**Saving image to sd card android example**

*by [Creatiosoft Team](https://www.creatiosoft.com/author/gamestore/)on November 8, 2012 , [1 comment](https://www.creatiosoft.com/saving-image-sd-card-android/" \l "comments)*

[https://secure.gravatar.com/avatar/?s=48&d=mm&r=g](https://www.creatiosoft.com/author/)

08/11/2012

To make the app more efficient we can  store the app image in sd card so that we don’t have to reload them each time we view the same activity again in the application. We can save images on sd card for various purposes depending upon are application requirement.

**Step 1**. Before we start saving the image to sd card ,we need to check whether the sd card is mounted or not. To do so, we use the following code:

if (android.os.Environment.getExternalStorageState().equals(                                  android.os.Environment.MEDIA\_MOUNTED))

{

Boolean mounted=true;

}

**Step 2.** For saving data in external storage i.e  sd card we need to add the following permission in the AndroidManifest.xml

<uses-permission android:name=”android.permission.WRITE\_EXTERNAL\_STORAGE”/>

**Step 3.**  If the sd card is mounted we use the following code

[sourcecode language=”java”]           // if sd card mounted  
                       if (android.os.Environment.getExternalStorageState().equals(  
                                                android.os.Environment.MEDIA\_MOUNTED))  
{  
              File sdCard = Environment.getExternalStorageDirectory();  
              File dir = new File(sdCard.getAbsolutePath() + "/Pictures");  
              dir.mkdirs();  
              File file = new File(dir, filename);  
              ByteArrayOutputStream baos = new ByteArrayOutputStream();  
              bitmap.compress(Bitmap.CompressFormat.PNG, 100, baos);  
              FileOutputStream f = null;  
              f = new FileOutputStream(file);  
                                              
                      if (f != null) {  
                             f.write(baos.toByteArray());  
                          f.flush();  
                             f.close();  
                                    }  
                        }  
                         
            }

[/sourcecode]

**Explaination:**

**Line 5**

The actual  file is referenced by a File object. A  File can be a directory or a regular file.

Environment.getExternalStorageDirectory() is used gets the Android external storage directory which is stored in File object sdcard.

**Line 6,**

File dir = new File(sdCard.getAbsolutePath() + “/Pictures”);

Here we are creating a new directory in the sd card named Pictures. Here we will be storing all our images

**Line 7**

dir.mkdirs();

Creates the directory named by the trailing filename of this file, including the complete directory path required to create this directory.

**Line 8**

File file = new File(dir, filename);

It constructs a new file using the specified directory and name. This will store our image.

**Line 9**

ByteArrayOutputStream  writes content to an byte array, which will be later users by FileOutputStream object .

**Line 10**

bitmap.compress(Bitmap.CompressFormat.PNG, 100, baos);

This  writes a compressed version of the bitmap to the specified outputstream.The target format of image is .png and 100 specifies compress for max quality.

**Line 12**

FileOutputStream(file);

FileOutputStream  writes to a file. If the output file  does not exist, a new file will be created.

**Line 15**

If f is not null then all the contents of the byte array buffer is written in it and then we close the stream to free resources.

This is all you need to do for saving images to the sd card. In similar way you can store different type of data no just images in sd card.