что у вас для SDCard указан Size или File. Если указаны, то попробуйте перезапустить AVD.