15.02.2021

**VSTORE v 0.1**

Библиотека VSTORE представляет набор функций подсистемы DAQ.JavaLikeCalc для работы с хранилищем значений.

Назначение – сохранение/восстановление произвольных пользовательских данных типа String, Integer, Real.

Примеры использования: операции с пользовательскими уставками, сохранение значений атрибутов между запусками openSCADA.

Библиотека реализована поверх подмножества пользовательского API openSCADA, предоставляющего доступ к БД с помощью SQL запросов.

В настоящее время реализован функционал сохранения/восстановления одиночных значений. Предполагается расширить функционал до оперирования пресетами, т.е. именованными наборами значений.

**Установка**

Библиотека оформлена в виде БД типа sqlite и подключается к проекту openSCADA путем добавления БД lib\_vstore.db в дерево БД. После этого в дереве проекта появится новая ветка /DAQ/JavaLikeCalc/vstore/. Библиотека готова к работе.

**Содержимое**

Библиотека экспортирует следующий набор функций:

|  |  |
| --- | --- |
| initStorage | Инициализация нового хранилища |
| setValueS | Установка значения типа String |
| setValueI | Установка значения типа Integer |
| setValueR | Установка значения типа Real |
| rmValue | Удаление значения из хранилища |
| getValueS | Запрос значения типа String |
| getValueI | Запрос значения типа Integer |
| getValueR | Запрос значения типа Real |
| test\_vstore | Проверка работоспособности библиотеки |

**Описание работы**

Порядок работы.

1. Перед началом использования хранилища его нужно инициализировать.

Для этого нужно вызвать функцию initStorage().

1. После инициализации хранилища можно сохранить в нем значение, вызвав одну из функций: setValueS(), setValueR(), setValueI().
2. Восстановить сохраненное значение или проверить наличие значения в хранилище можно с помощью функций: getValueS(), getValueR(), getValueI().
3. Удалить запись о значении из хранилища можно, вызвав rmValue().

**Описание функций**

initStorage()

|  |  |  |  |
| --- | --- | --- | --- |
| dbDB | String | Input | Идентификатор БД для содержания хранилища |
| force | Boolean | Input | Принудительная инициализация хранилища, если оно уже существует |
| retcode | Integer | Return | Код возврата |
| t\_err | String | Output | Текстовое сообщение об ошибке |

При вызове этой функции в БД с идентификатором dbDB будет сделана попытка создания таблицы с именем `vstore` и следующей структурой:

`vstore` (`**vid**` TEXT PRIMARY KEY,  
 `**preset**` TEXT,  
 `**v\_text**` TEXT NOT NULL,  
 `**v\_int**` INTEGER NOT NULL,  
 `**v\_real**` REAL NOT NULL)

Если флаг force=true, то таблица удаляется и пересоздается заново. Все ее содержимое теряется. Полезно, если производится переход на новую версию библиотеки vstore, в которой изменилась структура данных.

Если force=false, то при существовании таблицы в БД ничего не меняется.

Коды возврата:

|  |  |  |
| --- | --- | --- |
| **retcode** | **t\_err** | **Описание** |
| 0 | 0:Storage successfully created. | Хранилище ранее не существовало и успешно создано |
| 1 | 1:Error accessing the storage DB '%1'. | Ошибка доступа к БД для содержания хранилища |
| 2 | 2:Storage table already exists. | Попытка инициализировать хранилище, если оно уже существует и force=false. Никаких изменений в БД произведено не было. |
| 3 | 3:SQL request error: “…” | Ошибка в SQL запросе. Скорее всего это означает баг в коде библиотеки |
| 4 | 4:Storage successfully reinitialized. | Успешная попытка инициализировать хранилище, если оно уже существует и force=true |

Пример вызова:

dbDB = "SQLite.vstore\_test";

var t\_err;

retcode = DAQ.JavaLikeCalc.lib\_vstore.initStorage(dbDB, true, t\_err);

setValueS(), setValueR(), setValueI()

|  |  |  |  |
| --- | --- | --- | --- |
| vid | String | Input | Имя переменной, значение которой требуется установить |
| dbDB | String | Input | Идентификатор БД для содержания хранилища |
| setval | String/Real/Integer | Input | Значение, которое требуется присвоить переменной |
| t\_err | String | Output | Текстовое сообщение об ошибке |
| retcode | Integer | Return | Код возврата |

Коды возврата

|  |  |  |
| --- | --- | --- |
| **retcode** | **t\_err** | **Описание** |
| 0 | 0:Value updated successfully. | Успешная попытка установить значение переменной, которая до уже содержалась в хранилище. |
| 1 | 1:Error accessing the storage DB '%1'. | Ошибка доступа к БД для содержания хранилища |
| 2 | 2:Value not found but inserted successfully. | Успешная попытка установить значение переменной, которой до этого не было в хранилище. |
| 3 | 3:SQL request error: “…” | Ошибка в SQL запросе. Скорее всего это означает баг в коде библиотеки |

Пример вызова:

dbDB = "SQLite.vstore\_test";

var t\_err;

retcode = DAQ.JavaLikeCalc.lib\_vstore.setValueI("testInt", dbDB, rand(100).toInt(), t\_err);

getValueS, getValueR, getValueI

|  |  |  |  |
| --- | --- | --- | --- |
| vid | String | Input | Имя переменной, значение которой требуется восстановить |
| dbDB | String | Input | Идентификатор БД для содержания хранилища |
| retval | String/Real/Integer | Output | Значение, которое имеет переменная в хранилище |
| t\_err | String | Output | Текстовое сообщение об ошибке |
| retcode | Integer | Return | Код возврата |

Коды возврата

|  |  |  |
| --- | --- | --- |
| **retcode** | **t\_err** | **Описание** |
| 0 | 0:OK | Успешная попытка запросить значение переменной, которая уже содержится в хранилище. Ее значение будет содержаться в retval. |
| 1 | 1:Error accessing the storage DB '%1'. | Ошибка доступа к БД для содержания хранилища |
| 2 | 2:Value '%1' not found in the storage DB '%2'. | Переменная с запрашиваемым именем в БД не содержится. Значение retval будет установлено в <EVAL>. |
| 3 | 3:SQL request error: “…” | Ошибка в SQL запросе. Скорее всего это означает баг в коде библиотеки |

Пример вызова:

dbDB = "SQLite.vstore\_test";

var retvalI;

var t\_err;

retcode = DAQ.JavaLikeCalc.lib\_vstore.getValueI("testInt", dbDB, retvalI, t\_err);