

# Глава 3. Описание типовых событий

События изменения записей в базе данных .....	2
НаДобавить .....	2
НаПослеДобавить .....	2
НаСохранить .....	3
НаПослеСохранить .....	4
НаСкопировать .....	4
НаПослеСкопировать .....	5
НаРедактировать .....	5
НаИнициализировать .....	6
НаУдалить .....	6
НаПослеУдалить .....	7
Фильтрация и поиск записей .....	8
ФильтрЗаписей .....	8
НайтиЗаписьВыборки .....	8
Изменения в окнах и диалогах .....	9
НаОкноОткрыть .....	9
НаОкноВыбораОткрыть .....	10
ИмяДиалога .....	10
НаДиалогОткрыть .....	11
НаИзменениеПозиции .....	12
НаДиалогСохранить .....	12
НаДиалогОтменить .....	13
НаПомощникДалее .....	14
НаПомощникНазад .....	14
Свойства и изменение полей .....	15
ЦветПоля .....	15
НаРассчитать .....	15
НаИзменениеПоля .....	16
События документов .....	17
НаДокументОткрыть .....	17
НаДокументИнициализировать .....	17
НаДокументЗапустить .....	18
НаИзменениеСобытий .....	18
НаДокументОстановить .....	19
ПередНаДокументЗакрыть .....	19
НаДокументЗакрыть .....	20
Прочие события .....	20
ИмяОтчета .....	20
НаСообщение .....	21
НаУстройствоВвода .....	21

# События изменения записей в базе данных

## НаДобавить

Вызывается непосредственно перед вставкой в базу данных.

### Синтаксис

**функция** НаДобавить ( )

### Параметры

Используется без параметров.

### Возвращает

Ничего не возвращает.

### Пример

При создании исходящего платежа необходимо установить очередность, равную «б», если иная не проставлена:

```
функция НаДобавить ( )  
{  
    # Начальная инициализация очередности платежа  
    Если ( !Очередность )  
        Очередность = б;  
    вернуть Следфункция ( ) ;  
}  
# функция описана в обработчике «Исходящие  
платежи.sel-hdl»
```

## НаПоследДобавить

Вызывается непосредственно после вставки записи в базу данных.

### Синтаксис

**функция** НаПоследДобавить ( )

### Параметры

Используется без параметров.

## Возвращает

Ничего не возвращает.

## Пример

Функция широко применяется при формировании регламентированной отчётности, например, при создании сведений СЗВ:

```
функция НаПослеДобавить ( )
{
    ЗагрузитьФункции ( "ОтчетностьПФ" );
    ПФОбработчикЗаполненияФорм ( ТекущийКонтекст );
    Вернуть СледФункция ;
}
# функция описана в обработчике «СЗВ--4.sel-hdl»
```

## НаСохранить

Вызывается непосредственно перед сохранением/заменой записи в базе данных.

## Синтаксис

```
функция НаСохранить ( )
```

## Параметры

Используется без параметров.

## Возвращает

Ничего не возвращает.

## Пример

Функция широко применяется при формировании регламентированной отчётности, например, при заполнении сведений АДВ-11:

```
функция НаСохранить ( )
{
    перем оОкно = БазовоеОкно ( ) ;
    Если ( Размер ( Год ) != 4 или Год > Год (
оОкно.РасчПериод ) )
        Ошибка ( "Год введен неверно!" ) ;
    вернуть СледФункция ;
}
```

```
# функция описана в обработчике «Формат АДВ-11 1  
07_00.rec-hdl»
```

## НаПослеСохранить

Вызывается непосредственно после сохранения/замены записи в базе данных.

### Синтаксис

```
функция НаПослеСохранить ( )
```

### Параметры

Используется без параметров.

### Возвращает

Ничего не возвращает.

## НаСкопировать

Вызывается при копировании записи, например, по <F5>.

### Синтаксис

```
функция НаСкопировать ( пСтараяЗапись )
```

### Параметры

пСтараяЗапись (объект) - запись, которую необходимо скопировать.

### Возвращает

Ничего не возвращает.

### Пример

Функция применяется, например, при копировании отчетности с ошибками:

```
функция НаСкопировать ( пСтараяЗапись )  
{  
    СообщитьОбОшибке ( "Копирование запрещено!" );  
    Вернуть ( Нет );  
}  
# функция описана в обработчике «Формы отчетности с
```

ошибками .sel-hdl»

## НаПослеСкопировать

Вызывается после копирования записи.

### Синтаксис

функция **НаПослеСкопировать** (пСтараяЗапись)

### Параметры

пСтараяЗапись (объект) - запись, которую скопировали.

### Возвращает

Ничего не возвращает.

## НаРедактировать

Вызывается при попытке редактирования записи.

### Синтаксис

функция **НаРедактировать** (пСобытие)

### Параметры

пСобытие (текст) - может принимать одно из трех значений: «Создание» - нажато <Ins> или <Ctrl+Ins>; «Редактирование» - нажато <F3>; «Копирование» - нажато <F5>.

### Возвращает

**Строка** - принимает одно из трех значений:

- § “ ” (*пустая строка*) – продолжить обычную процедуру;
- § «Сохранить» - диалог не открывать, запись сохранить;
- § «Отменить» - диалог не открывать, запись не сохранять.

### Пример

Функция широко применяется в электронной отчетности, например, при заполнении сведений о заработке и доходе в форме АДВ-10, где добавление новых записей не допускается:

функция **НаРедактировать** ( пСобытие )

```

{
    ВыборПо ( пСобытие )
    {
        выбор "Создание": вернуть "Отменить";
    }
    Вернуть Следфункция;
}
# функция описана в обработчике «АДВ-10 список.rec-
hdl»

```

## НаИнициализировать

Вызывается при инициализации записи.

### Синтаксис

**функция НаИнициализировать (пЗаписьСкопирована)**

### Параметры

**пЗаписьСкопирована** (число) - если не ноль, то запись строится по <F5>.

### Возвращает

**Число** - если ноль, то запись проинициализирована, иначе инициализация не прошла.

### Пример

Нужно сделать скопированную запись узлом иерархии:

```

функция НаИнициализировать ( ЗаписьСкопирована )
{
    Если ( !ЭтоУзел () и Раздел.ЭтоКорень () )
        Узел ();
    вернуть Следфункция;
}
# функция описана в обработчике «Исходящая почта-
полная.sel-hdl»

```

## НаУдалить

Вызывается непосредственно перед удалением записи из базы данных, например, при нажатии клавиши <Del>.

### Синтаксис

**функция НаУдалить ( )**

## Параметры

Используется без параметров.

## Возвращает

**Строка** - принимает одно из трех значений:

- § *«Удалить»* - удалить запись;
- § *«Прервать»* - прервать удаление записи. Все изменения в базе данных, сделанные в процессе удаления, будут отменены;
- § *«Пропустить»* - не удалять запись, но остальные изменения в базе данных будут сделаны.

## Пример

Функция может использоваться, например, при попытке удаления отчетности с ошибками из реестра:

```
функция НаУдалить ( )  
{  
    СообщитьОбОшибке ( "Удаление запрещено!" );  
    Вернуть ( Нет );  
}  
# функция описана в обработчике «Формы отчетности с  
ошибками.sel-hdl»
```

## НаПослеУдалить

Вызывается непосредственно после удаления записи из базы данных.

## Синтаксис

**функция НаПослеУдалить ( )**

## Параметры

Используется без параметров.

## Возвращает

Ничего не возвращает.

## Фильтрация и поиск записей

### ФильтрЗаписей

Вызывается для фильтрации записей в выборке.

#### Синтаксис

**функция** `ФильтрЗаписей()`

#### Параметры

Используется без параметров.

#### Возвращает

**Да** (логическое значение) - запись попадает в выборку.

**Нет** (логическое значение) - запись не попадает в выборку.

#### Пример

Функция может использоваться, например, при распознавании ошибочных сведений пачки ПФ:

```
функция ФильтрЗаписей()
{
    перемен оИнф = ПрочитатьОбъект( ПараметрыОтчета );
    Если ( Есть ( оИнф.мОшибки ) )
    {
        Если ( оИнф.мОшибки.КолОшибок или
оИнф.мОшибки.КолПредупреждений )
            Вернуть Да;
        Иначе
            Вернуть Нет;
    }
    Иначе
        Вернуть Нет;
}
# функция описана в обработчике «Ошибочные сведения
пачки ПФ.sel-hdl»
```

### НайтиЗаписьВыборки

Вызывается для поиска очередной записи выборки.

## Синтаксис

функция НайтиЗаписьВыборки (пВперед)

## Параметры

пВперед (логическое значение) - направление поиска: «Да» соответствует поиску вперед, «Нет» - поиску назад.

## Возвращает

Да (логическое значение) - нашли следующую запись выборки.

Нет (логическое значение) - записей больше нет.

# Изменения в окнах и диалогах

## НаОкноОткрыть

Вызывается при создании окна/диалога с браузером.

## Синтаксис

функция НаОкноОткрыть ( )

## Параметры

Используется без параметров.

## Возвращает

Ничего не возвращает.

## Пример

При открытии окна «Выбор налогоплательщика» в электронной отчетности происходит сброс параметров «ДатНач» и «ДатКнц» с тем, чтобы все налогоплательщики были доступны для выбора:

```
функция НаОкноОткрыть ( )
{
    перем пПараметры = ПараметрыВыборки ( ) ;
    пПараметры.ДатНач = пПараметры.ДатКнц =
пПараметры.ДатКнц = пПараметры.ДатКнц = 00.00.00 ;
    УстановитьПараметрыВыборки ( пПараметры ) ;
    Вернуть СледФункция ;
}
```

```
# функция описана в обработчике  
«Налогоплательщики (Мастер) .sel-hdl»
```

## НаОкноВыбораОткрыть

Вызывается при открытии окна выбора кнопкой связи.

### Синтаксис

```
функция НаОкноВыбораОткрыть (пОкноОткудаВызвали)
```

### Параметры

пОкноОткудаВызвали (объект) - окно диалога, на кнопке связи которого стоит обработчик.

### Возвращает

Ничего не возвращает.

### Пример

При выборе гос. инспекции для организации в окне выбора сразу предлагается выбор из списка налоговых органов:

```
функция НаОкноВыбораОткрыть ( пОкноОткудаВызвали )  
{  
    .оформа = пОкноОткудаВызвали;  
    оформа .ПослатьСообщение ( "УстановитьСписокНИ",  
НаОкноВыбораОткрыть );  
    Вернуть ( СледФункция );  
}  
# функция описана в обработчике «Гос. инспекции.sel-  
hdl»
```

## ИмяДиалога

Возвращает имя диалога для редактирования данной записи.

### Синтаксис

```
функция ИмяДиалога (пИмя)
```

### Параметры

пИмя (текст) - текущее имя диалога.

## Возвращает

Ничего не возвращает.

## Пример

При выборе гос. инспекции для организации в окне выбора сразу предлагается выбор из списка налоговых органов:

```
функция НаОкноВыбораОткрыть ( пОкноОткудаВызвали )
{
    .оФорма = пОкноОткудаВызвали;
    оФорма.ПослатьСообщение ( "УстановитьСписокНИ" ,
НаОкноВыбораОткрыть );
    Вернуть ( СледФункция );
}
# функция описана в обработчике «Гос. инспекции.sel-
hdl»
```

## НаДиалогОткрыть

Вызывается при открытии диалога.

## Синтаксис

```
функция НаДиалогОткрыть ( )
```

## Параметры

Используется без параметров.

## Возвращает

Ничего не возвращает.

## Пример

Вот как, например, выполняется начальная инициализация флага «Возврат налога» во входящем платежном поручении:

```
функция НаДиалогОткрыть ( )
{
    Если ( НалогПлатежи.Загружена ( ) )
        ВозвратНалого = "+Возврат налога";
    ПоказатьСпрятатьПоля ( ) ;
    вернуть СледФункция ;
}
```

```
# функция описана в обработчике «Входящая  
платежка.dlg-hdl»
```

## НаИзменениеПозиции

Вызывается при изменении позиции в одной из таблиц диалога.

### Синтаксис

```
функция НаИзменениеПозиции (пИмяТабл)
```

### Параметры

пИмяТабл (текст) - имя таблицы в диалоге.

### Возвращает

Ничего не возвращает.

### Пример

Функция вызывается, например, при переходе от одной записи к другой в таблице диалога «Обновление справочника банков»: меняются значения полей сравнения в зависимости от выбора записи в таблице:

```
функция НаДиалогОткрыть ()  
{  
    Если ( НалогПлатежи.Загружена () )  
        ВозвратНалога = "+Возврат налога";  
    ПоказатьСпрятатьПоля ();  
    вернуть СледФункция ;  
}  
# функция описана в обработчике «Обновление  
справочника Банки.dlg-hdl»
```

## НаДиалогСохранить

Вызывается при нажатии кнопки "Сохранить" в диалоге.

### Синтаксис

```
функция НаДиалогСохранить ()
```

### Параметры

Используется без параметров.

## Возвращает

**Текст** - если пустая строка, то диалог будет закрыт, если непустая строка, то диалог не может быть закрыт, а возвращенная строка трактуется как имя поля и это поле становится активным.

## Пример

Функция, например, проверяет правильность заполнения поля «ИНН» (если он указан) при сохранении сведений о сотруднике:

```
функция НаДиалогСохранить ()
{
    ПроверитьИНН2 ();
    вернуть Следфункция ();
}
# функция описана в обработчике «Сотрудник.dlg-hdl»
```

## НаДиалогОтменить

Вызывается при нажатии кнопки "Отменить" в диалоге или клавиши <Esc>.

## Синтаксис

```
функция НаДиалогОтменить ()
```

## Параметры

Используется без параметров.

## Возвращает

Ничего не возвращает.

## Пример

Функция, например, проверяет изменения в протоколе проверки файла со статистической отчетностью:

```
функция НаДиалогОтменить ()
{
    ПроверитьИзмененияВфайле ( Да );
    Вернуть ( Следфункция );
}
# функция описана в обработчике «Статистика протокол
  проверки файла.dlg-hdl»
```

## НаПомощникДалее

Вызывается в диалог-помощнике при нажатии на кнопку "Далее".

### Синтаксис

**функция** НаПомощникДалее (пСледСтраница)

### Параметры

**пСледСтраница** (текст) - имя страницы, на которую будет переходить помощник.

### Комментарии

Показательные примеры использования функции в электронной отчетности при создании налогоплательщика («**Мастер создания налогоплательщика.dlg-hdl**») или при создании ответственного лица («**Мастер ответственного лица.dlg-hdl**»).

### Возвращает

**Да** (логическое значение) - можно переходить на следующую страницу.

**Нет** (логическое значение) - нельзя переходить на следующую страницу, помощник остается на текущей странице.

**Текст** - имя страницы, на которую нужно перейти.

## НаПомощникНазад

Вызывается в диалог-помощнике при нажатии на кнопку "Назад".

### Синтаксис

**функция** НаПомощникНазад (пПредСтраница)

### Параметры

**пПредСтраница** (текст) - имя страницы, на которую будет переходить помощник.

### Комментарии

Показательные примеры использования функции в электронной отчетности при создании налогоплательщика («**Мастер создания налогоплательщика.dlg-hdl**»).

тельщика.dlg-hdl») или при создании ответственного лица («Мастер ответственного лица.dlg-hdl»).

### **Возвращает**

**Да** (логическое значение) - можно переходить на предыдущую страницу.

**Нет** (логическое значение) - нельзя переходить на предыдущую страницу, помощник остается на текущей странице.

**Текст** - имя страницы, на которую нужно перейти.

## **Свойства и изменение полей**

### **ЦветПоля**

Вызывается для получения цвета поля.

### **Синтаксис**

**функция** ЦветПоля (пИмяПоля)

### **Параметры**

пИмяПоля (текст) - имя поля, чей цвет необходимо получить.

### **Возвращает**

**Целое** - номер цвета в 16-ти цветной палитре.

**Текст** - название цвета.

### **НаРассчитать**

Вызывается для расчета дополнительных полей выборки.

### **Синтаксис**

**функция** НаРассчитать ()

### **Параметры**

Используется без параметров.

### **Комментарии**

Функцию нужно использовать очень осторожно:

1. будут тормозить процессы, выполняемые в программе;
2. во время этого метода нельзя вызывать ф-ю «Отладить», могут возникнуть проблемы.

## Возвращает

Ничего не возвращает.

## Пример

Для перевода в настройке учета информации о налогообложении из флагов "Флаги/НО\_..." и "УСНОбъект" в строку "\_ВидНалог" можно использовать следующий код:

```

функция НаРассчитать ()
{
    _ВидНалог = "";
    Если( Флаг( флаги, "Налог на прибыль" ) )
        _ВидНалог = "Налог на прибыль";
    иначе
    {
        Если( Флаг( флаги, "УСН" ) )
            _ВидНалог = "УСН (" + УСНОбъект + ")";
    }
    Если( Флаг( флаги, "ЕНВД" ) )
        _ВидНалог += ( _ВидНалог ? " + " : "" ) +
"ЕНВД";
    вернуть Следфункция();
}
# функция описана в обработчике «Настройки
учета.sel-hdl»

```

## НаИзменениеПоля

Вызывается при изменении поля в диалоге.

## Синтаксис

```

функция НаИзменениеПоля (пИмяПоля)

```

## Параметры

пИмяПоля (текст) - имя измененного поля.

## Возвращает

Ничего не возвращает.

## Пример

Фрагмент проверки контрольной суммы корр. счета при заполнении карточки банка:

```
функция НаИзменениеПоля ( пИмяПоля )
{
    Результат = да;
    Если ( пИмяПоля == "КоррСчет" и !ПроверитьКорСчет (
КоррСчет, БИК ) )
        Сообщить ( "Неверная контрольная сумма
корр. счета" );
    Вернуть ( НаИзменениеПоляЛицо ( НаИзменениеПоля,
пИмяПоля ) и Результат );
}
# функция описана в обработчике «Банк.dlg-hdl»
```

# События документов

## НаДокументОткрыть

Вызывается при открытии (<Ctrl+Z>) документа.

## Синтаксис

```
функция НаДокументОткрыть ( )
```

## Параметры

Используется без параметров.

## Возвращает

Ничего не возвращает.

## НаДокументИнициализировать

Вызывается при инициализации документа или при смене правила операции.

## Синтаксис

**функция** `НаДокументИнициализировать (пСобытие)`

## Параметры

**пСобытие** (текст) - возможные варианты события: «Создание» - при создании (<Ins>) нового документа; «Изменение» - при изменении правила операции.

## Возвращает

Ничего не возвращает.

## Пример

При оформлении прихода ТМЦ по товарному чеку, в накладной снимаем флаг НДС, установленный по умолчанию:

```
функция НаДокументИнициализировать ( пСобытие )  
{  
    УстановитьНДС ( Нет ) ;  
}  
# правило «Приобретены ТМЦ по тов. чеку».
```

## НаДокументЗапустить

Вызывается при запуске документа в трассировку (в обработку).

## Синтаксис

**функция** `НаДокументЗапустить ( )`

## Параметры

Используется без параметров.

## Возвращает

Ничего не возвращает.

## НаИзменениеСобытий

Вызывается при любом изменении состояния трассировки (обработки) документа.

## Синтаксис

**функция НаИзменениеСобытий ( )**

### **Параметры**

Используется без параметров.

### **Возвращает**

Ничего не возвращает.

## **НаДокументОстановить**

Вызывается при остановке трассировки событий по документу.

### **Синтаксис**

**функция НаДокументОстановить ( )**

### **Параметры**

Используется без параметров.

### **Возвращает**

Ничего не возвращает.

## **ПередНаДокументЗакрыть**

Вызывается перед вызовом закрытия документа (<Ctrl+Z>).

### **Синтаксис**

**функция ПередНаДокументЗакрыть ( )**

### **Параметры**

Используется без параметров.

### **Возвращает**

Ничего не возвращает.

### **Пример**

Перед закрытием документа проверяем состояние НДС в документе, и приводим в соответствие с видом налогообложения:

**функция ПередНаДокументЗакрыть ( )**

```
{
    Документ . ПроверитьИУстановитьФлагНДС ("Покупка") ;
}
# правило «Модернизация ОС».
```

## НаДокументЗакрыть

Вызывается при закрытии документа (<Ctrl+Z>).

### Синтаксис

функция НаДокументЗакрыть ( )

### Параметры

Используется без параметров.

### Возвращает

Ничего не возвращает.

## Прочие события

### ИмяОтчета

Вызывается при печати по <F4>.

### Синтаксис

функция ИмяОтчета (пИмяОтчета)

### Параметры

пИмяОтчета (текст) - имя отчета, который будет печататься.

### Возвращает

**Да** (логическое значение) - отчет печатается так же, как если бы не был вызван обработчик.

**Нет** (логическое значение) - отчет не будет напечатан.

**Текст** - имя отчета, который должен быть напечатан.

## Пример

При печати сведений СЗВ-4-2 из реестра СЗВ-4 программа выдаст предупреждающее сообщение:

```
функция ИмяОтчета ( пИмяОтчета )
{
    ЗагрузитьФункции ( "ОтчетностьПФОперации" );
    перем оИнф = ПрочитатьОбъект ( ПараметрыОтчета );
    Если ( оИнф.ТипИндСведений == "СЗВ--4-2" )
    {
        Сообщить ( "Сведения СЗВ--4-2 необходимо
печатать из реестра пачек. Для перехода в этот
реестр нажмите кнопку \"формы отчетности\", встаньте
на запись пачки СЗВ--4-2 и нажмите F4." );
        Вернуть Нет;
    }
    Иначе
        вернуть Да;
}
# функция описана в обработчике «СЗВ--4.sel-hdl»
```

## НаСообщение

Вызывается при приходе сообщения, посланного функцией **ПослатьСообщение**.

### Синтаксис

```
функция НаСообщение (пСообщение , пДанные)
```

### Параметры

**пСообщение** (текст) - имя сообщения.

**пДанные** (объект) - объект с данными, которому посылается сообщение.

### Возвращает

**Да** (логическое значение) - сообщение было обработано.

**Нет** (логическое значение) - сообщение не было обработано.

## НаУстройствоВвода

Вызывается при приходе события от дополнительного устройства ввода.

## **Синтаксис**

**функция** НаУстройствоВвода (пКод, пУстройство)

## **Параметры**

**пКод** (текст) - код, пришедший от устройства.

**пУстройство** (объект) - запись устройства.

## **Возвращает**

**Да** (логическое значение) - событие отработано.

**Нет** (логическое значение) - событие не отработано и будет передано дальше.