API логических модулей

Обработки логических модулей размещаются в каталоге ExtFiles\LogicalModules.
В основной форме обработки необходимо реализовать определенный API.
Реквизиты и методы API логических модулей делится на обязательные и функциональные.

Обязательные реквизиты и методы API

Основная форма обработки экспортирует следующие реквизиты и методы.
Реквизиты:

```
_Источник - произвольный тип, при выполнении обработчика устанавливается объект-источник вызова.
_Отказ - булево, используется для отказа дальнейшего выполнения кода в источнике.
_РезультатВызова - произвольный тип, используется для получения результата в постобработчике после выполнения функции источника.
```

Методы:

```
Функция ИнициализироватьМодуль(ПоЗависимости = Ложь) экспорт
...
КонецФункции

Процедура ДеИнициализироватьМодуль() Экспорт
...
КонецПроцедуры
```

Функция ИнициализироватьМодуль

Функция является обязательной к экспорту. Вызывается при инициализации внешнего логического модуля.
Параметры:

- ПоЗависимости – Тип – Булево. По умолчанию данный параметр не передается в метод.

Необходим для определения, вызвался ли данный метод напрямую или по зависимости из стороннего логического модуля.
В данной функции выполняется подписка на определенные обработчики в конфигурации или в защищенной компоненте.
Чтобы подписать модуль логики на обработку определенного события нужно выполнить код:

```
МенеджерОбъектов.ПолучитьОбщийОбъект("LogicalModules").НазначитьОбработчик(ИмяСобытия, ЭтаФорма, Неопределено);
```

ИмяСобытия - строка, определяющая конкретный обработчик, например

«Авторизация.ВнешнееСобытие.ПредОбработка».

Процедура ДеИнициализироватьМодуль

Процедура является обязательной к экспорту. Вызывается при деинициализации внешнего логического модуля.
В данной процедуре выполняется код при завершении работы модуля.
Обязательно выполнить отключение подключенных обработчиков:

```
МенеджерОбъектов.ПолучитьОбщийОбъект("LogicalModules").СнятьОбработчик(ИмяСобытия, ЭтаФорма);
```

Функциональные методы API

Функциональные методы API содержат методы, которые позволяют расширить функционал программного обеспечения.
Данные методы делятся на 2 типа: предобработчик (выполняется в начале функции, позволяют полностью переопределить оригинальную функцию) и постобработчик (выполняется в конце функции, позволяет переопределить результат оригинальной функции).
Для работы данных методов необходимо:
1. подписать модуль логики на определенный пред-постобработчик (см. ИнициализироватьМодуль). 2. разместить в модуле, на который был подписан обработчик экспортную функцию или процедуру, которую вызывает оригинальная конфигурация или защищенная компонента.
Например,

```
Процедура Авторизация_ВнешнееСобытие_ПредОбработка(Источник, Событие, Данные)
Экспорт
__Отказ = Ложь;
...
_Отказ = __Отказ;
КонецПроцедуры
```

From:
https://wiki.ilexx.ru/ - **Штрих-М: Документация**

Permanent link:
https://wiki.ilexx.ru/doku.php?id=%D0%BA%D0%B0%D1%81%D1%81%D0%B8%D1%805:%D0%B2%D0%BD%D0%B5%D1%88%D0%BD%D0%B8%D0%B5%D0%BC%D0%BE%D0%B4%D1%83%D0%BB%D0%B8:api%D0%BB%D0%BE%D0%B3%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B8%D1%85%D0%BC%D0%BE%D0%B4%D1%83%D0%BB%D0%B5%D0%B9&rev=1551176074

Last update: **2019/02/26 10:14**