**[Reading and Writing a file to SD card sample program in Android](http://www.java-samples.com/showtutorial.php?tutorialid=1523)**

By: Ashley Emailed: 1503 times Printed: 1819 times

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Latest comments**   |  | | --- | | By: rohit kumar - [how this program is work](http://www.java-samples.com/showcomment.php?commentid=40085) | | By: Kirti - [Hi..thx for the hadoop in](http://www.java-samples.com/showcomment.php?commentid=40083) | | By: Spijker - [I have altered the code a](http://www.java-samples.com/showcomment.php?commentid=40082) | | By: ali mohammed - [why we use the java in ne](http://www.java-samples.com/showcomment.php?commentid=40081) | | By: ali mohammed - [why we use the java in ne](http://www.java-samples.com/showcomment.php?commentid=40080) | | By: mizhelle - [when I exported the data](http://www.java-samples.com/showcomment.php?commentid=40078) | | By: raul - [no output as well, i'm ge](http://www.java-samples.com/showcomment.php?commentid=40077) | | By: Rajesh - [thanx very much...](http://www.java-samples.com/showcomment.php?commentid=40076) | | By: Suindu De - [Suppose we are executing](http://www.java-samples.com/showcomment.php?commentid=40075) | |

This sample android program shows you how write and read a file from SD Card in Android. In this program four buttons are shown and a Edit box. When you type some text into the edit box and click, Save to SD Card button, the text is saved to a text file and saved to the SD Card. When you click clear button, the edit box contents are cleared. When you click, Read Sd card button the file is read from the SD card and the contents are copied to the edit box.

The FileDemo2.java file is as follows:

package com.javasamples;

import java.io.\*;

import android.app.Activity;

import android.os.Bundle;

import android.view.\*;

import android.view.View.OnClickListener;

import android.widget.\*;

public class FileDemo2 extends Activity {

// GUI controls

EditText txtData;

Button btnWriteSDFile;

Button btnReadSDFile;

Button btnClearScreen;

Button btnClose;

@Override

public void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.main);

// bind GUI elements with local controls

txtData = (EditText) findViewById(R.id.txtData);

txtData.setHint("Enter some lines of data here...");

btnWriteSDFile = (Button) findViewById(R.id.btnWriteSDFile);

btnWriteSDFile.setOnClickListener(new OnClickListener() {

public void onClick(View v) {

// write on SD card file data in the text box

try {

File myFile = new File("/sdcard/mysdfile.txt");

myFile.createNewFile();

FileOutputStream fOut = new FileOutputStream(myFile);

OutputStreamWriter myOutWriter =

new OutputStreamWriter(fOut);

myOutWriter.append(txtData.getText());

myOutWriter.close();

fOut.close();

Toast.makeText(getBaseContext(),

"Done writing SD 'mysdfile.txt'",

Toast.LENGTH\_SHORT).show();

} catch (Exception e) {

Toast.makeText(getBaseContext(), e.getMessage(),

Toast.LENGTH\_SHORT).show();

}

}// onClick

}); // btnWriteSDFile

btnReadSDFile = (Button) findViewById(R.id.btnReadSDFile);

btnReadSDFile.setOnClickListener(new OnClickListener() {

public void onClick(View v) {

// write on SD card file data in the text box

try {

File myFile = new File("/sdcard/mysdfile.txt");

FileInputStream fIn = new FileInputStream(myFile);

BufferedReader myReader = new BufferedReader(

new InputStreamReader(fIn));

String aDataRow = "";

String aBuffer = "";

while ((aDataRow = myReader.readLine()) != null) {

aBuffer += aDataRow + "\n";

}

txtData.setText(aBuffer);

myReader.close();

Toast.makeText(getBaseContext(),

"Done reading SD 'mysdfile.txt'",

Toast.LENGTH\_SHORT).show();

} catch (Exception e) {

Toast.makeText(getBaseContext(), e.getMessage(),

Toast.LENGTH\_SHORT).show();

}

}// onClick

}); // btnReadSDFile

btnClearScreen = (Button) findViewById(R.id.btnClearScreen);

btnClearScreen.setOnClickListener(new OnClickListener() {

public void onClick(View v) {

// clear text box

txtData.setText("");

}

}); // btnClearScreen

btnClose = (Button) findViewById(R.id.btnClose);

btnClose.setOnClickListener(new OnClickListener() {

public void onClick(View v) {

// clear text box

finish();

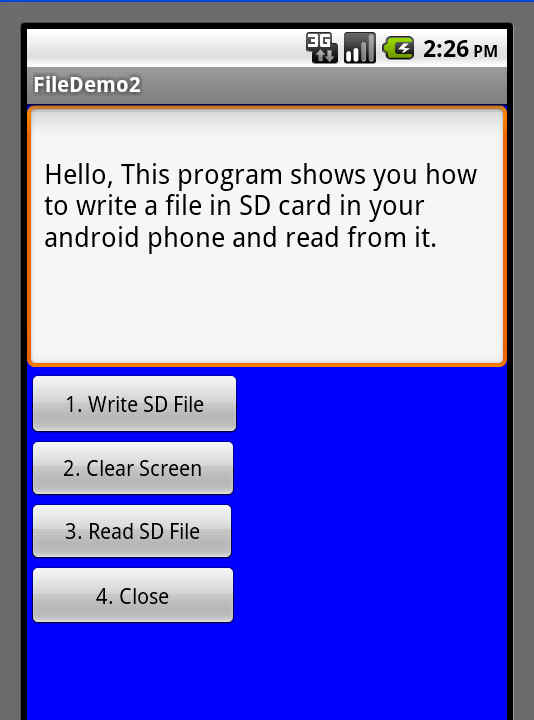
}

}); // btnClose

}// onCreate

}// AndSDcard

The output of this program will be as shown in the android emulator below.



The main.xml file in your res/layout folder is as follows:

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

<LinearLayout

android:id="@+id/widget28"

android:layout\_width="fill\_parent"

android:layout\_height="fill\_parent"

android:background="#ff0000ff"

android:orientation="vertical"

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

>

<EditText

android:id="@+id/txtData"

android:layout\_width="fill\_parent"

android:layout\_height="180px"

android:textSize="18sp" />

<Button

android:id="@+id/btnWriteSDFile"

android:layout\_width="143px"

android:layout\_height="44px"

android:text="1. Write SD File" />

<Button

android:id="@+id/btnClearScreen"

android:layout\_width="141px"

android:layout\_height="42px"

android:text="2. Clear Screen" />

<Button

android:id="@+id/btnReadSDFile"

android:layout\_width="140px"

android:layout\_height="42px"

android:text="3. Read SD File" />

<Button

android:id="@+id/btnClose"

android:layout\_width="141px"

android:layout\_height="43px"

android:text="4. Close" />

</LinearLayout>