**assignstringlist**

*Копирование содержимого одного списка строк в другой список строк.*

**Синтаксис:**

**assignstringlist**(*src\_id*, *dest\_id*)*;*

**Аргументы:**

*src\_id* – идентификатор списка строк, из которого копируются строки,

*dest\_id* – идентификатор списка строк, в который копируются строки.

**Описание:**

*assignstringlist(src\_id, dest\_id)* – копируется содержимое списка строк с идентификатором *src\_id* в список строк с идентификатором *dest\_id.*

**Результат:**

нет

**Пример:**

|  |  |
| --- | --- |
|  | **var**  s\_01:string = "string1" + chr(13) + "string2";  //создание списков строк  slist = **createstringlist**;  slist\_copy = **createstringlist;**  //загрузим строку s\_01 в список  **settextstringlist**(slist, s\_01);  //скопируем список slist в slist\_copy  **assignstringlist**(slist, slist\_copy);  //уничтожим списки  **freeobject(**slist**);**  **freeobject(**slist\_copy**);** |

В процессе выполнения примера при помощи функции *createstringlist* будут созданы списки строк с идентификатором *slist* и  *slist\_copy*. В список строк *slist* при помощи функции *settextstringlist* будет загружена строка, объявленная с разделителем «перевод строки», в результате чего переменная *slist* будет содержать идентификатор списка строк

“” *string1*”

“*string2*””.

Далее, при помощи функции *assignstringlist* в список строк с идентификатором *slist\_copy* будет скопирован список строк *slist*.

В результате переменная *slist* будет содержать идентификатор списка строк

“”string1”

“string2””

, переменная *slist\_copy* будет содержать идентификатор списка строк

“”string1”

“string2””

– копию списка строк с идентификатором *slist*.

Перед завершением примера списки строк удаляются при помощи функции *freeobject*.