

## Как прочитать текстовый файл в табличную часть

&НаКлиенте

Процедура ЗаполнитьИзФайла (Команда)

```
Проводник = Новый ДиалогВыбораФайла (РежимДиалогаВыбораФайла.Открытие);  
Проводник.Фильтр = "Текстовый документ | *.txt";  
  
ЧтоДелатьПослеВыбораФайла = Новый ОписаниеОповещения ("ПослеВыбораФайла", ЭтотОбъект);  
Проводник.Показать (ЧтоДелатьПослеВыбораФайла);
```

КонецПроцедуры

### 1. Создание диалога выбора файла:

- Проводник = Новый ДиалогВыбораФайла (РежимДиалогаВыбораФайла.Открытие) - создаёт новый диалог для выбора файла в режиме открытия (а не сохранения).

### 2. Установка фильтра файлов:

- Проводник.Фильтр = "Текстовый документ | \*.txt" - задаёт фильтр так, что пользователь будет видеть только текстовые файлы с расширением .txt.

### 3. Подготовка обработчика события:

- ЧтоДелатьПослеВыбораФайла = Новый ОписаниеОповещения ("ПослеВыбораФайла", ЭтогоОбъект) - создаёт описание оповещения, которое указывает, что после выбора файла должна быть вызвана процедура ПослеВыбораФайла текущего объекта.

### 4. Отображение диалога:

- Проводник.Показать (ЧтоДелатьПослеВыбораФайла) - показывает диалог выбора файла и передаёт ему обработчик события выбора файла.

```

&НаКлиенте
Процедура ПослеВыбораФайла(ВыбранныеФайлы, ДополнительныеПараметры) Экспорт

    Если ВыбранныеФайлы = Неопределено Тогда
        Возврат;
    КонецЕсли;

    ПутьКФайлу = ВыбранныеФайлы[0];

    Текст = Новый ТекстовыйДокумент;
    Текст.Прочитать(ПутьКФайлу, КодировкаТекста.UTF8);

    КоличествоСтрок = Текст.КоличествоСтрок();

    Для НомерСтроки=1 По КоличествоСтрок Цикл
        ТекСтрока =Текст.ПолучитьСтроку(НомерСтроки);
        ЧастиСтроки = СтрРазделить(ТекСтрока, ";");
        Наименование = ЧастиСтроки[0];
        Количество = ЧастиСтроки[1];
        Цена = ЧастиСтроки[2];

        НоваяСтрока = Объект.Товары.Добавить();
        НоваяСтрока.Номенклатура = НайтиНоменклатуруПоНаименованию(Наименование);
        НоваяСтрока.Количество = Количество;
        НоваяСтрока.Цена = Цена;

        ИменаРеквизитов = "ЕдиницаИзмерения,СтавкаНДС";
        РеквизитыНоменклатуры = ОбщегоНазначенияВызовСервера.ПолучитьРеквизитыНоменклатуры(НоваяСтрока.Номенклатура, ИменаРеквизитов);

        Если РеквизитыНоменклатуры <> Неопределено Тогда
            НоваяСтрока.ЕдиницаИзмерения = РеквизитыНоменклатуры.ЕдиницаИзмерения;
            НоваяСтрока.СтавкаНДС = РеквизитыНоменклатуры.СтавкаНДС;
        КонецЕсли;

        ОбработкаТабличныхЧастейКлиент.РассчитатьСуммыВСтрокеТабличнойЧасти(НоваяСтрока);

    КонецЦикла;
КонецПроцедуры

```

## 1. Объявление процедуры:

Процедура ПослеВыбораФайла(ВыбранныеФайлы,  
ДополнительныеПараметры) Экспорт

- Вызывается после выбора файла пользователем
- Принимает массив выбранных файлов и дополнительные параметры
- Экспорт означает, что процедура доступна из других модулей

## 2. Проверка на наличие выбранных файлов:

Сору

Если ВыбранныеФайлы = Неопределено Тогда  
Возврат;  
КонецЕсли;

- Если файлы не выбраны, процедура завершается

## 3. Получение пути к файлу:

ПутьКФайлу = ВыбранныеФайлы[0];

- Берется первый файл из массива

#### 4. Чтение файла:

```
Текст = Новый ТекстовыйДокумент;  
Текст.Прочитать(ПутьКФайлу, КодировкаТекста.UTF8);
```

- Создается объект текстового документа
- Файл читается в кодировке UTF-8

#### 5. Обработка содержимого файла:

```
Сору  
  
КоличествоСтрок = Текст.КоличествоСтрок();  
  
Для НомерСтроки=1 По КоличествоСтрок Цикл  
    ТекСтрока = Текст.ПолучитьСтроку(НомерСтроки);  
    ЧастиСтроки = СтрРазделить(ТекСтрока, ";");  
    Наименование = ЧастиСтроки[0];  
    Количество = ЧастиСтроки[1];  
    Цена = ЧастиСтроки[2];
```

- Определяется количество строк в файле
- Каждая строка разделяется по символу ";" (CSV-формат)
- Извлекаются: Наименование, Количество и Цена

#### 6. Добавление новой строки в табличную часть:

```
Сору  
  
НоваяСтрока = Объект.Товары.Добавить();  
НоваяСтрока.Номенклатура =  
НайтиНоменклатуруПоНаименованию(Наименование);  
НоваяСтрока.Количество = Количество;  
НоваяСтрока.Цена = Цена;
```

- Создается новая строка в табличной части "Товары"
- Заполняются реквизиты строки

#### 7. Получение дополнительных реквизитов номенклатуры:

```
Сору  
  
ИменаРеквизитов = "ЕдиницаИзмерения,СтавкаНДС";  
РеквизитыНоменклатуры =  
ОбщегоНазначенияВызовСервера.ПолучитьРеквизитыНоменклатуры(
```

```
НоваяСтрока.Номенклатура,  
ИменаРеквизитов  
);
```

- Запрашиваются с сервера дополнительные свойства номенклатуры

#### 8. Заполнение дополнительных реквизитов:

```
Сору  
  
Если РеквизитыНоменклатуры <> Неопределено Тогда  
    НоваяСтрока.ЕдиницаИзмерения =  
    РеквизитыНоменклатуры.ЕдиницаИзмерения;  
    НоваяСтрока.СтавкаНДС = РеквизитыНоменклатуры.СтавкаНДС;  
КонецЕсли;
```

- Если реквизиты получены, они сохраняются в строку

#### 9. Расчет суммы строки:

```
ОбработкаТабличныхЧастейКлиент.РассчитатьСуммуВСтрокеТабличнойЧас  
ти(НоваяСтрока);
```

- Вызывается метод для расчета суммы в строке

&НаСервереБезКонтекста

Функция НайтиНоменклатуруПоНаименованию(Наименование)

```
Запрос = Новый Запрос;  
Запрос.Текст = "ВЫБРАТЬ  
| Номенклатура.Ссылка КАК Ссылка  
| ИЗ  
| Справочник.Номенклатура КАК Номенклатура  
| ГДЕ  
| Номенклатура.Наименование = &Наименование";  
  
Запрос.УстановитьПараметр("Наименование", Наименование);  
  
Результат = Запрос.Выполнить();  
  
Если НЕ Результат.Пустой() Тогда  
  
    Выборка = Результат.Выбрать();  
  
    Выборка.Следующий();  
  
    Номенклатура = Выборка.Ссылка;  
Иначе  
    Номенклатура = Справочники.Номенклатура.ПустаяСсылка();  
КонецЕсли;  
  
Возврат Номенклатура
```

КонецФункции

## 1. Объявление функции:

Сору

Функция НайтиНоменклатуруПоНаименованию(Наименование)

- Функция принимает один параметр - наименование номенклатуры
- Работает на сервере без контекста (не имеет доступа к клиентским данным)

## 2. Создание запроса:

Сору

```
Запрос = Новый Запрос;  
Запрос.Текст = "ВЫБРАТЬ  
| Номенклатура.Ссылка КАК Ссылка  
| ИЗ  
| Справочник.Номенклатура КАК Номенклатура  
| ГДЕ  
| Номенклатура.Наименование = &Наименование";
```

- Создается новый объект запроса
- Формируется текст запроса на встроенном языке 1С
- Запрос выбирает ссылку на номенклатуру, где наименование соответствует параметру

### 3. Установка параметра:

Сору

```
Запрос.УстановитьПараметр("Наименование", Наименование);
```

- Параметр "Наименование" связывается с переданным значением

### 4. Выполнение запроса:

Сору

```
Результат = Запрос.Выполнить();
```

- Запрос выполняется в базе данных

### 5. Обработка результатов:

Сору

```
Если НЕ Результат.Пустой() Тогда
```

```
    Выборка = Результат.Выбрать();
```

```
    Выборка.Следующий();
```

```
    Номенклатура = Выборка.Ссылка;
```

```
Иначе
```

```
    Номенклатура = Справочники.Номенклатура.ПустаяСсылка();
```

```
КонецЕсли;
```

- Если результат не пустой, извлекается первая найденная запись
- Если ничего не найдено, возвращается пустая ссылка на номенклатуру

### 6. Возврат результата:

Сору

```
Возврат Номенклатура
```

- Функция возвращает либо найденную ссылку, либо пустую ссылку