Лабораторная 03,

ПББМС,3 курс, ПОиБМС

**Работа с файловой системой**

Разработайте приложение, имеющее следующий внешний вид:

|  |  |
| --- | --- |
|  |  |
|  |  |

С каждой клавишей на экране приложения должен быть связан Click-обработчик, заполняющий поля. Все поля заполняются на основе значений, получаемых с помощью методов объектов типа **File** или **Environment** в соответствии со следующей таблицей.

|  |  |
| --- | --- |
| **Поле** | **Заполнение (функция)** |
| Absolute | File.getAbsolutePath() |
| Name | File.getName() |
| Path | File.getPath() |
| Read/Write | File.canRead()/File.canWrite(), вывести yes или no |
| External | Environment.getEnternalStorageState() |

Объект типа **File** (java.io.File) должен быть получен для каждого Click-обработчика клавиши с помощью функции из следующей таблицы.

|  |  |
| --- | --- |
| **Поле** | **функция** |
| GETFILESDIR | AppCompatActivity.getFilesDir() |
| GETCACHEDIR | AppCompatActivity. getCacheDir() |
| GETEXTERNALFILESDIR | AppCompatActivity. getExternalFilesDir() |
| GETEXTERNALCACHEDIR | AppCompatActivity. getExternalCacheDir() |
| GETEXTERNALSTORAGEDIRECTORY | Environment. getExternalStorageDirectory() |
| GETEXTERNALSTORAGEPUBLICDIRECTORY | Environment.getExternalStoragePublicDirectory() |

Убедитесь в работоспособности приложения.

Внесите изменения в файл манифеста приложения.



Выполните приложение. Объясните разницу.

**Теоретические сведения.**

Память Android :

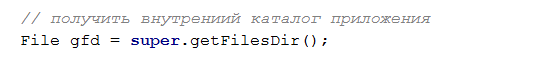
* ROM (постоянная память, ПЗУ),
* RAM (оперативная память, ОЗУ),
* Internal (внутренняя память),
* External (внешняя память).

ROM-память: энергонезависимая, только для чтения (Read Only Memory), записывается однократно при изготовлении, включает несколько разделов, одна из частей ОС, в которую можно писать с правами ROOT.

RAM-память: энергозависимая память, оперативная память для выполнения приложений.

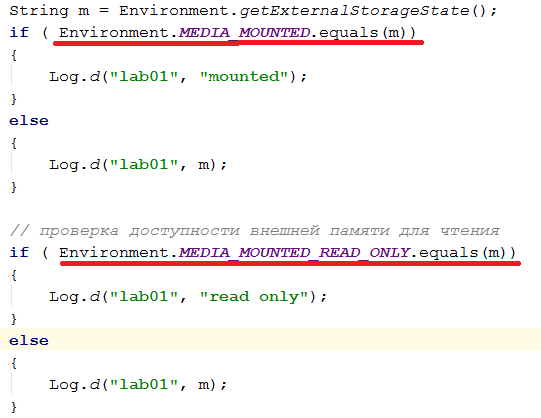
Internal-память: энергонезависимая память, для хранения apk-файлов (приложений) и данных приложений (в т.ч. БД), всегда доступна для приложения, не требует разрешения, при деинсталляции приложения все файлы удаляются.

External-память: энергонезависимая память, может располагаться как в собственной памяти устройства (эмуляция), так и на внешнем носителе (SD-карте), память SD-карты не всегда доступна: монтируется (подключается) и размонтируется (отключается), для доступа к External-памяти требуется разрешение, устанавливаемое в манифесте приложения, при деинсталляции приложения автоматически удалятся файлы только из директории getExternalFilesDir().



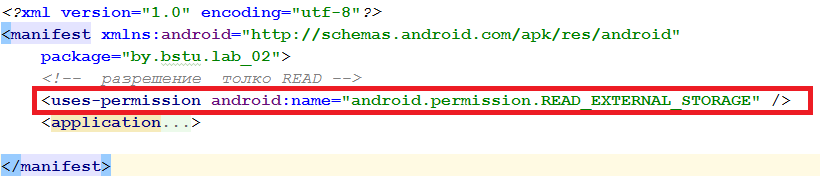


Проверка состояния External-памяти:



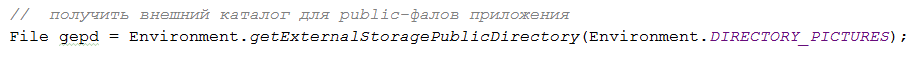
External-память: разрешение READ/WRITE

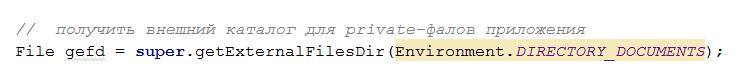




Для External-памяти:

* Public files – доступны для других приложений, при деинсталляции приложения файлы сохраняются.
* Private files – доступны для других приложений, при деинсталляции приложения файлы автоматически удаляются. Если указать параметр null вернет директорий External-корня для приложения.





**Основные функции Android API для работы с файлами:**

|  |  |
| --- | --- |
| [getFilesDir()](http://developer.android.com/reference/android/content/Context.html#getFilesDir()) | Возвращает File представляющий собой внутренний каталог вашего приложения. |
| getCacheDir() | Возвращает File представляющий собой внутренний каталог для временных файлов кэша вашего приложения. Если в системе возникает нехватка места, она может удалить ваши файлы кэша без предупреждения. |
| openFileOutput ([String](https://developer.android.com/reference/java/lang/String.html) name, int mode) | Возвращает FileOutputStream, который пишет в файл во внутреннем каталоге приложения. |
| openFileOutput ([String](https://developer.android.com/reference/java/lang/String.html) name, int mode) | Возвращает FileInputStream, который читает из файла во внутреннем каталоге приложения. |
| getDir (String name, int mode) | Возвращает File, представляющий собой каталог, в котором приложение может разместить свои собственные файлы данных. |
| fileList () | Возвращает массив строк с именами файлов, находящихся во внутреннем каталоге приложения. |
| deleteFile (String name) | Удалить указанный файл во внутреннем каталоге приложения. |
| Environment.getExternalStorageState () | Возвращает текущее состояние общего / внешнего носителя на данном пути.(MEDIA\_MOUNTED – можно работать) |
| Environment .getExternalStoragePublicDirectory ([String](https://developer.android.com/reference/java/lang/String.html) type) | Возвращает File, представляющий собой каталог для размещения файлов определенного типа (пользовательских). |
| getExternalFilesDir (String type) | Возвращает File, представляющий собой абсолютный путь к каталогу на внешнем устройстве, где приложение может поместить постоянные файлы. Эти файлы являются внутренними по отношению к приложениям, и как правило, не видны пользователю |
| getDatabasePath([String](https://developer.android.com/reference/java/lang/String.html) name) | Возвращает абсолютный путь в файловой системе, где хранится база данных. |
| getFreeSpace () | Возвращает количество нераспределенных байтов в разделе. |
| getTotalSpace () | Возвращает полный размер раздела |
| create NewFile() | Создает новый пустой, если файл с таким именем еще не существует. |
| Static createTempFile([String](https://developer.android.com/reference/java/lang/String.html) prefix,  [String](https://developer.android.com/reference/java/lang/String.html) suffix, [File](https://developer.android.com/reference/java/io/File.html) directory) | Создает новый пустой файл в указанном каталоге , используя данные префикса и суффикса строки для создания его имени. |
| delete() | Удаляет файл или каталог. |
| deleteOnExit() | Запрос на удаление файла, когда виртуальная Java-машина завершает свою работу (выключение телефона). |
| exists() | Проверяет, существует ли файл или каталог. |
| isDirectory() | Проверяет, является ли файл каталогом. |
| isFile() | Проверяет, является ли файл файлом. |
| length() | Возвращает размер файла. |
| mkdir() | Создаёт директорию. |
| mkdirs() | Создает каталог, в том числе любые необходимые, но несуществующие промежуточные каталоги. |
| renameTo([File](https://developer.android.com/reference/java/io/File.html) dest) | Переименовывает файл |
| getName() | Возвращает имя файла |
| getParent() | Возвращает имя родительского каталога |
| getAbsolutePath() | Возвращает абсолютный путь файла, начиная с корня системы. Корневым элементом является символ слеша (/) |