Как то так,

/\*\*

\* Копирует файл из ресурсов

\*/

public static void copyFileFromRaw(Context mContext,Sting FILE\_NAME)

{

getCurrentVersionCode();

File file1 = new File(FILE\_PATH);

try

{

//проверяем существует ли такой файл, если нет, то выпадет ошибка и нужно копровать, если нет, то ничего делать не будем

mContext.openFileInput(FILE\_NAME);

}

catch (IOException e)

{

e.printStackTrace();

try

{

Input Stream inputStream = this.getResources().openRawResource(R.raw.file);

//Context.MODE\_WORLD\_READABLE | Context.MODE\_WORLD\_WRITEABLE - дает доступ другим приложениям к этому файлу, если оне не нужен то можно

//поставить Context.Context.MODE\_PRIVATE,

FileOutputStream fout = mContext.openFileOutput(FILE\_NAME, Context.MODE\_WORLD\_READABLE | Context.MODE\_WORLD\_WRITEABLE);

copyFromTo(inputStream, fout);

}

catch (IOException e1)

{

e1.printStackTrace();

}

}

catch (Exception e)

{

e.printStackTrace();

}

}

//выполняет запрос для текстового файла

public static String executeHttpGet(String uri) throws Exception

{

String result = "";

try

{

URL url = new URL(uri);

BufferedReader in = new BufferedReader(new InputStreamReader(url.openStream()));

String str;

while ((str = in.readLine()) != null)

{

result +=str;

}

in.close();

}

catch (MalformedURLException e)

{

e.printStackTrace();

}

catch (IOException e)

{

e.printStackTrace();

}

return result;

}

//получет по ссылке новое содержимое файла, и перезаписывает

public void rewriteFileFromUrl(String url)

{

Sring file\_souce = executeHttpGet(url);

rewriteUpdateFile(file\_souce,FILE\_NAME);

}

/\*\*

\* Переписывает файл

\* тут, когда вы получите текст по ссылке, то просто посылаем в этот метод

\* @param str - текст файла

\*/

private void rewriteUpdateFile(String str, Sting FILE\_NAME)

{

try

{

InputStream is = new ByteArrayInputStream(str.getBytes());

FileOutputStream fout = mContext.openFileOutput(FILE\_NAME, Context.MODE\_WORLD\_READABLE | Context.MODE\_WORLD\_WRITEABLE);

copyFromTo(is, fout);

}

catch (IOException e1)

{

e1.printStackTrace();

}

}

/\*\*

\* Копирует из одного потока в другой информацию

\* @param fin поток копируемого файла

\* @param fout поток файла в который будет все записано

\* @throws IOException

\*/

private void copyFromTo(InputStream fin, OutputStream fout) throws IOException

{

byte[] b = new byte[1024];

int noOfBytes = 0;

while ((noOfBytes = fin.read(b)) != -1)

{

fout.write(b, 0, noOfBytes);

}

fin.close();

fout.close();

}

Как то так, накидал, но не проверял. Не претендует на самую лучшую реализацию, но для начала пойдет

/\*\*

\* Копирует файл из ресурсов

\*/

public static void copyFileFromRaw(Context mContext,Sting FILE\_NAME)

{

getCurrentVersionCode();

File file1 = new File(FILE\_PATH);

try

{

//проверяем существует ли такой файл, если нет, то выпадет ошибка и нужно копровать, если нет, то ничего делать не будем

mContext.openFileInput(FILE\_NAME);

}

catch (IOException e)

{

e.printStackTrace();

try

{

Input Stream inputStream = this.getResources().openRawResource(R.raw.file);

//Context.MODE\_WORLD\_READABLE | Context.MODE\_WORLD\_WRITEABLE - дает доступ другим приложениям к этому файлу, если оне не нужен то можно

//поставить Context.Context.MODE\_PRIVATE,

FileOutputStream fout = mContext.openFileOutput(FILE\_NAME, Context.MODE\_WORLD\_READABLE | Context.MODE\_WORLD\_WRITEABLE);

copyFromTo(inputStream, fout);

}

catch (IOException e1)

{

e1.printStackTrace();

}

}

catch (Exception e)

{

e.printStackTrace();

}

}

//выполняет запрос для текстового файла

public static String executeHttpGet(String uri) throws Exception

{

String result = "";

try

{

URL url = new URL(uri);

BufferedReader in = new BufferedReader(new InputStreamReader(url.openStream()));

String str;

while ((str = in.readLine()) != null)

{

result +=str;

}

in.close();

}

catch (MalformedURLException e)

{

e.printStackTrace();

}

catch (IOException e)

{

e.printStackTrace();

}

return result;

}

//получет по ссылке новое содержимое файла, и перезаписывает

public void rewriteFileFromUrl(String url)

{

Sring file\_souce = executeHttpGet(url);

rewriteUpdateFile(file\_souce,FILE\_NAME);

}

/\*\*

\* Переписывает файл

\* тут, когда вы получите текст по ссылке, то просто посылаем в этот метод

\* @param str - текст файла

\*/

private void rewriteUpdateFile(String str, Sting FILE\_NAME)

{

try

{

InputStream is = new ByteArrayInputStream(str.getBytes());

FileOutputStream fout = mContext.openFileOutput(FILE\_NAME, Context.MODE\_WORLD\_READABLE | Context.MODE\_WORLD\_WRITEABLE);

copyFromTo(is, fout);

}

catch (IOException e1)

{

e1.printStackTrace();

}

}

/\*\*

\* Копирует из одного потока в другой информацию

\* @param fin поток копируемого файла

\* @param fout поток файла в который будет все записано

\* @throws IOException

\*/

private void copyFromTo(InputStream fin, OutputStream fout) throws IOException

{

byte[] b = new byte[1024];

int noOfBytes = 0;

while ((noOfBytes = fin.read(b)) != -1)

{

fout.write(b, 0, noOfBytes);

}

fin.close();

fout.close();

}

|  |
| --- |
| [поделиться](http://ru.stackoverflow.com/a/216904/212443)[править](http://ru.stackoverflow.com/posts/216904/edit) |