

# Раздел

Версия 8.0



Эта документация предоставляется с ограничениями на использование и защищена законами об интеллектуальной собственности. За исключением случаев, прямо разрешенных в вашем лицензионном соглашении или разрешенных законом, вы не можете использовать, копировать, воспроизводить, переводить, транслировать, изменять, лицензировать, передавать, распространять, демонстрировать, выполнять, публиковать или отображать любую часть в любой форме или посредством любые значения. Обратный инжиниринг, дизассемблирование или декомпиляция этой документации, если это не требуется по закону для взаимодействия, запрещены.

Информация, содержащаяся в данном документе, может быть изменена без предварительного уведомления и не может гарантировать отсутствие ошибок. Если вы обнаружите какие-либо ошибки, сообщите нам о них в письменной форме.

# Содержание

<b>Раздел</b>	<b>4</b>
Контейнеры раздела	4
Структура раздела	5
Создать раздел	14
Добавить пользовательскую колонку в реестр раздела	15
Настроить условия отображения реестра раздела	16
Добавить действие для записей раздела	17
Настроить блок быстрых фильтров раздела	18
Удалить раздел	19
<b>Добавить в раздел действие для единичной записи</b>	<b>20</b>
Создать схему замещающей модели представления раздела	20
Результат выполнения примера	23
<b>Добавить в раздел действие для нескольких записей</b>	<b>25</b>
Создать схему замещающей модели представления раздела	25
Результат выполнения примера	29
<b>Настроить обработку действия для нескольких записей раздела</b>	<b>30</b>
Создать схему замещающей модели представления раздела	30
Результат выполнения примера	34
<b>Настроить обработку действия с использованием справочника для нескольких записей раздела</b>	<b>35</b>
Создать схему замещающей модели представления раздела	36
Результат выполнения примера	39
<b>Настроить условия отображения реестра раздела</b>	<b>41</b>
Создать схему замещающей модели представления раздела	41
Результат выполнения примера	43
<b>Настроить блок быстрых фильтров раздела</b>	<b>44</b>
Создать схему замещающей модели представления раздела	44
Результат выполнения примера	48
<b>Схема BaseSectionV2</b>	<b>48</b>
Сообщения	48
Атрибуты	49
Методы	50

# Раздел

## Основы

**Раздел** — элемент интерфейса, который отражает определенную бизнес-сущность и содержит набор записей. Примерами раздела являются разделы [ *Контрагенты* ] ([ *Accounts* ]), [ *Контакты* ] ([ *Contacts* ]), [ *Активности* ] ([ *Activities* ]) и т. д. Разделы доступны в боковой панели, их можно сгруппировать в рабочие места для удобства работы отдельных ролей. Раздел описан в блоке статей [Раздел](#).

## Контейнеры раздела

Элементы пользовательского интерфейса приложения, которые относятся к разделу, размещены в соответствующих контейнерах. Контейнеры конфигурируются в базовой схеме раздела или схеме замещающей модели представления раздела. Контейнеры зависят от представления раздела.

**Представления** раздела:

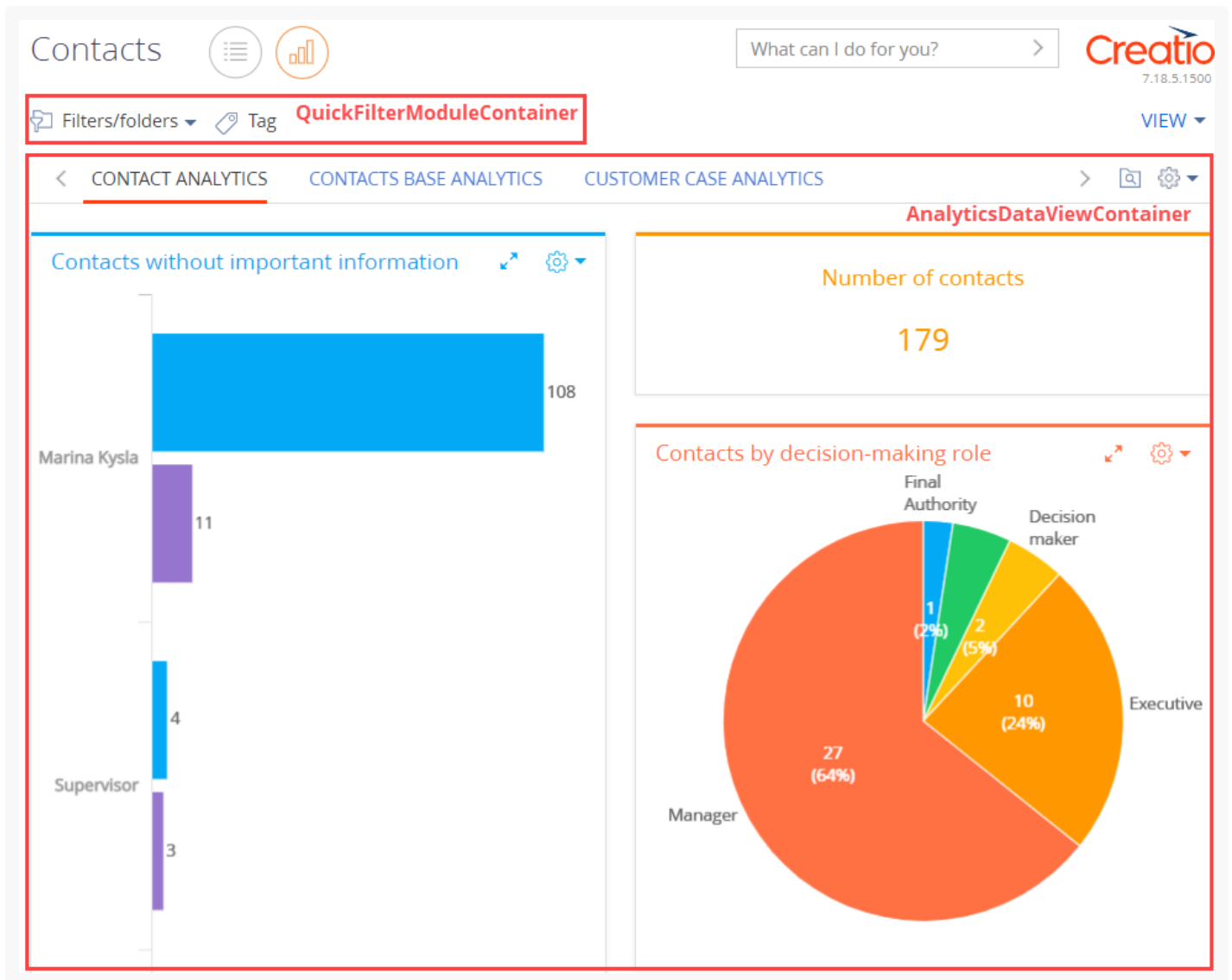
- Реестр раздела.
- Аналитика раздела.

**На заметку.** В приложении используются мета-имена html-контейнеров. На основании мета-имен приложение формирует фактические идентификаторы соответствующих html-элементов раздела.

Основные **контейнеры реестра** раздела представлены на рисунке ниже.

The screenshot displays the 'Contacts' section of the Creatio application. The interface includes a top navigation bar with the 'Contacts' title, a search bar, and the Creatio logo. Below the navigation bar, there is a header area with a 'NEW CONTACT' button, an 'ACTIONS' dropdown, and a 'VIEW' dropdown. The main content area is divided into two sections: a 'Filters/folders' section and a 'Tag' section. The 'Tag' section contains a table of contacts, with each row representing a contact record. The table columns include 'Job title', 'Email', 'Business phone', and 'Mobile phone'. The table is labeled 'GridDataViewContainer'. The 'Filters/folders' section is labeled 'QuickFilterModuleContainer'. The 'ACTIONS' dropdown is labeled 'ActionButtonsContainer'. The 'NEW CONTACT' button is labeled 'NEW CONTACT'. The 'VIEW' dropdown is labeled 'VIEW'. The 'Tag' section is labeled 'Tag'. The 'Filters/folders' section is labeled 'Filters/folders'. The 'Job title' column is labeled 'Job title'. The 'Email' column is labeled 'Email'. The 'Business phone' column is labeled 'Business phone'. The 'Mobile phone' column is labeled 'Mobile phone'. The 'OPEN' button is labeled 'OPEN'. The 'COPY' button is labeled 'COPY'. The 'DELETE' button is labeled 'DELETE'.

Основные **контейнеры аналитики** раздела представлены на рисунке ниже.



- Контейнер кнопок действий ( `ActionButtonsContainer` ) — содержит кнопку действия раздела и кнопку с выпадающим списком действий.
- Контейнер фильтров ( `QuickFilterViewContainer` ) — содержит фильтры и теги.
- Контейнер отображения реестра раздела ( `GridDataViewContainer` ) — содержит записи раздела. В активной записи реестра размещаются кнопки действий редактирования, копирования и удаления текущей записи. Отображается в реестре раздела.
- Контейнер отображения аналитики раздела ( `AnalyticsDataViewContainer` ) — содержит дашборды раздела. Отображается в аналитике раздела.

## Структура раздела

**Составляющие** раздела:

- **Реестр** — компонент, в котором в плиточном или списочном представлении отображается список записей раздела. Отображается в контейнере `GridDataViewContainer`.
- **Аналитика** — компонент, который используется для визуализации статистических данных с

помощью графиков, единичных показателей или списков. Блоки итогов и пользовательские дашборды отображаются в контейнере `AnalyticsDataViewContainer`.

- **Действия** — функциональный элемент, который представляет набор операций над активной записью реестра раздела. Действия вызываются при помощи кнопок, которые размещены в контейнере `ActionButtonsContainer`, и в активной записи реестра раздела.
- **Фильтр** — инструмент для поиска и сегментации записей реестра по заданным условиям. Отображается в контейнере `QuickFilterViewContainer`.
- **Тег** — метка, которая используется для сегментации записей вручную. Как и фильтр, отображается в контейнере `QuickFilterViewContainer`.

## Реестр

**Реестр** — элемент интерфейса для отображения перечня записей, которые добавлены в раздел или на деталь. Например, перечень контактов в разделе [ *Контакты* ] ([ *Contacts* ]).

**Виды** реестра раздела:

- **Вертикальный реестр** — способ отображения реестра, при котором можно переключаться между записями, не закрывая страницу.
- **Редактируемый реестр** — реестр записей, которые предоставляет возможность редактирования записей в самом реестре, без перехода к их страницам.

**Представления** реестра раздела:

- **Плиточное представление** — отображает поля записи реестра в несколько строк. Установлено по умолчанию.
- **Списочное представление** — отображает записи в виде простой таблицы, в которой каждой записи соответствует одна строка. При этом последовательность расположения полей в списочном представлении может не совпадать с последовательностью расположения полей в плиточном представлении.

Плиточное представление реестра раздела [ *Контакты* ] ([ *Contacts* ]) представлено на рисунке ниже.

The screenshot shows the 'Contacts' section of the Creatio interface. The header includes the 'Contacts' title, navigation icons, a search bar with the text 'What can I do for you?', and the Creatio logo with version '7.18.5.1500'. Below the header are buttons for 'NEW CONTACT' and 'ACTIONS'. The main area displays two contact records in a tiled view. The first record is for Alexander Wilson, CEO of Alpha Business, with contact information: Business phone +1 212 542 4238 and Mobile phone +1 212 854 7512. The second record is for Alice Phillips, CEO of Axiom, with contact information: Business phone +1 212 1440 5222 and Mobile phone +1 212 1204 5477. Both records are highlighted with a red border and labeled 'Запись реестра' (Record) and 'Активная запись реестра' (Active record). At the bottom of the records are buttons for 'OPEN', 'COPY', and 'DELETE'.

Списочное представление реестра раздела [ *Контакты* ] ([ *Contacts* ]) представлено на рисунке ниже.

Contact name ^	Account	Job title	Business phone	Mobile phone	Email
Nora Wesley	Gtech	Head of department	+1 213829 58 33		TyroneRigg@hotmail.com
Winter Hodge	Console Solutions	Sales manager	+1 212 566 84 89		winterhodge@gmail.com

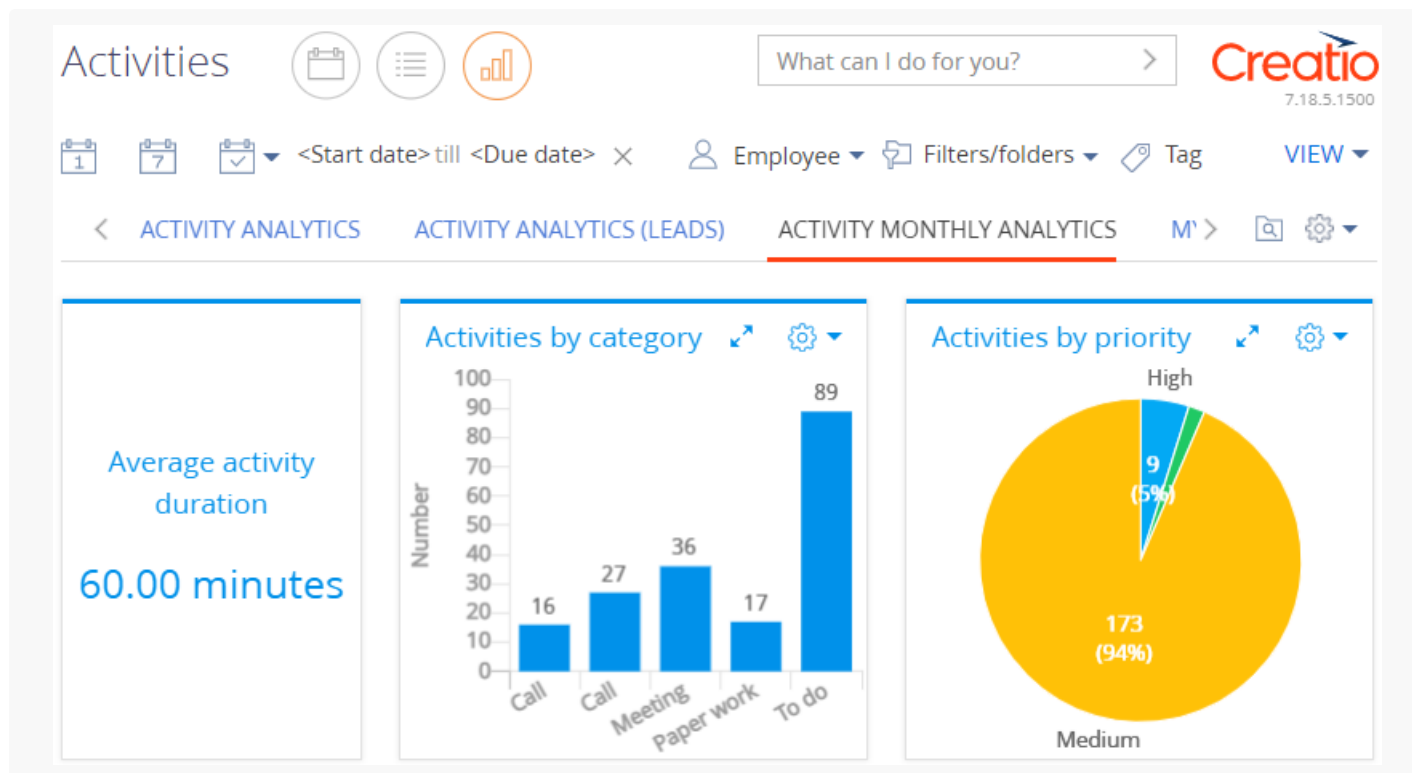
Каждому разделу соответствует конфигурационная схема, которая описывает структуру соответствующей таблицы базы данных с данными записи и содержит определенные инструкции по обработке этих данных. Например, конфигурирование раздела [ *Контакты* ] ([ *Contacts* ]) выполняется в схеме `ContactSectionV2`. Все схемы разделов являются наследниками базовой схемы `BaseSectionV2` раздела. Условно каждая строка таблицы соответствует записи реестра раздела. Например, разделу [ *Контакты* ] ([ *Contacts* ]) соответствует схема бизнес-объекта `Contact`, которая содержит перечень и свойства колонок таблицы `[Contact]`. По умолчанию, реестр раздела отображает не все колонки соответствующей таблицы базы данных. Это необходимо для избежания избыточности отображаемых данных. Посмотреть все колонки можно на [странице записи](#) раздела.

Внешний вид реестра, перечень отображаемых полей и сортировку данных позволяет настроить выпадающее меню кнопки [ *Вид* ] ([ *View* ]) панели инструментов раздела. Подробнее читайте в статье [Реестр раздела](#).

## Аналитика

**Назначение** аналитики раздела — анализ статистических данных разделов. По умолчанию раздел отображается в представлении реестра. Чтобы **перейти в представление аналитики раздела**, нажмите на кнопку [ *Итоги* ] ([ *Dashboards* ]) возле названия раздела. Чтобы **перейти в представление аналитики всех разделов**, откройте раздел [ *Итоги* ] ([ *Dashboards* ]).

Пример аналитики раздела [ *Активности* ] ([ *Activities* ]) представлены на рисунке ниже.



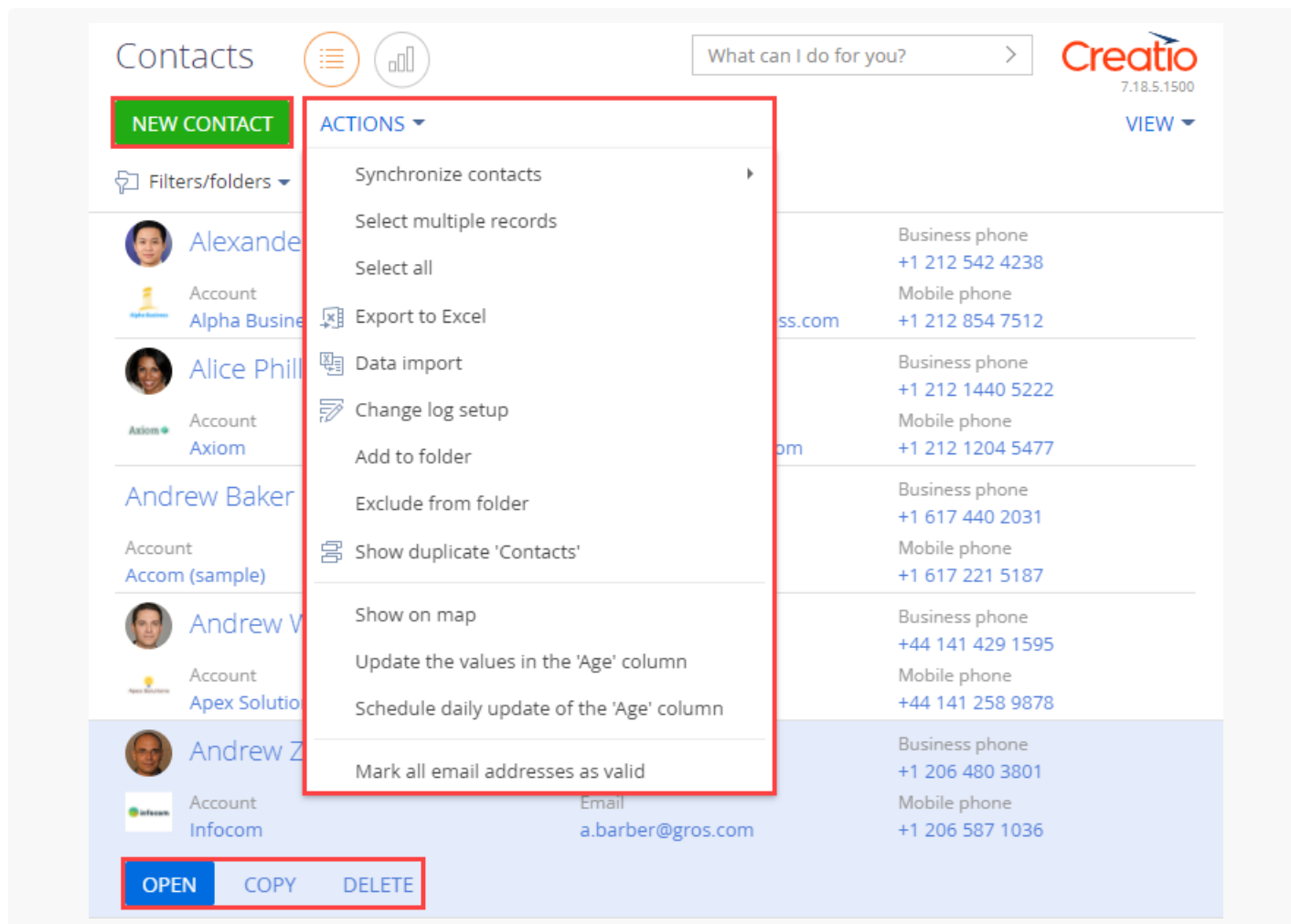
Аналитическая информация отображается на **дашбордах**. Дашборды описаны в статье [Дашборды](#) и блоке статей [Аналитика](#). Дашборды размещены на **панели итогов**. Панель итогов описана в статье [Добавить аналитику в раздел](#).

## Действия

**Действия раздела** — функциональные элементы, которые представляют собой перечень операций с одной или несколькими записями реестра. Действие можно вызвать при помощи кнопок разного вида, которые размещены как в контейнере действий текущего раздела, так и в контейнере активной записи.

Действия раздела [ *Контакты* ] ([ *Contacts* ]) представлены на рисунке ниже.





### Виды действий раздела:

- Стандартные.
- Дополнительные.

### Стандартные действия раздела [ Контакты ] ([ Contacts ]):

- [ Добавить контакт ] ([ New contact ]) — вызывает всплывающее окно для добавления и сохранения новой записи раздела.
- [ Открыть ] ([ Open ]) — открывает страницу активной записи раздела.
- [ Копировать ] ([ Copy ]) — открывает страницу записи раздела, копирует в нее данные активной записи и при сохранении создает новую запись.
- [ Удалить ] ([ Delete ]) — удаляет активную запись.
- [ Выбрать несколько записей ] ([ Select multiple records ]) — выполняет множественный выбор записей реестра.
- [ Выбрать все ] ([ Select all ]) — выполняет выбор всех записей реестра.
- [ Экспорт в Excel ] ([ Export to Excel ]) — экспортирует все записи реестра текущего раздела в \*.xlsx-файл. Экспорт данных описан в статье [Экспорт в Excel](#).
- [ Импорт данных ] ([ Data import ]) — импортирует в Creatio данные из \*.xlsx-файла. Импорт данных

описан в статье [Импорт из Excel](#).

- [ *Настроить журнал изменений* ] ([ *Change log setup* ]) — открывает страницу управления логированием и выбора колонок раздела, которые будут логироваться при изменении записи.

**Дополнительные** действия реализуют функциональность, которая зависит от бизнес-логики раздела. Дополнительные действия раздела [ *Контакты* ] ([ *Contacts* ]):

- [ *Запустить синхронизацию* ] ([ *Synchronize now* ]) — выполняет синхронизацию Creatio с контактами Google. Синхронизация контактов из Creatio в Google выполняется только для записей, которые отмечены тегом, указанным в настройках синхронизации. Синхронизация контактов описана в статье [Синхронизировать контакты и активности с Google](#).
- [ *Добавить аккаунт для синхронизации* ] ([ *Add new account for synchronization* ]) — выполняет синхронизацию с Google для Creatio Cloud. Синхронизация описана в статье [Синхронизировать контакты и активности с Google](#).
- [ *Поместить в группу* ] ([ *Add to folder* ]) — вызывает всплывающее окно для выбора группы, в которую необходимо поместить активную запись.
- [ *Исключить из группы* ] ([ *Exclude from folder* ]) — исключить активную запись из всех групп, в которую она входит.
- [ *Перейти к дублиям раздела 'Контакты'* ] ([ *Show duplicate 'Contacts'* ]) — открывает дополнительную страницу, которая содержит все предполагаемые дубли контактов. Записи добавляются на эту страницу автоматически после выполнения поиска дублей.
- [ *Показать на карте* ] ([ *Show on map* ]) — позволяет отобразить на карте местоположение активных контактов. По действию открывается окно с картой, на которой отмечены выбранные в реестре контакты. Если для всех выбранных контактов адрес не заполнен, то действие не выполняется. Если для некоторых контактов адрес не заполнен или заполнен некорректно, то в окне отображается соответствующая информация.
- [ *Обновить возраст* ] ([ *Update the values in the 'Age' column* ]) — обновляет значения в колонке [ *Возраст* ] ([ *Age* ]) на странице активного контакта.
- [ *Настроить время обновления возраста* ] ([ *Schedule daily update of the 'Age' column* ]) — вызывает всплывающее окно для настройки времени ежедневного обновления информации о возрасте контакта.
- [ *Снять признак 'Неактуальный' у email-адресов* ] ([ *Mark all email addresses as valid* ]) — устанавливает признак [ *Является актуальным* ] ([ *Valid* ]) email-адресу активной записи контакта.

## Фильтры

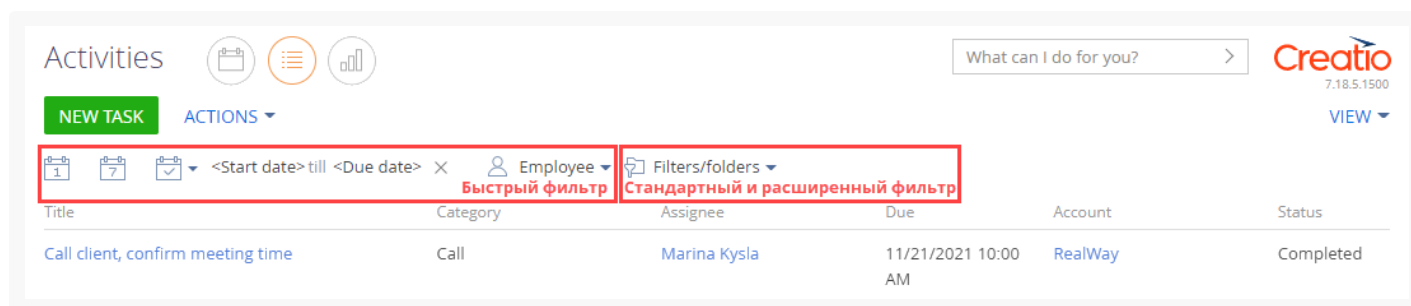
**Фильтр** — инструмент для поиска и сегментации записей системы по заданным условиям. Фильтры могут использоваться как самостоятельный инструмент в разделах Creatio. Также инструменты фильтрации применяются при выполнении других настроек, например, при настройке динамических групп, дашбордов, бизнес-процессов. Элементы управления фильтрами отображаются над реестром разделов приложения. Фильтры описаны в статье [Фильтры](#).

**Виды фильтров:**

- Быстрый фильтр.
- Расширенный фильтр.

- Стандартный фильтр.

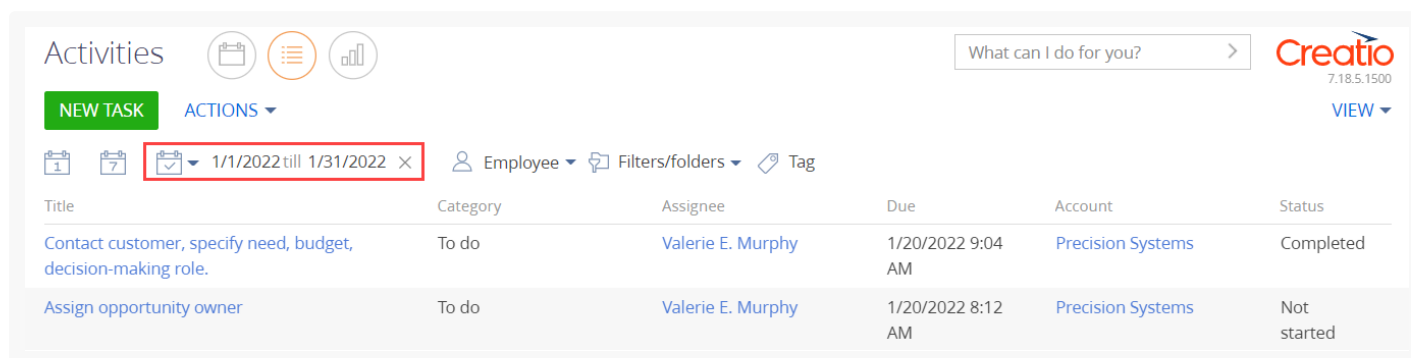
Управление быстрым фильтром осуществляется из панели инструментов, а управление стандартным и расширенным фильтрами — из меню [ *Фильтры/группы* ] ([ *Filters/folders* ]).



**Быстрый фильтр** — фильтр, который используется в реестре некоторых разделов Creatio для поиска данных по наиболее часто используемым параметрам (обычно — по периоду и ответственному). Например, быстрый фильтр присутствует в разделе [ *Активности* ] ([ *Activities* ]), поскольку чаще всего нужно просматривать активности одного сотрудника за указанный период времени. Быстрый фильтр описан в статье [Фильтры](#).

Чтобы **установить быстрый фильтр**, в выпадающем меню [ *Выбрать период* ] ([ *Select period* ]) выберите период, по которому планируется отфильтровать записи реестра раздела.

Пример быстрого фильтра, который фильтрует активности по текущему месяцу (в выпадающем меню [ *Выбрать период* ] ([ *Select period* ]) выбрано значение [ *Текущий месяц* ] ([ *Current month* ])), представлен на рисунке ниже.

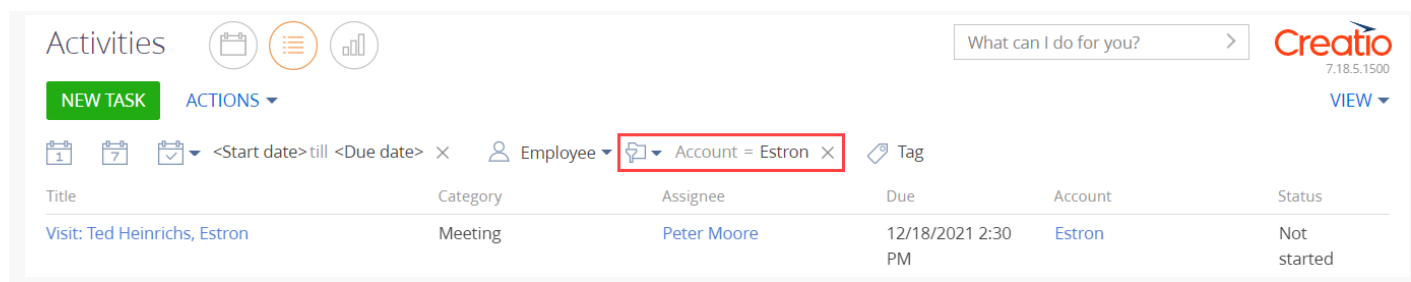


**Стандартный фильтр** — фильтр, который используется для быстрого поиска записей по значениям одной или нескольких колонок текущего раздела. Доступен в реестре большинства разделов. Стандартный фильтр описан в статье [Фильтры](#).

Чтобы **установить стандартный фильтр**:

- В выпадающем меню [ *Фильтры/группы* ] ([ *Filters/folders* ]) выберите пункт [ *Добавить условие* ] ([ *Add filter* ]).
- Выберите поле и значение поля, по которому планируется отфильтровать записи реестра раздела.

Пример стандартного фильтра, который фильтрует активности по значению контрагента (для поля [ *Контрагент* ] ([ *Account* ])) выбрано значение [ *Estron* ]), представлен на рисунке ниже.

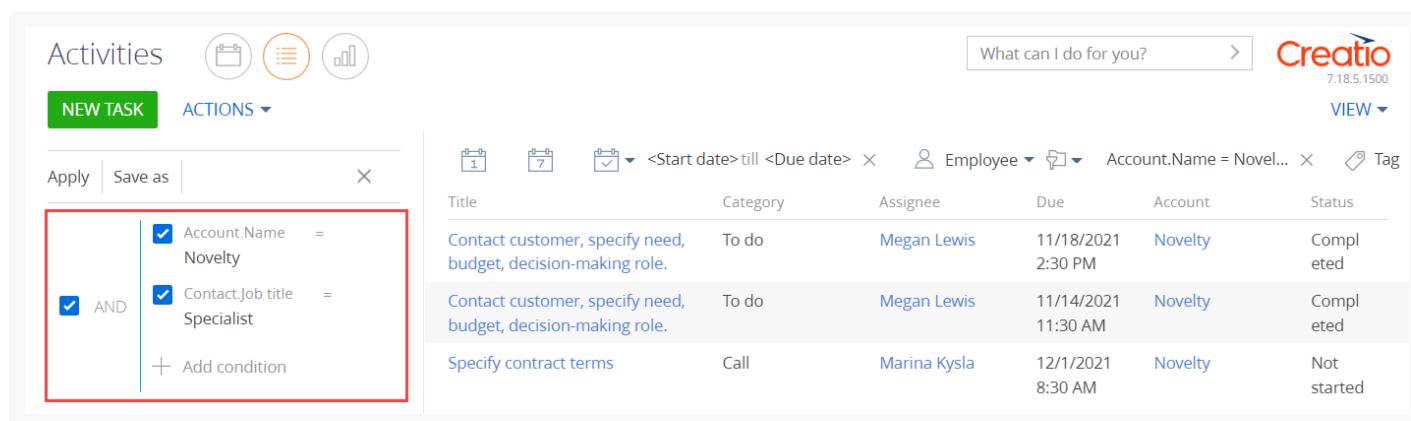


**Расширенный фильтр** — фильтр, который состоит из нескольких параметров и сложных условий поиска. Например, при помощи расширенной фильтрации вы можете отобразить в разделе [ *Активности* ] ([ *Activities* ]) все встречи по новым клиентам. Также, настроив условия расширенного фильтра, можно сохранить динамическую группу для дальнейшего использования. Расширенный фильтр описан в статье [Фильтры](#).

Чтобы **установить расширенный фильтр**:

- В выпадающем меню [ *Фильтры/группы* ] ([ *Filters/folders* ]) выберите пункт [ *Перейти в расширенный режим* ] ([ *Switch to advanced mode* ]).
- Выберите объект, колонку объекта и значение колонки, по которой планируется отфильтровать записи реестра раздела.

Пример расширенного фильтра, который фильтрует активности по значению контрагента (для поля [ *Название* ] ([ *Name* ]) объекта [ *Контрагент* ] ([ *Account* ]) выбрано значение [ *Novelty* ]) и по значению должности (для поля [ *Должность* ] ([ *Job title* ]) объекта [ *Контакт* ] ([ *Contact* ]) выбрано значение [ *Специалист* ] ([ *Specialist* ])), представлен на рисунке ниже.



**Группа** — инструмент фильтрации записей путем объединения их по определенным условиям. Группы могут формировать иерархическую структуру. Группы описаны в статье [Группы](#).

**Виды групп:**

- **Динамическая группа** — группа, содержащая только те записи раздела, которые соответствуют заданным условиям фильтрации. Например, динамической может быть группа [ *Новые клиенты* ] ([ *New client* ]), поскольку отобрать записи в такую группу можно, используя фильтр по дате создания записи в системе. Наполнение такой группы формируется и обновляется автоматически.
- **Статическая группа** — группа, содержащая только те записи раздела, которые были добавлены в нее пользователем (вручную или при конвертации из динамической группы). Например, статическими может быть группа [ *VIP-клиенты* ] ([ *VIP Customers* ]), поскольку решение о включении клиента в

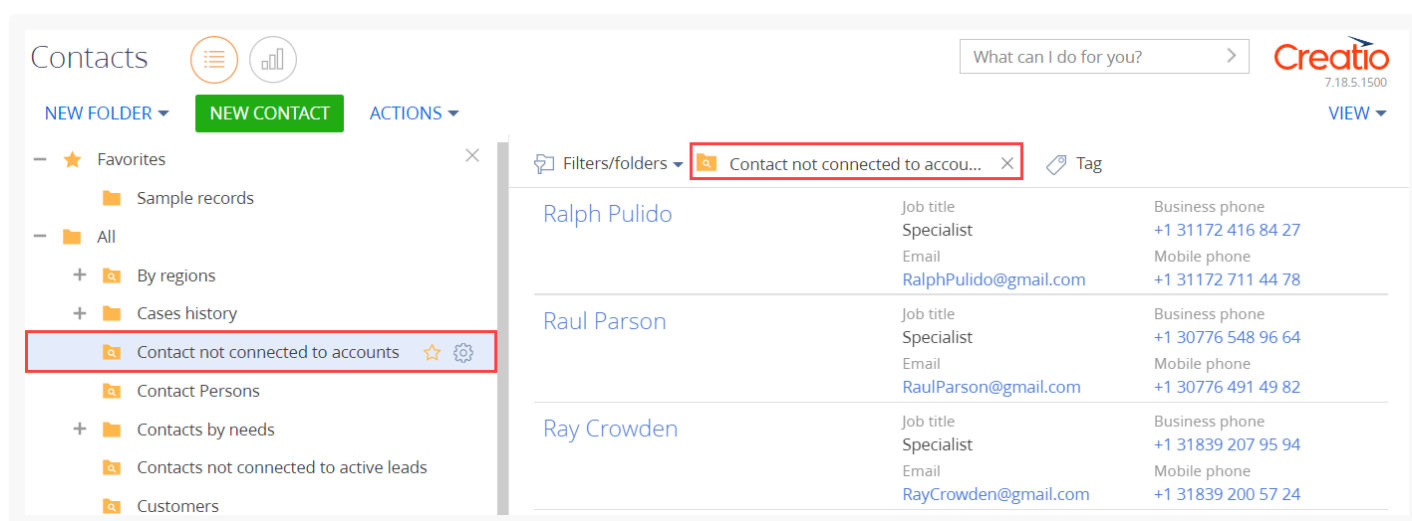
такую группу принимается менеджером или руководителем лично.

Включение записи в группу, как и исключение из нее, возможно только для статических групп. В динамической группе запись отображается автоматически, если она отвечает условиям фильтрации группы. Если запись не соответствует условиям фильтра группы, то она автоматически исключается из группы.

Чтобы **установить группу**:

- В выпадающем меню [ *Фильтры/группы* ] ([ *Filters/folders* ]) выберите пункт [ *Показать группы* ] ([ *Show folders* ]).
- Выберите группу, по которой планируется отфильтровать записи реестра раздела.

Пример группы контактов без контрагентов (группа [ *Контакты без контрагента* ] ([ *Contact not connected to accounts* ])), представлен на рисунке ниже.



## Теги

**Тег** — специальная метка, которую можно использовать для сегментации записей вручную. Например, тегировав записи в разделе [ *Контакты* ] ([ *Contacts* ]), можно выделить VIP-клиентов или определить черный список клиентов. Тегирование записей выполняется вручную. Чтобы **выбрать тег**, нажмите на кнопку [ *Тег* ] ([ *Tag* ]) и выберите тег, по которому планируется отфильтровать записи реестра раздела. Теги описаны в статье [Теги](#).

**Виды** тегов, которые предоставляет Creatio:

- **Личные** — теги, которые может видеть и использовать только тот сотрудник, который их создал. Любой пользователь системы может создать необходимое количество личных тегов. Ни администраторы системы, ни руководители не смогут увидеть личные теги сотрудников. Личные теги в Creatio отображаются зеленым цветом.
- **Корпоративные** — теги, которые отображаются для всех сотрудников компании. Любой сотрудник может установить или снять корпоративный тег. Создавать новые корпоративные теги могут все сотрудники/роли, которым предоставлено право на операцию [ *Управление корпоративными тегами* ] ([ *Corporate tags management* ], код `CanManageCorporateTags`). Корпоративные теги в Creatio отображаются голубым цветом.

- **Публичные** — теги, которые отображаются для всех сотрудников компании, а также для пользователей портала самообслуживания. Любой сотрудник может установить или снять публичный тег. Создавать новые публичные теги могут все сотрудники/роли, которым предоставлено право на операцию [ *Управление публичными тегами* ] ([ *Public tags management* ], код `CanManagePublicTags` ).  
Публичные теги в Creatio отображаются красным цветом.

Пример VIP-клиентов, которыми являются контакты (выбран корпоративный тег [ *VIP* ] ([ *VIP customer* ])), представлен на рисунке ниже.

The screenshot shows the Creatio Contacts interface. At the top, there's a 'Contacts' header with a search bar 'What can I do for you?' and the Creatio logo. Below the header, there are buttons for 'NEW CONTACT' and 'ACTIONS'. A 'Filters/folders' dropdown is set to 'VIP customer'. The main content area displays a contact card for Alexander Wilson, CEO of Alpha Business. The card includes his job title, email (a.wilson@alphabusiness.com), business phone (+1 212 542 4238), and mobile phone (+1 212 854 7512).

## Создать раздел

Создать раздел можно с использованием мастера разделов. Чтобы создать раздел, воспользуйтесь инструкцией, которая приведена в статье [Добавить новый раздел](#).

Схемы объектов и клиентские схемы, которые создаются в пользовательском пакете при создании раздела, приведены в таблице ниже.

Схемы, которые создаются мастером разделов

Название	Назначение	Родительский объект
Схемы объектов		
[Имя объекта раздела]	Основной объект раздела.	Базовый объект <code>BaseEntity</code> .
[Имя объекта раздела]Folder	Группа объекта. Служебный объект для корректной группировки записей в разделе. Формирует общую структуру дерева групп раздела.	Базовая группа <code>BaseFolder</code> .
[Имя объекта раздела]InFolder	Объект в группе. Служебный объект для корректной работы группировки записей в разделе. Определяет связи между записями раздела и группами, в которые они входят.	Базовый элемент <code>BaseItemInFolder</code> в группе.
[Имя объекта раздела]File	Объект для детали [ <i>Файлы и ссылки</i> ] ([ <i>Attachments</i> ]).	Файл <code>File</code> .
[Имя объекта раздела]Tag	Тег раздела.	Базовый тег <code>BaseTag</code> .
[Имя объекта раздела]InTag	Тег в объекте раздела.	Базовый тег в базовом объекте <code>BaseEntityInTag</code> .
Клиентские схемы		
[Имя объекта раздела]Section	Схема раздела.	Базовая схема <code>BaseSectionV2</code> раздела.
[Имя объекта раздела]Page	Схема страницы записи раздела.	Базовая схема <code>BaseModulePageV2</code> страницы записи раздела.

## Добавить пользовательскую колонку в реестр раздела

Добавление пользовательской колонки в реестр раздела значит, что колонка добавляется в схему бизнес-объекта раздела. При добавлении пользовательской колонки в текущем пакете создается схема

замещающей модели представления схемы, которая унаследует колонки базового объекта. В созданную схему будет добавлена новая пользовательская колонка.

**Инструменты**, которые позволяют добавить пользовательскую колонку в схему бизнес-объекта раздела:

- Мастер разделов.
- Creatio IDE.

Чтобы добавить пользовательскую колонку в реестр раздела с использованием **мастера разделов**, воспользуйтесь инструкцией, которая приведена в статье [Реестр раздела](#).

Чтобы добавить пользовательскую колонку на страницу раздела с использованием **Creatio IDE**:

1. Создайте схему замещающей модели представления страницы раздела, на которой будет размещена пользовательская колонка. Для этого воспользуйтесь инструкцией, которая приведена в статье [Разработка конфигурационных элементов](#).
2. В качестве родительского объекта выберите схему модели представления, которую необходимо заместить.
3. В схему замещающей модели представления страницы раздела добавьте пользовательскую колонку соответствующего типа.
4. Настройте отображение пользовательской колонки в реестре раздела.

## Настроить условия отображения реестра раздела

Creatio позволяет настроить условия отображения реестра раздела. Записи реестра, для которых выполняются заданные условия, отображаются в соответствии с установленными стилями.

Чтобы **настроить условия отображения реестра раздела**:

1. Создайте схему замещающей модели представления страницы раздела. Для этого воспользуйтесь инструкцией, которая приведена в статье [Разработка конфигурационных элементов](#).
2. В качестве родительского объекта выберите схему модели представления, которую необходимо заместить.
3. Настройте **условия отображения реестра раздела**. Для этого в свойстве `methods` реализуйте метод `prepareResponseCollectionItem()`, который переопределяет базовый метод и модифицирует строку данных перед загрузкой в реестр. В методе реализуйте присвоение определенного значения свойству `customStyle`, которое отвечает за отображение строки реестра. Является объектом, который содержит свойства-аналоги CSS-свойств и формирует стиль отображения необходимых записей реестра.

Пример настройки свойства `customStyle` представлен ниже.

### Пример настройки свойства `customStyle`

```
item.customStyle = {
  /* Цвет текста – белый. */
  "color": "white",
  /* Цвет фона – оранжевый. */
```



```
"background": "orange"
};
```

## Добавить действие для записей раздела

Приложение предоставляет возможность добавить пользовательские действия как для единичной записи, так и для нескольких записей. Базовое наполнение выпадающего меню [ Действия ] ([ Actions ]) страницы раздела реализовано в схеме `BaseSectionV2` пакета `NUI`.

Чтобы **добавить действие для единичной записи раздела**:

1. Создайте схему замещающей модели представления страницы раздела. Для этого воспользуйтесь инструкцией, которая приведена в статье [Разработка конфигурационных элементов](#).
2. В качестве родительского объекта выберите схему модели представления, которую необходимо заместить.
3. Перечень действий раздела является экземпляром класса `Terrasoft.BaseViewModelCollection`. Каждый элемент перечня действий — модель представления. Настройка действия выполняется в конфигурационном объекте, который позволяет как явно задавать свойства модели представления действий, так и использовать базовый механизм привязки.

Настройте **логику работы пункта меню**. Для этого в свойстве `methods` реализуйте метод `getSectionActions()` — переопределенный метод родительской схемы, который возвращает коллекцию действий раздела. При замещении базовых разделов в методе `getSectionActions()` замещающего модуля необходимо вызвать родительскую реализацию этого метода для инициализации действий родительского раздела. Для этого выполните метод `this.callParent(arguments)`, который возвращает коллекцию действий базового раздела.

Добавление действия в коллекцию выполняется с помощью вызова метода `addItem()`. В качестве параметра методу передается callback-метод `getButtonItem()`. Метод создает экземпляр модели представления действия по конфигурационному объекту, который передан ему в качестве параметра.

Обращение к выделенной записи выполняется через атрибут `ActiveRow` модели представления раздела. Атрибут возвращает значение первичной колонки выделенной записи. В свою очередь, значение первичной колонки выделенной записи может использоваться для получения значений, загруженных в реестр полей выбранного объекта.

Значение первичных колонок выбранных записей хранится в свойстве `SelectedRows` модели представления раздела. Эти значения могут использоваться для получения значений, загруженных в реестр полей выбранных объектов, например, из коллекции данных списочного реестра, которая хранится в свойстве `GridData` модели представления реестра.

Базовая реализация добавления действия представлена ниже.

**BaseSectionV2**

```
/**
 * Возвращает коллекцию действий раздела в режиме отображения реестра.
 * @protected
 * @virtual
```

```

* @return {Terrasoft.BaseViewModelCollection} Коллекция действий раздела.
*/
getSectionActions: function() {
    /* Коллекция действий. Экземпляр Terrasoft.BaseViewModelCollection. */
    var actionMenuItems = this.Ext.create("Terrasoft.BaseViewModelCollection");
    /* Добавление действия в коллекцию. В качестве callback-метода передается метод, который инс
    actionMenuItems.addItem(
        this.getButtonMenuItem({
            /* Конфигурационный объект настройки действия. */
            ...
        })
    );
    return actionMenuItems;
}

```

Свойства конфигурационного объекта действия раздела, который передается в качестве параметра в метод `getButtonMenuItem()` представлены в таблице ниже.

Свойства конфигурационного объекта

Свойство	Описание
Type	Тип элемента выпадающего меню [ Действия ] ([ Actions ]). Это свойство позволяет добавить в меню действий горизонтальную линию для отделения блоков меню. Для этого в качестве значения свойства необходимо указать <code>Terrasoft.MenuSeparator</code> . Если значение свойства не указано, то по умолчанию добавляется пункт меню.
Caption	Заголовок пункта меню [ Действия ] ([ Actions ]). Для заголовков рекомендуется использовать <a href="#">локализуемые строки</a> .
Click	Привязка метода-обработчика действия по имени метода.
Enabled	Логическое свойство, которое регулирует доступность пункта меню.
Visible	Логическое свойство, которое регулирует видимость пункта меню.

## Настроить блок быстрых фильтров раздела

Приложение предоставляет возможность реализовать пользовательский блок быстрых фильтров для записей реестра раздела.

Чтобы **настроить блок быстрых фильтров раздела**:

1. Создайте схему замещающей модели представления страницы раздела. Для этого воспользуйтесь инструкцией, которая приведена в статье [Разработка конфигурационных элементов](#).
2. В качестве родительского объекта выберите схему модели представления, которую необходимо заместить.

3. Реализуйте **логику работы фильтрации**. Для этого в свойстве `methods` реализуйте метод `initFixedFiltersConfig()`, который инициализирует фиксированные фильтры. В методе создайте конфигурационный объект с массивом фильтров, присвойте ссылку на объект атрибуту `fixedFiltersConfig` модели представления.

## Удалить раздел

1. Проверьте наличие доступа к конфигурации системы и базе данных.
2. Снимите блокировку с соответствующего удаляемому разделу файла в SVN-хранилище.
3. Удалите раздел, созданный на основании объекта, который планируется удалить. Для этого выполните SQL-запрос.

### SQL-запрос

```
DECLARE @UId UNIQUEIDENTIFIER
DECLARE @ModuleEntityUID UNIQUEIDENTIFIER;
DECLARE @ModuleID UNIQUEIDENTIFIER;
DECLARE @Name NVARCHAR(max) = 'ToDelete';
select @UId = UId from SysSchema where Name Like @Name
select @ModuleEntityUID = Id from SysModuleEntity where
SysEntitySchemaUId = @UId
select @ModuleID = Id from SysModule where SysModuleEntityId = @ModuleEntityUID;
delete from SysModuleInWorkplace where SysModuleId = @ModuleID;
delete from SysModule where Id = @ModuleID;
delete from SysModuleEdit where SysModuleEntityId = @ModuleEntityUID;
delete from SysModuleEntity where Id = @ModuleEntityUID;
delete from SysDetail where EntitySchemaUId = @UId;
delete from SysLookup where SysEntitySchemaUId = @UId;
delete from [Lookup] where SysEntitySchemaUId = @UId;
```

Значение `ToDelete` замените на название схемы пользовательского раздела.

4. После очистки базы данных удалите пользовательские схемы в разделе [ *Конфигурация* ] ([ *Configuration* ]).

**Порядок** удаления конфигурационных схем:

- a. `ToDeleteFile` .
- b. `ToDeleteInFolder` .
- c. `ToDeleteInTag` .
- d. `ToDeleteTag` .
- e. `ToDeleteFolder` .
- f. `ToDelete` .

# Добавить в раздел действие для единичной записи

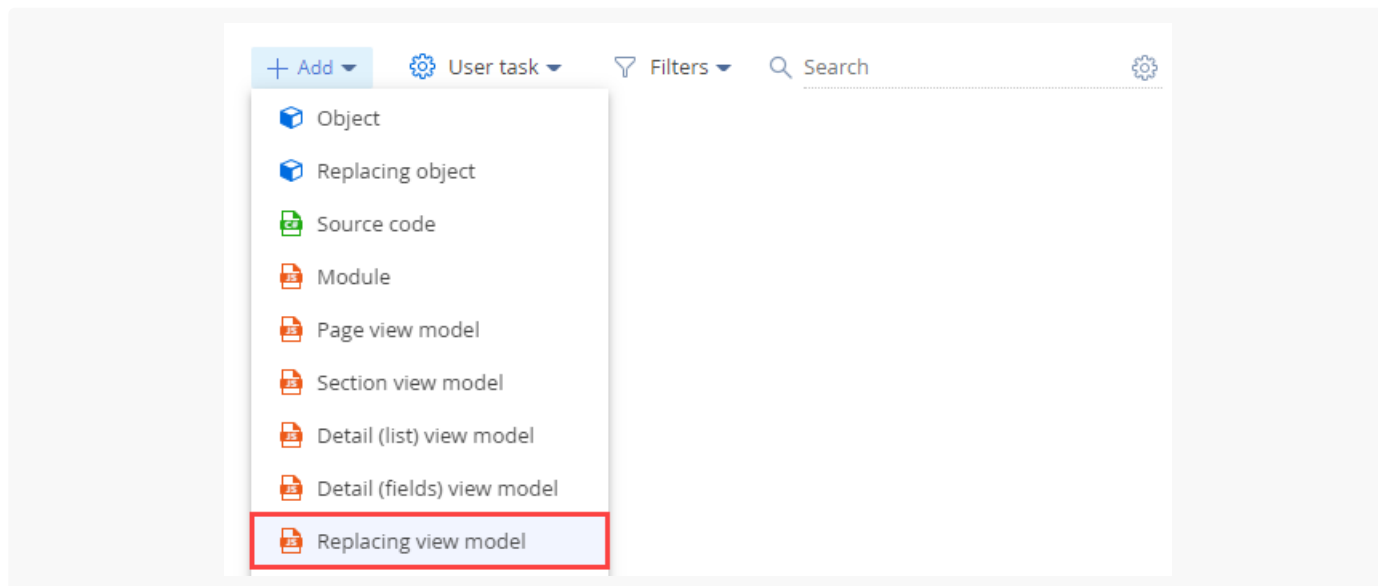
 Средний

Пример реализован для продуктов линейки Sales Creatio.

**Пример.** Для записи реестра раздела [ Заказы ] ([ Orders ]) добавить действие, которое в информационном окне отображает дату создания заказа. Действие активно для заказов, которые находятся на стадии [ Исполнение ] ([ In progress ]).

## Создать схему замещающей модели представления раздела

1. [Перейдите в раздел \[ Конфигурация \]](#) ([ Configuration ]) и выберите пользовательский [пакет](#), в который будет добавлена схема.
2. На панели инструментов реестра раздела нажмите [ Добавить ] —> [ Замещающая модель представления ] ([ Add ] —> [ Replacing view model ]).



3. Заполните **свойства схемы**.

- [ Код ] ([ Code ]) — "OrderSectionV2".
- [ Заголовок ] ([ Title ]) — "Раздел заказа" ("Order section").
- [ Родительский объект ] ([ Parent object ]) — выберите "OrderSectionV2".

**Module** [X]

Code  
OrderSectionV2

Title \*  
OrderSectionV2 [X]

Parent object \*  
OrderSectionV2 (OrderSectionV2) [v]

Package  
sdkAddSectionActionForSingleRowPackage

Description [X]

CANCEL APPLY

4. Добавьте **локализуемую строку** с текстом пункта меню, который планируется добавить.
  - a. В контекстном меню узла [ *Локализуемые строки* ] ([ *Localizable strings* ]) нажмите кнопку **+**.
  - b. Заполните **свойства локализуемой строки**.
    - [ *Код* ] ([ *Code* ]) — "CreationDateActionCaption".
    - [ *Значение* ] ([ *Value* ]) — "Показать дату создания" ("Show creation date").

**Localizable Strings** [X]

Code \*  
CreationDateActionCaption

Value  
Show creation date [X]

CANCEL APPLY

- e. Для добавления локализуемой строки нажмите [ *Добавить* ] ([ *Add* ]).
5. Реализуйте **логику работы пункта меню**. Для этого в свойстве `methods` реализуйте **методы**:
  - `isRunning()` — проверяет, находится ли выбранный в реестре заказ на стадии [ *Исполнение* ] ([ *In progress* ]).
  - `isCustomActionEnabled()` — определяет доступность добавленного пункта меню.
  - `showOrderInfo()` — метод-обработчик действия, который отображает в информационном окне дату создания выбранного заказа.
  - `getSectionActions()` — переопределенный метод родительской схемы, который возвращает коллекцию действий раздела.

Обращение к выделенной записи выполняется через атрибут `ActiveRow` модели представления раздела. Атрибут возвращает значение первичной колонки выделенной записи. В свою очередь, значение первичной колонки выделенной записи может использоваться для получения значений, загруженных в реестр полей выбранного объекта.

Исходный код схемы замещающей модели представления раздела представлен ниже.

#### OrderSectionV2

```
define("OrderSectionV2", ["OrderConfigurationConstants"], function(OrderConfigurationConstant
    return {
        /* Название схемы объекта раздела. */
        entitySchemaName: "Order",
        /* Методы модели представления раздела. */
        methods: {
            /* Проверяет стадию заказа. activeRowId – значение первичной колонки выделенной з
            isRunning: function(activeRowId) {
                /* Получает коллекцию данных списочного представления реестра раздела. */
                var gridData = this.get("GridData");
                /* Получает модель выбранного заказа по заданному значению первичной колонки.
                var selectedOrder = gridData.get(activeRowId);
                /* Получает свойства модели – статуса выбранного заказа. */
                var selectedOrderStatus = selectedOrder.get("Status");
                /* Метод возвращает true, если статус заказа [Исполнение], иначе возвращает f
                return selectedOrderStatus.value === OrderConfigurationConstants.Order.Orders
            },
            /* Определяет доступность пункта меню. */
            isCustomActionEnabled: function() {
                /* Попытка получения идентификатора активной записи. */
                var activeRowId = this.get("ActiveRow");
                /* Если идентификатор определен и статус выбранного заказа [Исполнение], то в
                return activeRowId ? this.isRunning(activeRowId) : false;
            },
            /* Метод-обработчик действия. Отображает в информационном окне дату создания зака
            showOrderInfo: function() {
                var activeRowId = this.get("ActiveRow");
                var gridData = this.get("GridData");
                /* Получает дату создания заказа. Колонка должна быть добавлена в реестр. */
                var dueDate = gridData.get(activeRowId).get("Date");
                /* Отображает информационное окно. */
                this.showInformationDialog(dueDate);
            },
            /* Переопределение базового виртуального метода, который возвращает коллекцию дей
            getSectionActions: function() {
                /* Вызывается родительская реализация метода для получения коллекции проиници
                var actionMenuItems = this.callParent(arguments);
                /* Добавляет линию-разделитель. */
                actionMenuItems.addItem(this.getButtonItem({
```

```

        Type: "Terrasoft.MenuSeparator",
        Caption: ""
    }));
    /* Добавляет пункт меню в список действий раздела. */
    actionMenuItems.addItem(this.getButtonMenuItem({
        /* Привязка заголовка пункта меню к локализуемой строке схемы. */
        "Caption": {bindTo: "Resources.Strings.CreationDateActionCaption"},
        /* Привязка метода-обработчика действия. */
        "Click": {bindTo: "showOrderInfo"},
        /* Привязка свойства доступности пункта меню к значению, которое возвраща
        "Enabled": {bindTo: "isCustomActionEnabled"}
    }));
    /* Возврат дополненной коллекции действий раздела. */
    return actionMenuItems;
}
}
};
});

```

6. На панели инструментов дизайнера нажмите [ Сохранить ] ([ Save ]).

## Результат выполнения примера

Чтобы **посмотреть результат выполнения примера**:

1. Очистите кэш браузера.
2. Обновите страницу раздела [ Заказы ] ([ Orders ]).

В результате выполнения примера на страницу заказа добавлено действие [ Показать дату создания ] ([ Show creation date ]).

Если заказ находится на стадии [ Исполнение ] ([ In progress ]), то действие [ Показать дату создания ] ([ Show creation date ]) активно.

Orders

NEW ORDER ACTIONS

What can I do for you? >

7.18.5.1500

VIEW

Owner Filters/folders Tag

Account	Contact	Owner
Apex Solutions	Andrew Wayne	Marina Kysla
Apex Solutions	Andrew Wayne	Symon Clarke
Streamline Development	Bruce Clayton	Marina Kysla
Gateway	James Smith	Marina Kysla
Infocom	Andrew Z. Barber	Symon Clarke
Alpha Business	Jordan Anderson	Marina Kysla
Factorial Services	Taylor P. Johnson	Symon Clarke

OPEN COPY DELETE PRINT

В результате выбора действия [ Показать дату создания ] ([ Show creation date ]), в информационном окне отображается дата создания заказа.

Tue Nov 09 2021 04:00:00 GMT+0200 (Eastern European Standard Time)

OK

Если заказ не находится на стадии [ Исполнение ] ([ In progress ]), то действие [ Показать дату создания ] ([ Show creation date ]) неактивно.



Orders

What can I do for you?

7.18.5.1500

VIEW

NEW ORDER ACTIONS

Select multiple records

Select all

Export to Excel

Data import

Change log setup

Send for approval

Show creation date

Owner Filters/folders Tag

Account	Contact	Owner
Apex Solutions	Andrew Wayne	Marina Kysla
Apex Solutions	Andrew Wayne	Symon Clarke
Streamline Development	Bruce Clayton	Marina Kysla
Gateway	James Smith	Marina Kysla
Infocom	Andrew Z. Barber	Symon Clarke
Alpha Business	Jordan Anderson	Marina Kysla
Factorial Services	Taylor P. Johnson	Symon Clarke
Alpha Business		Mary King

OPEN COPY DELETE PRINT

## Добавить в раздел действие для нескольких записей

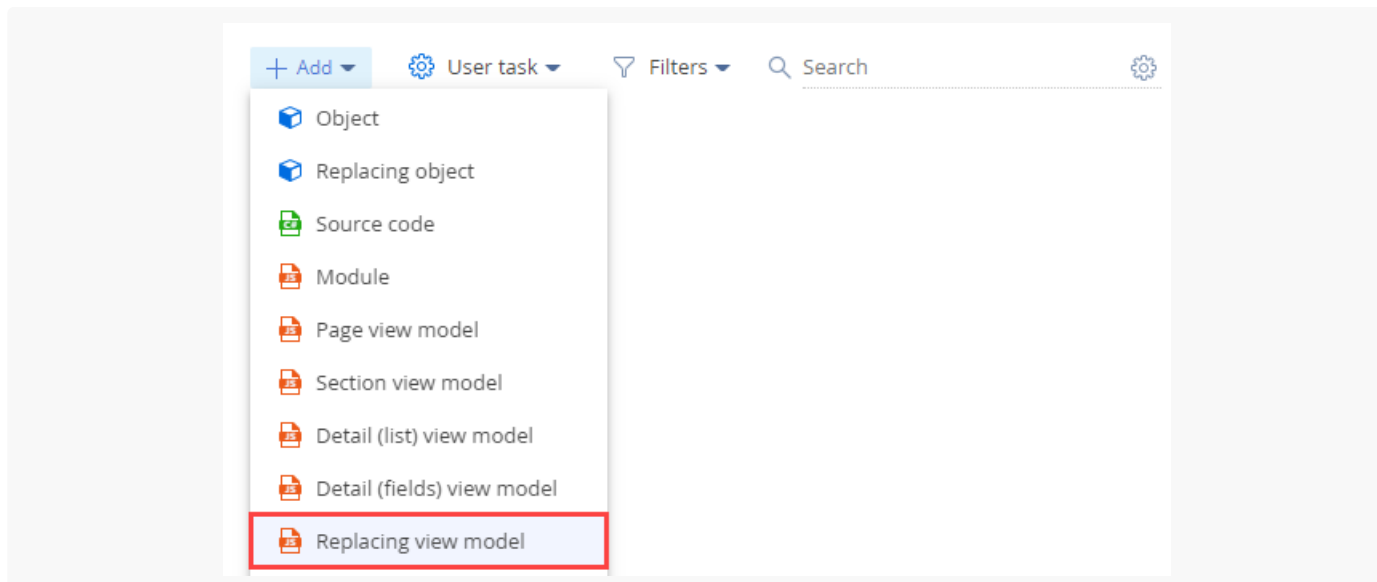
Средний

Пример реализован для продуктов линейки Sales Creatio.

**Пример.** Для записей реестра раздела [ *Заказы* ] ([ *Orders* ]) добавить действие, которое в информационном окне отображает перечень контрагентов выбранных заказов.

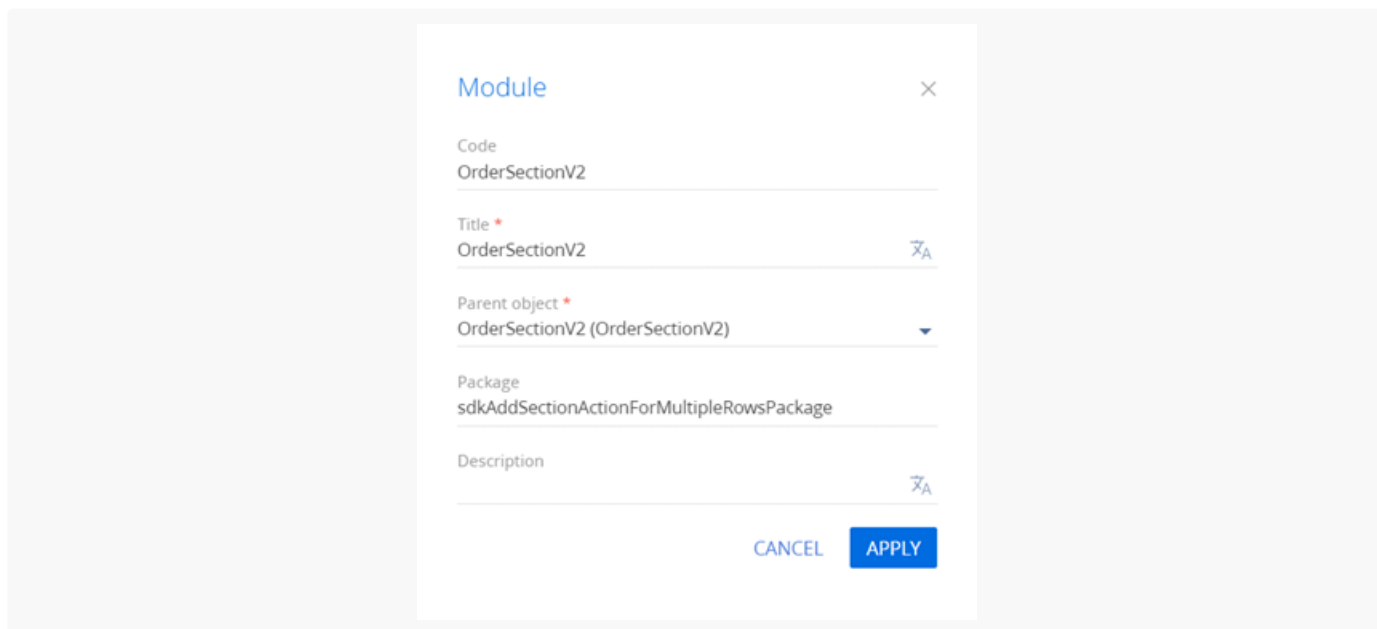
## Создать схему замещающей модели представления раздела

1. [Перейдите в раздел \[ Конфигурация \]](#) ([ *Configuration* ]) и выберите пользовательский [пакет](#), в который будет добавлена схема.
2. На панели инструментов реестра раздела нажмите [ *Добавить* ] —> [ *Замещающая модель представления* ] ([ *Add* ] —> [ *Replacing view model* ]).



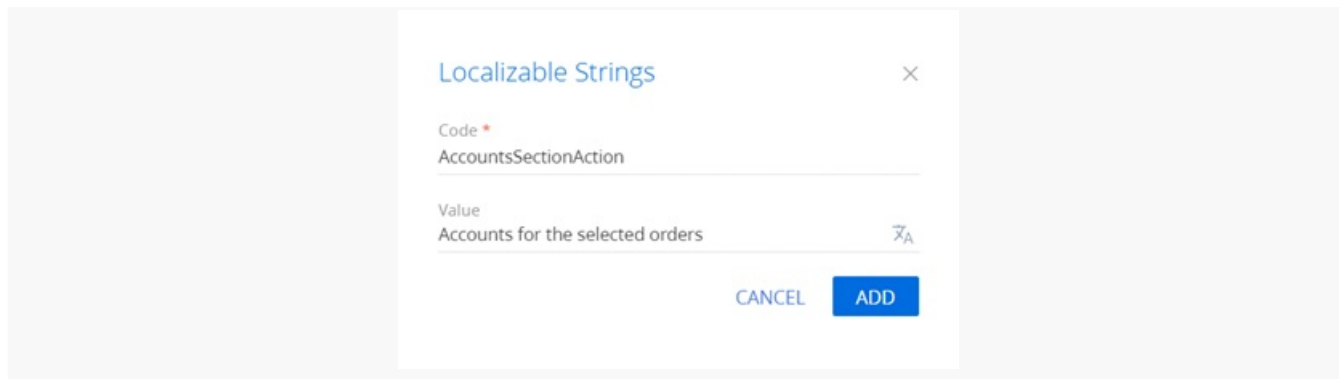
### 3. Заполните **свойства схемы**.

- [ Код ] ([ Code ]) — "OrderSectionV2".
- [ Заголовок ] ([ Title ]) — "Раздел заказа" ("Order section").
- [ Родительский объект ] ([ Parent object ]) — выберите "OrderSectionV2".



### 4. Добавьте **локализуемую строку** с текстом пункта меню, который планируется добавить.

- В контекстном меню узла [ Локализуемые строки ] ([ Localizable strings ]) нажмите кнопку **+**.
- Заполните **свойства локализуемой строки**.
  - [ Код ] ([ Code ]) — "AccountsSectionAction".
  - [ Значение ] ([ Value ]) — "Отобразить контрагентов по выбранным заказам" ("Accounts for the selected orders").



е. Для добавления локализуемой строки нажмите [ *Добавить* ] ([ *Add* ]).

5. Реализуйте **логику работы пункта меню**. Для этого в свойстве `methods` реализуйте **методы**:

- `isCustomActionEnabled()` — определяет доступность добавленного пункта меню.
- `showOrdersInfo()` — метод-обработчик действия, который отображает в информационном окне перечень контрагентов выбранных заказов.
- `getSectionActions()` — переопределенный метод родительской схемы, который возвращает коллекцию действий раздела.

Значение первичных колонок выбранных записей хранится в свойстве `SelectedRows` модели представления раздела. Эти значения могут использоваться для получения значений, загруженных в реестр полей выбранных объектов, например, из коллекции данных списочного реестра, которая хранится в свойстве `GridData` модели представления реестра.

Исходный код схемы замещающей модели представления раздела представлен ниже.

#### OrderSectionV2

```
define("OrderSectionV2", ["OrderConfigurationConstants"], function(OrderConfigurationConstant
    return {
        /* Название схемы объекта раздела. */
        entitySchemaName: "Order",
        /* Методы модели представления раздела. */
        methods: {
            /* Определяет доступность пункта меню. */
            isCustomActionEnabled: function() {
                /* Попытка получить массив идентификаторов выбранных записей. */
                var selectedRows = this.get("SelectedRows");
                /* Если массив содержит элементы (выбрана хотя бы одна запись в реестре), то
                return selectedRows ? (selectedRows.length > 0) : false;
            },
            /* Метод-обработчик действия. Отображает в информационном окне перечень контраген
            showOrdersInfo: function() {
                /* Получает массив идентификаторов выбранных записей. */
                var selectedRows = this.get("SelectedRows");
                /* Получает коллекцию данных записей реестра. */
                var gridData = this.get("GridData");
```

```

/* Переменная для хранения модели объекта выбранного заказа. */
var selectedOrder = null;
/* Переменная для хранения названия контрагента выбранного заказа. */
var selectedOrderAccount = "";
/* Переменная для формирования текста информационного окна. */
var infoText = "";
/* Обработка массива идентификаторов выбранных записей реестра. */
selectedRows.forEach(function(selectedRowId) {
    /* Получает модель объекта выбранного заказа. */
    selectedOrder = gridData.get(selectedRowId);
    /* Получает название контрагента выбранного заказа. Колонка должна быть д
    selectedOrderAccount = selectedOrder.get("Account").displayValue;
    /* Добавляет название контрагента в текст информационного окна. */
    infoText += "\n" + selectedOrderAccount;
});
/* Отображает информационное окно. */
this.showInformationDialog(infoText);
},
/* Переопределение базового виртуального метода, который возвращает коллекцию дей
getSectionActions: function() {
    /* Вызывается родительская реализация метода для получения коллекции проиници
    var actionMenuItems = this.callParent(arguments);
    /* Добавляет линию-разделитель. */
    actionMenuItems.addItem(this.getButtonMenuItem({
        Type: "Terrasoft.MenuSeparator",
        Caption: ""
    }));
    /* Добавляет пункт меню в список действий раздела. */
    actionMenuItems.addItem(this.getButtonMenuItem({
        /* Привязка заголовка пункта меню к локализуемой строке схемы. */
        "Caption": {bindTo: "Resources.Strings.AccountsSectionAction"},
        /* Привязка метода-обработчика действия. */
        "Click": {bindTo: "showOrdersInfo"},
        /* Привязка свойства доступности пункта меню к значению, которое возвраща
        "Enabled": {bindTo: "isCustomActionEnabled"},
        /* Поддержка режима множественного выбора. */
        "IsEnabledForSelectedAll": true
    }));
    /* Возврат дополненной коллекции действий раздела. */
    return actionMenuItems;
}
}
};
});

```

6. На панели инструментов дизайнера нажмите [ Сохранить ] ([ Save ]).

## Результат выполнения примера

Чтобы **посмотреть результат выполнения примера**:

1. Очистите кэш браузера.
2. Обновите страницу раздела [ *Заказы* ] ([ *Orders* ]).

По умолчанию для реестра используется режим выбора одной записи. Для выбора в реестре нескольких записей в меню кнопки [ *Действия* ] ([ *Actions* ]) нажмите [ *Выбрать несколько записей* ] ([ *Select multiple records* ]). После этого изменяется визуальное представление реестра — появляются элементы для выбора записей.

В результате выполнения примера на страницу заказа добавлено действие [ *Отобразить контрагентов по выбранным заказам* ] ([ *Accounts for the selected orders* ]).

The screenshot shows the 'Orders' page in the Creatio system. The 'ACTIONS (3)' menu is open, displaying various actions. The action 'Accounts for the selected orders' is highlighted with a red box. Below the menu, a table of orders is visible, with several rows selected (checked).

Order ID	Date	Status	Account	Contact	Owner
ORD-29	11/9/2021 12:00 AM	In progress	Apex Solutions	Andrew Wayne	Marina Kysla
ORD-25	11/7/2021 4:00 AM	Completed	Apex Solutions	Andrew Wayne	Symon Clarke
ORD-35	11/6/2021 12:00 AM	Completed	Streamline Development	Bruce Clayton	Marina Kysla
ORD-24	10/31/2021 4:00 AM	In progress	Gateway	James Smith	Marina Kysla
ORD-23	10/27/2021 4:00 AM	Completed	Infocom	Andrew Z. Barber	Symon Clarke
ORD-34	10/24/2021 1:00 AM	In progress	Alpha Business	Jordan Anderson	Marina Kysla
		In progress	Factorial Services	Taylor P. Johnson	Symon Clarke
		Completed	Alpha Business		Mary King
		Completed	Alpha Business	Alexander Wilson	Marina Kysla
		Completed	Infocom	Andrew Z. Barber	Mary King
		Completed	Axiom	Alice Phillips	Marina Kysla
		Completed	Axiom	Alice Phillips	Marina Kysla
		Completed	Build Technologies		Mary King

В результате выбора действия [ *Отобразить контрагентов по выбранным заказам* ] ([ *Accounts for the selected orders* ]), в информационном окне отображается перечень контрагентов выбранных заказов.

Alpha Business  
Axiom  
Build Technologies

OK

Для отмены режима выбора нескольких записей в меню кнопки [ Действия ] ([ Actions ]) нажмите [ Отменить множественный выбор ] ([ Cancel multiple selection ]).

Если в реестре раздела [ Заказы ] ([ Orders ]) не выбрана ни одна запись, то действие [ Отобразить контрагентов по выбранным заказам ] ([ Accounts for the selected orders ]) неактивно.

Orders

NEW ORDER ACTIONS

What can I do for you? >

7.18.5.1500

VIEW

Select multiple records

Select all

Export to Excel

Data import

Change log setup

Send for approval

Accounts for the selected orders

Owner	Filters/folders	Tag
Account	Contact	Owner
ORD-30	11/18/2	ORD-30
ORD-31	11/17/2	ORD-31
ORD-26	11/14/2	ORD-26
ORD-28	11/13/2	ORD-28
ORD-33	11/13/2	ORD-33

## Настроить обработку действия для нескольких записей раздела

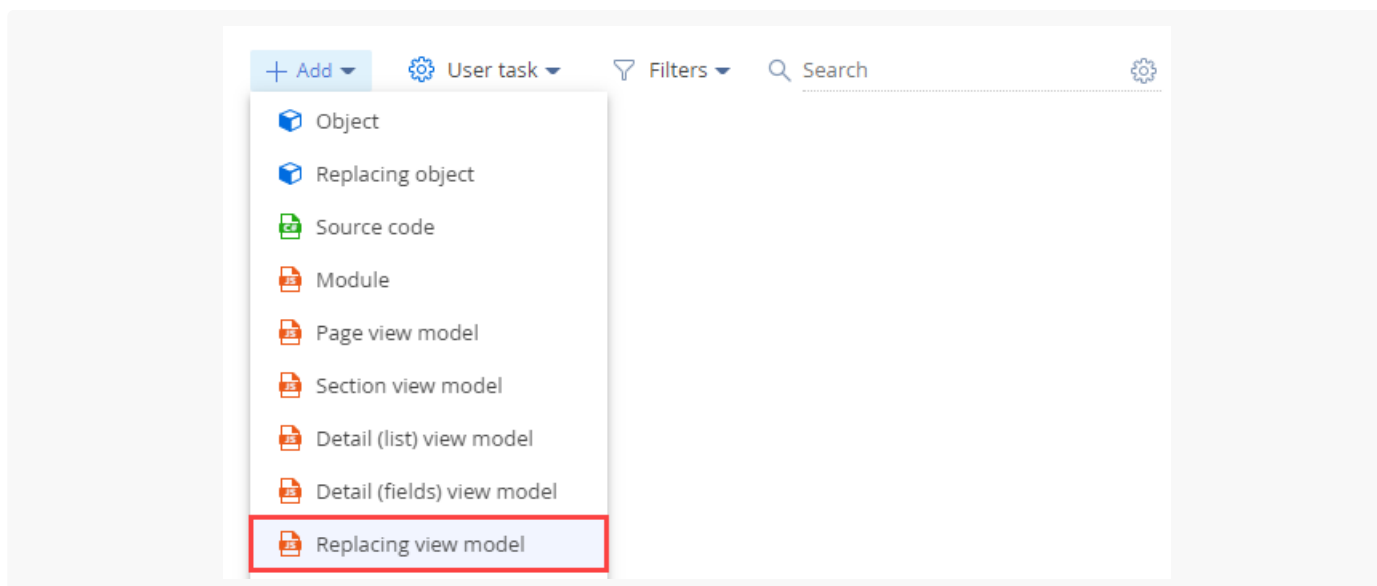
Средний

**Пример.** Для реестра раздела [ Активности ] ([ Activities ]) настроить обработку действия. Действие устанавливает состояние [ Завершена ] ([ Completed ]) для выбранных активностей.

## Создать схему замещающей модели представления раздела

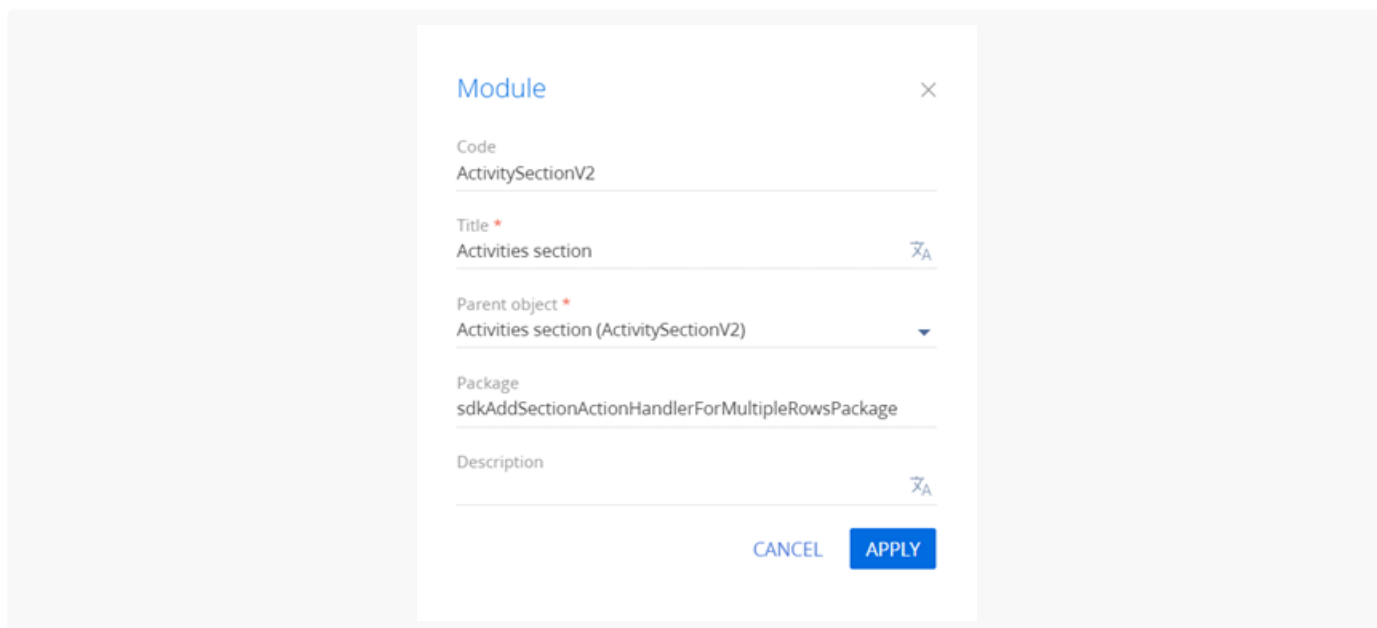
1. [Перейдите в раздел \[ Конфигурация \]](#) ([ Configuration ]) и выберите пользовательский [пакет](#), в который будет добавлена схема.


2. На панели инструментов реестра раздела нажмите [ *Добавить* ] —> [ *Замещающая модель представления* ] ([ *Add* ] —> [ *Replacing view model* ]).



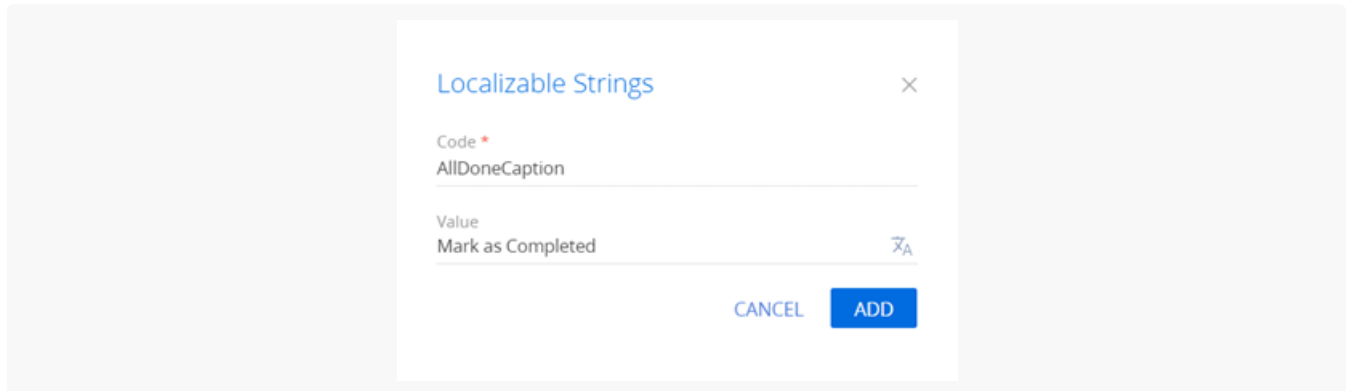
3. Заполните **свойства схемы**.

- [ *Код* ] ([ *Code* ]) — "ActivitySectionV2".
- [ *Заголовок* ] ([ *Title* ]) — "Раздел активности" ("Activities section").
- [ *Родительский объект* ] ([ *Parent object* ]) — выберите "ActivitySectionV2".



4. Добавьте **локализуемую строку** с текстом пункта меню, который планируется добавить.
- В контекстном меню узла [ *Локализуемые строки* ] ([ *Localizable strings* ]) нажмите кнопку .
  - Заполните **свойства локализуемой строки**.
    - [ *Код* ] ([ *Code* ]) — "AllDoneCaption".

- [ *Значение* ] ([ *Value* ]) — "Отметить как "Завершены"" ("Mark as Completed").



e. Для добавления локализуемой строки нажмите [ *Добавить* ] ([ *Add* ]).

5. Реализуйте **логику работы пункта меню**. Для этого в свойстве `methods` реализуйте **методы**:

- `isCustomActionEnabled()` — определяет доступность добавленного пункта меню.
- `setAllDone()` — метод-обработчик действия, который устанавливает состояние [ *Завершена* ] ([ *Completed* ]) для выбранных активностей.
- `getSectionActions()` — переопределенный метод родительской схемы, который возвращает коллекцию действий раздела.

Исходный код схемы замещающей модели представления раздела представлен ниже.

#### ActivitySectionV2

```
define("ActivitySectionV2", ["ConfigurationConstants"], function(ConfigurationConstants) {
    return {
        /* Название схемы объекта раздела. */
        entitySchemaName: "Activity",
        /* Методы модели представления раздела. */
        methods: {
            /* Определяет доступность пункта меню. */
            isCustomActionEnabled: function() {
                /* Попытка получить массив идентификаторов выбранных записей. */
                var selectedRows = this.get("SelectedRows");
                /* Если массив содержит элементы (выбрана хотя бы одна запись в реестре), то
                return selectedRows ? (selectedRows.length > 0) : false;
            },
            /* Метод-обработчик действия. Устанавливает для выбранных записей статус [Выполне
            setAllDone: function() {
                /* Получает массив идентификаторов выбранных записей. */
                var selectedRows = this.get("SelectedRows");
                /* Обработка запускается в случае, если выбрана хотя бы одна запись. */
                if (selectedRows.length > 0) {
                    /* Создает экземпляр класса пакетных запросов. */
                    var batchQuery = this.Ext.create("Terrasoft.BatchQuery");
```



```

        /* Обновляет каждую из выбранных записей. */
        selectedRows.forEach(function(selectedRowId) {
            /* Создает экземпляр класса UpdateQuery с корневой схемой Activity. */
            var update = this.Ext.create("Terrasoft.UpdateQuery", {
                rootSchemaName: "Activity"
            });
            /* Применяет фильтр для определения записи для обновления. */
            update.enablePrimaryColumnFilter(selectedRowId);
            /* Для колонки [Status] устанавливается значение "Завершена" с помощью */
            update.setParameterValue("Status", ConfigurationConstants.Activity.St
            /* Добавляет запрос на обновление записи в пакетный запрос. */
            batchQuery.add(update);
        }, this);
        /* Выполняет пакетный запрос к серверу. */
        batchQuery.execute(function() {
            /* Обновляет реестр. */
            this.reloadGridData();
        }, this);
    }
},
/* Переопределение базового виртуального метода, который возвращает коллекцию дей
getSectionActions: function() {
    /* Вызывается родительская реализация метода для получения коллекции проиници
    var actionMenuItems = this.callParent(arguments);
    /* Добавляет линию-разделитель. */
    actionMenuItems.addItem(this.getButtonMenuItem({
        Type: "Terrasoft.MenuSeparator",
        Caption: ""
    }));
    /* Добавляет пункт меню в список действий раздела. */
    actionMenuItems.addItem(this.getButtonMenuItem({
        /* Привязка заголовка пункта меню к локализуемой строке схемы. */
        "Caption": { bindTo: "Resources.Strings.AllDoneCaption" },
        /* Привязка метода-обработчика действия. */
        "Click": { bindTo: "setAllDone" },
        /* Привязка свойства доступности пункта меню к значению, которое возвраща
        "Enabled": { bindTo: "isCustomActionEnabled" },
        /* Поддержка режима множественного выбора. */
        "IsEnabledForSelectedAll": true
    }));
    /* Возврат дополненной коллекции действий раздела. */
    return actionMenuItems;
}
}
};
});

```

6. На панели инструментов дизайнера нажмите [ *Сохранить* ] ([ *Save* ]).

## Результат выполнения примера

Чтобы **посмотреть результат выполнения примера**:

1. Очистите кэш браузера.
2. Обновите страницу раздела [ *Активности* ] ([ *Activities* ]).

По умолчанию для реестра используется режим выбора одной записи. Для выбора в реестре нескольких записей в меню кнопки [ *Действия* ] ([ *Actions* ]) нажмите [ *Выбрать несколько записей* ] ([ *Select multiple records* ]). После этого изменяется визуальное представление реестра — появляются элементы для выбора записей.

В результате выполнения примера на страницу активности добавлено действие [ *Отметить как "Завершены"* ] ([ *Mark as Completed* ]).

The screenshot shows the 'Activities' page in the Creatio interface. At the top, there are icons for calendar, list, and bar chart, along with a search bar and the Creatio logo. A green button labeled 'ACTIONS (3)' has a dropdown menu open, listing various actions. The 'Mark as Completed' action is highlighted with a red rectangular box. Below the menu, a table displays a list of activities. The table has columns for Category, Assignee, Due date, Account, and Status. The last three rows of the table are selected (highlighted in blue), and their status is 'Completed'.

Category	Assignee	Due	Account	Status
Meeting	Valerie E. Murphy	11/28/2021 2:00 PM		Not started
Call	Marina Kysla	11/24/2021 9:45 AM	Alpha Business	Not started
Call	Marina Kysla	11/25/2021 10:00 AM	Alpha Business	Not started
Meeting	Peter Moore	12/16/2021 3:40 PM	Console Solutions	Not started
To do	Marina Kysla	11/25/2021 7:15 AM		Not started
Call client, confirm meeting time	Valerie E. Murphy	11/21/2021 9:30 AM	Build Technologies	Completed
Web-form optimization for Firefox	Marina Kysla	9/26/2021 4:00 PM		Completed
Testing the hardware (System Patriot Tower E3-1220V3)	William Walker	9/16/2021 4:15 PM		Completed

В результате выбора действия [ *Отметить как "Завершены"* ] ([ *Mark as Completed* ]) для выбранных активностей в колонке [ *Статус* ] ([ *Status* ]) установлено значение [ *Завершена* ] ([ *Completed* ]).

<input type="checkbox"/>	Work with emails, calendar	To do	Marina Kysla	11/25/2021 7:15 AM		Not started
<input type="checkbox"/>	Call client, confirm meeting time	Call	Valerie E. Murphy	11/21/2021 9:30 AM	Build Technologies	Completed
<input type="checkbox"/>	Web-form optimization for Firefox	To do	Marina Kysla	9/26/2021 4:00 PM		Completed
<input type="checkbox"/>	Testing the hardware (System Patriot Tower E3-1220V3)	To do	William Walker	9/16/2021 4:15 PM		Completed
<input type="checkbox"/>	Contact customer, specify need, budget, decision-making role.	To do	Mary King	11/21/2021 11:16 AM		Completed

Для отмены режима выбора нескольких записей в меню кнопки [ Действия ] ([ Actions ]) нажмите [ Отменить множественный выбор ] ([ Cancel multiple selection ]).

Если в реестре раздела [ Активности ] ([ Activities ]) не выбрана ни одна запись, то действие [ Отметить как "Завершены" ] ([ Mark as Completed ]) неактивно.

The screenshot shows the 'Activities' section of the software. The 'ACTIONS' menu is open, displaying several options. The 'Mark as Completed' option is highlighted with a red box. The background shows a list of activities with columns for Assignee, Due, Account, and Status.

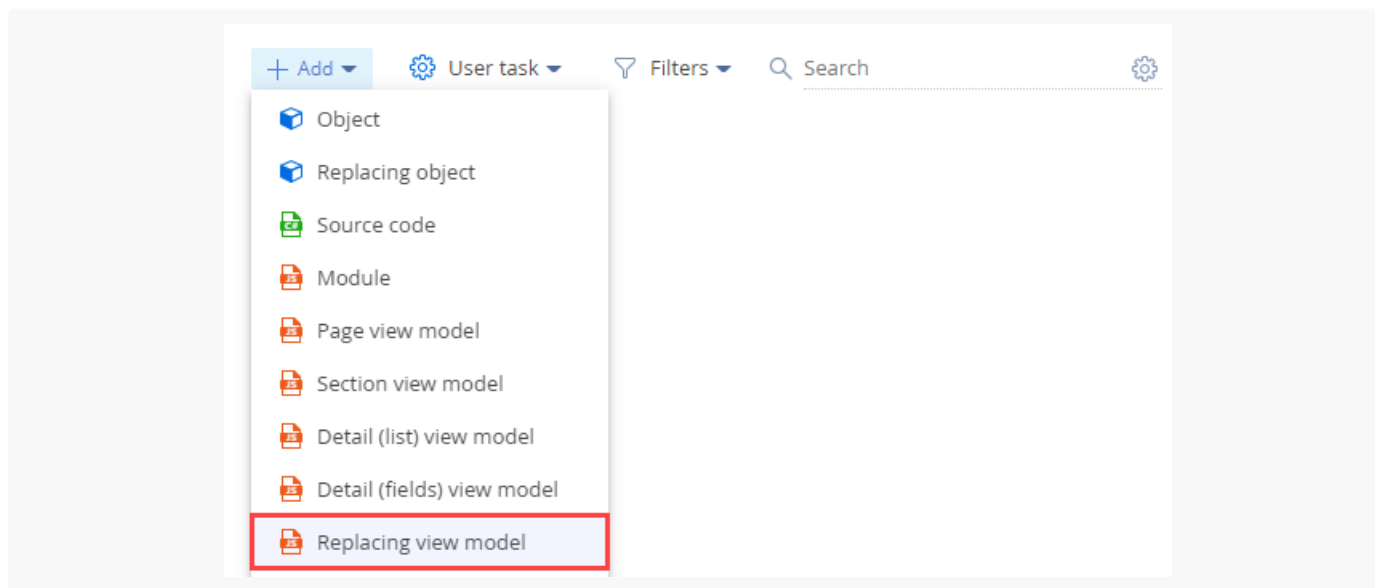
## Настроить обработку действия с использованием справочника для нескольких записей раздела

 Средний

**Пример.** Для реестра раздела [ Активности ] ([ Activities ]) настроить обработку действия. Действие вызывает окно справочника [ Контакты ] ([ Contacts ]). Выбранный контакт устанавливается в качестве ответственного для выбранных активностей.

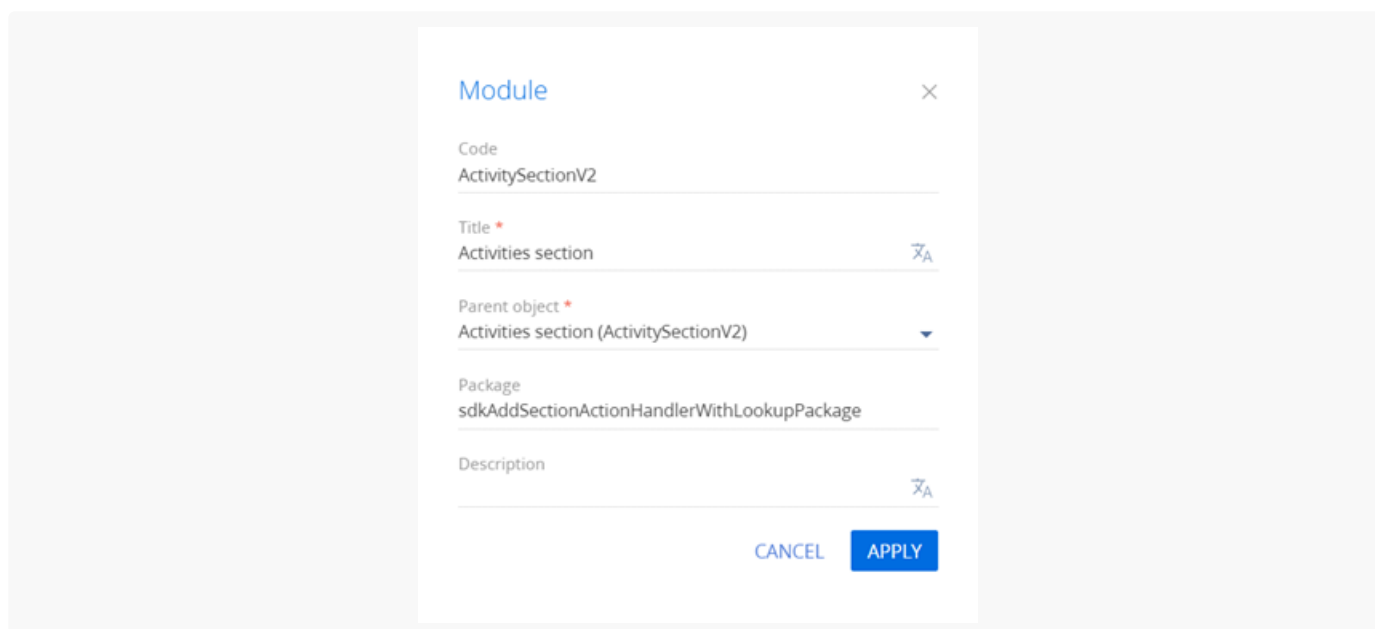
## Создать схему замещающей модели представления раздела


1. [Перейдите в раздел \[ Конфигурация \]](#) ([ *Configuration* ]) и выберите пользовательский [пакет](#), в который будет добавлена схема.
2. На панели инструментов реестра раздела нажмите [ *Добавить* ] —> [ *Замещающая модель представления* ] ([ *Add* ] —> [ *Replacing view model* ]).

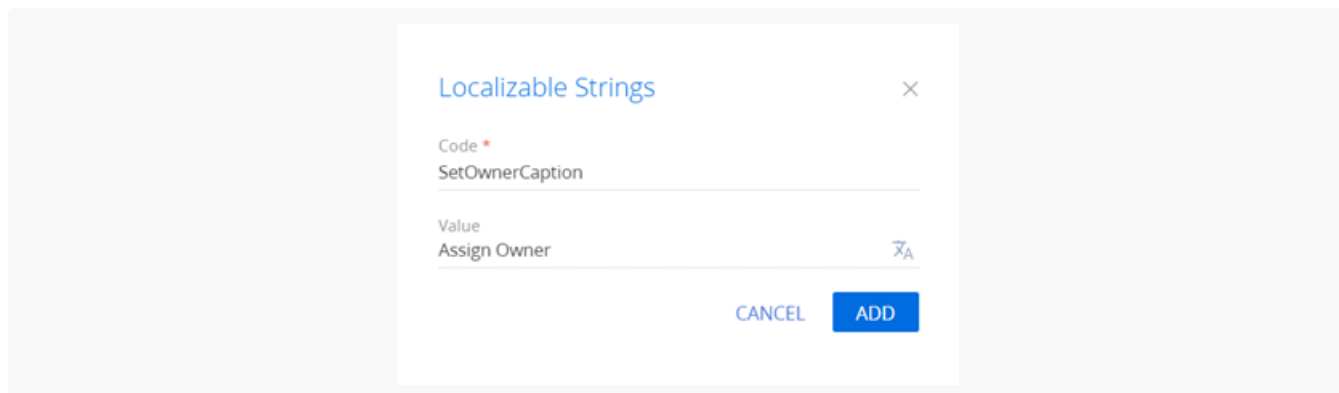


### 3. Заполните **свойства схемы**.

- [ *Код* ] ([ *Code* ]) — "ActivitySectionV2".
- [ *Заголовок* ] ([ *Title* ]) — "Раздел активности" ("Activities section").
- [ *Родительский объект* ] ([ *Parent object* ]) — выберите "ActivitySectionV2".



4. Добавьте **локализуемую строку** с текстом пункта меню, который планируется добавить.
  - a. В контекстном меню узла [ *Локализуемые строки* ] ([ *Localizable strings* ]) нажмите кнопку .
  - b. Заполните **свойства локализуемой строки**.
    - [ *Код* ] ([ *Code* ]) — "SetOwnerCaption".
    - [ *Значение* ] ([ *Value* ]) — "Назначить ответственного" ("Assign Owner").



- e. Для добавления локализуемой строки нажмите [ *Добавить* ] ([ *Add* ]).
5. Реализуйте **логику работы пункта меню**. Для этого в свойстве `methods` реализуйте **методы**:
    - `isCustomActionEnabled()` — определяет доступность добавленного пункта меню.
    - `setOwner()` — метод-обработчик действия, который вызывает открытие справочника [ *Контакты* ] ([ *Contacts* ]).
    - `lookupCallback()` — callback-метод, который устанавливает выбранный в справочнике контакт в качестве ответственного для выбранных активностей.
    - `getSectionActions()` — переопределенный метод родительской схемы, который возвращает коллекцию действий раздела.

Исходный код схемы замещающей модели представления раздела представлен ниже.

#### ActivitySectionV2

```
define("ActivitySectionV2", ["ConfigurationConstants"], function(ConfigurationConstants) {
  return {
    /* Название схемы объекта раздела. */
    entitySchemaName: "Activity",
    /* Методы модели представления раздела. */
    methods: {
      /* Определяет доступность пункта меню. */
      isCustomActionEnabled: function() {
        /* Попытка получить массив идентификаторов выбранных записей. */
        var selectedRows = this.get("SelectedRows");
        /* Если массив содержит элементы (выбрана хотя бы одна запись в реестре), то
        return selectedRows ? (selectedRows.length > 0) : false;
      }
    }
  };
});
```

```

},
/* Метод-обработчик действия. Вызывает открытие справочника [Контакты]. */
setOwner: function() {
    /* Определяет конфигурацию справочника. */
    var config = {
        /* Схема [Contact]. */
        entitySchemaName: "Contact",
        /* Множественный выбор отключен. */
        multiSelect: false,
        /* Отображаемая колонка – [Name]. */
        columns: ["Name"]
    };
    /* Открывает справочник с определенной конфигурацией и callback-функцией, кот
    this.openLookup(config, this.lookupCallback, this);
},
/* Выполняет установку выбранного в справочнике контакта в качестве ответственног
lookupCallback: function(args) {
    /* Идентификатор выбранной из справочника записи. */
    var activeRowId;
    /* Получает выбранные в справочнике записи. */
    var lookupSelectedRows = args.selectedRows.getItems();
    if (lookupSelectedRows && lookupSelectedRows.length > 0) {
        /* Получает Id первой выбранной в справочнике записи. */
        activeRowId = lookupSelectedRows[0].Id;
    }
    /* Получает массив идентификаторов выбранных записей. */
    var selectedRows = this.get("SelectedRows");
    /* Обработка запускается в случае, если выбрана хотя бы одна запись и в справ
    if ((selectedRows.length > 0) && activeRowId) {
        /* Создает экземпляр класса пакетных запросов. */
        var batchQuery = this.Ext.create("Terrasoft.BatchQuery");
        /* Обновляет каждую из выбранных записей. */
        selectedRows.forEach(function(selectedRowId) {
            /* Создает экземпляр класса UpdateQuery с корневой схемой Activity. *
            var update = this.Ext.create("Terrasoft.UpdateQuery", {
                rootSchemaName: "Activity"
            });
            /* Применяет фильтр для определения записи для обновления. */
            update.enablePrimaryColumnFilter(selectedRowId);
            /* Для колонки [Owner] устанавливается значение, равное идентификатор
            update.setParameterValue("Owner", activeRowId, this..Terrasoft.DataVa
            /* Добавляет запрос на обновление записи в пакетный запрос. */
            batchQuery.add(update);
        }, this);
    /* Выполняет пакетный запрос к серверу. */
    batchQuery.execute(function() {
        /* Обновляет реестр. */
        this.reloadGridData();
    }, this);
}

```

```

    }
},
/* Переопределение базового виртуального метода, который возвращает коллекцию дей
getSectionActions: function() {
    /* Вызывается родительская реализация метода для получения коллекции проиници
    var actionMenuItems = this.callParent(arguments);
    /* Добавляет линию-разделитель. */
    actionMenuItems.addItem(this.getButtonItem({
        Type: "Terrasoft.MenuSeparator",
        Caption: ""
    }));
    /* Добавляет пункт меню в список действий раздела. */
    actionMenuItems.addItem(this.getButtonItem({
        /* Привязка заголовка пункта меню к локализуемой строке схемы. */
        "Caption": { bindTo: "Resources.Strings.SetOwnerCaption" },
        /* Привязка метода-обработчика действия. */
        "Click": { bindTo: "setOwner" },
        /* Привязка свойства доступности пункта меню к значению, которое возвраща
        "Enabled": { bindTo: "isCustomActionEnabled" },
        /* Поддержка режима множественного выбора. */
        "IsEnabledForSelectedAll": true
    }));
    /* Возврат дополненной коллекции действий раздела. */
    return actionMenuItems;
}
}
};
});

```

6. На панели инструментов дизайнера нажмите [ *Сохранить* ] ([ *Save* ]).






## Результат выполнения примера


Чтобы **посмотреть результат выполнения примера**:





1. Очистите кэш браузера.
2. Обновите страницу раздела [ *Активности* ] ([ *Activities* ]).







По умолчанию для реестра используется режим выбора одной записи. Для выбора в реестре нескольких записей в меню кнопки [ *Действия* ] ([ *Actions* ]) нажмите [ *Выбрать несколько записей* ] ([ *Select multiple records* ]). После этого изменяется визуальное представление реестра — появляются элементы для выбора записей.

В результате выполнения примера на страницу заказа добавлено действие [ *Назначить ответственного* ] ([ *Assign Owner* ]).

Activities    What can I do for you?  **Creatio**  
7.18.5.1500  
[VIEW](#) 

**ACTIONS (3)** 

- Synchronize activities 
- Cancel multiple selection
- Deselect all
- Select all
-  Export to Excel
-  Merge records
-  Change log setup
- Delete
- Assign Owner**

	<Due date> 	 Employee 	 Filters/folders 	 Tag	
Category	Assignee	Due	Account	Status	
Meeting	Valerie E. Murphy	11/28/2021 2:00 PM		Not started	
Call	Marina Kysla	11/24/2021 9:45 AM	Alpha Business	Not started	
Call	Marina Kysla	11/25/2021 10:00 AM	Alpha Business	Not started	
Meeting	Peter Moore	12/16/2021 3:40 PM	Console Solutions	Not started	
To do	Marina Kysla	11/25/2021 7:15 AM		Not started	
<input checked="" type="checkbox"/> Call client, confirm meeting time	Valerie E. Murphy	11/21/2021 9:30 AM	Build Technologies	Completed	
<input checked="" type="checkbox"/> Web-form optimization for Firefox	Marina Kysla	9/26/2021 4:00 PM		Completed	
<input checked="" type="checkbox"/> Testing the hardware (System Patriot Tower E3-1220V3)	William Walker	9/16/2021 4:15 PM		Completed	

В результате выбора действия [ Назначить ответственного ] ([ Assign Owner ]) отображается окно справочника [ Контакты ] ([ Contacts ]). Выбранный контакт устанавливается в колонке [ Ответственный ] ([ Assignee ]) в качестве ответственного для выбранных активностей.

<input type="checkbox"/>	Work with emails, calendar	To do	Marina Kysla	11/25/2021 7:15 AM		Not started
<input type="checkbox"/>	Call client, confirm meeting time	Call	Tran Manzo	11/21/2021 9:30 AM	Build Technologies	Completed
<input type="checkbox"/>	Web-form optimization for Firefox	To do	Tran Manzo	9/26/2021 4:00 PM		Completed
<input type="checkbox"/>	Testing the hardware (System Patriot Tower E3-1220V3)	To do	Tran Manzo	9/16/2021 4:15 PM		Completed
<input type="checkbox"/>	Contact customer, specify need, budget, decision-making role.	To do	Mary King	11/21/2021 11:16 AM		Completed

Для отмены режима выбора нескольких записей в меню кнопки [ Действия ] ([ Actions ]) нажмите [ Отменить множественный выбор ] ([ Cancel multiple selection ]).

Если в реестре раздела [ Активности ] ([ Activities ]) не выбрана ни одна запись, то действие [ Назначить ответственного ] ([ Assign Owner ]) неактивно.



The screenshot shows the 'Activities' section in the Creatio interface. On the left, there's a sidebar with 'NEW TASK' and 'ACTIONS' buttons. The 'ACTIONS' menu is open, showing options like 'Synchronize activities', 'Select multiple records', 'Select all', 'Export to Excel', 'Data import', 'Change log setup', and 'Assign Owner' (which is highlighted with a red box). The main area displays a table of activities with columns: Assignee, Due, Account, and Status. The table contains several rows of data, including tasks assigned to Valerie E. Murphy, Marina Kysla, and Peter Moore.

Assignee	Due	Account	Status
Valerie E. Murphy	11/28/2021 2:00 PM		Not started
Marina Kysla	11/24/2021 9:45 AM	Alpha Business	Not started
Marina Kysla	11/25/2021 10:00 AM	Alpha Business	Not started
Peter Moore	12/16/2021 3:40 PM	Console Solutions	Not started
Marina Kysla	11/25/2021 7:15 AM		Not started

## Настроить условия отображения реестра раздела

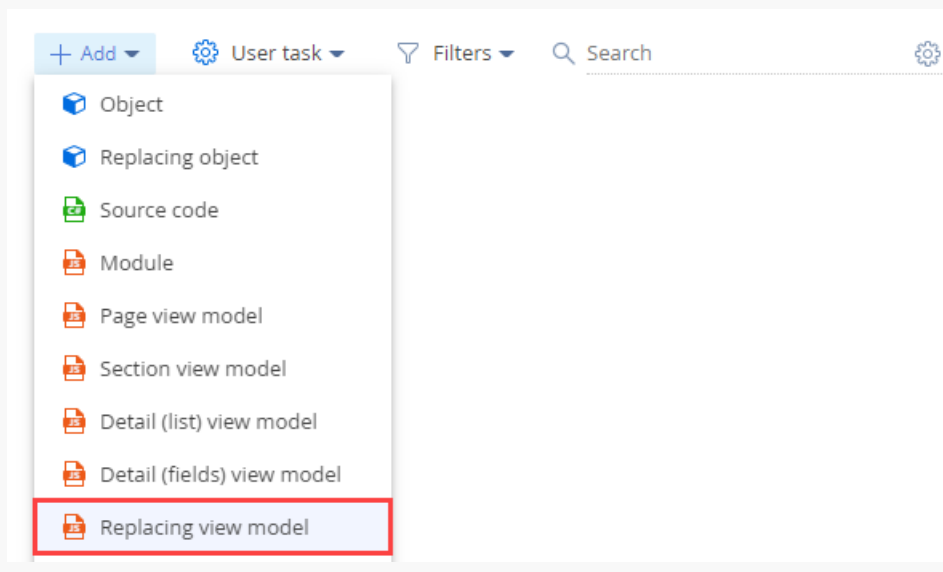
Средний

Пример реализован для продуктов линейки Sales Creatio.

**Пример.** Для реестра раздела [ Заказы ] ([ Orders ]) выделять цветом заказы, которые находятся на стадии [ Исполнение ] ([ In progress ]).

## Создать схему замещающей модели представления раздела

1. [Перейдите в раздел \[ Конфигурация \]](#) ([ Configuration ]) и выберите пользовательский [пакет](#), в который будет добавлена схема.
2. На панели инструментов реестра раздела нажмите [ Добавить ] —> [ Замещающая модель представления ] ([ Add ] —> [ Replacing view model ]).



### 3. Заполните **свойства схемы**.

- [ Код ] ([ Code ]) — "OrderSectionV2".
- [ Заголовок ] ([ Title ]) — "Раздел заказа" ("Order section").
- [ Родительский объект ] ([ Parent object ]) — выберите "OrderSectionV2".

### 4. Настройте **условия отображения реестра раздела**. Для этого в свойстве `methods` реализуйте метод `prepareResponseCollectionItem()`, который переопределяет базовый метод, модифицирует строку данных перед загрузкой в реестр, а также добавляет настраиваемые стили к определенным строкам реестра.

Исходный код схемы замещающей модели представления раздела представлен ниже.

```
OrderSectionV2
```

```

define("OrderSectionV2", ["OrderConfigurationConstants"], function(OrderConfigurationConstant
    return {
        /* Название схемы объекта раздела. */
        entitySchemaName: "Order",
        /* Методы модели представления раздела. */
        methods: {
            /* Переопределение базового метода, который модифицирует строку данных перед загр
            prepareResponseCollectionItem: function(item) {
                /* Вызывает базовый метод. */
                this.callParent(arguments);
                item.customStyle = null;
                /* Определяет статус заказа. */
                var running = item.get("Status");
                /* Если состояние заказа "Исполнение", меняется стиль записи. */
                if (running.value === OrderConfigurationConstants.Order.OrderStatus.Running)
                    item.customStyle = {
                        /* Цвет текста — белый. */
                        "color": "white",
                        /* Цвет фона — зеленый. */
                        "background": "#8ecb60"
                    };
            }
        }
    }
});

```

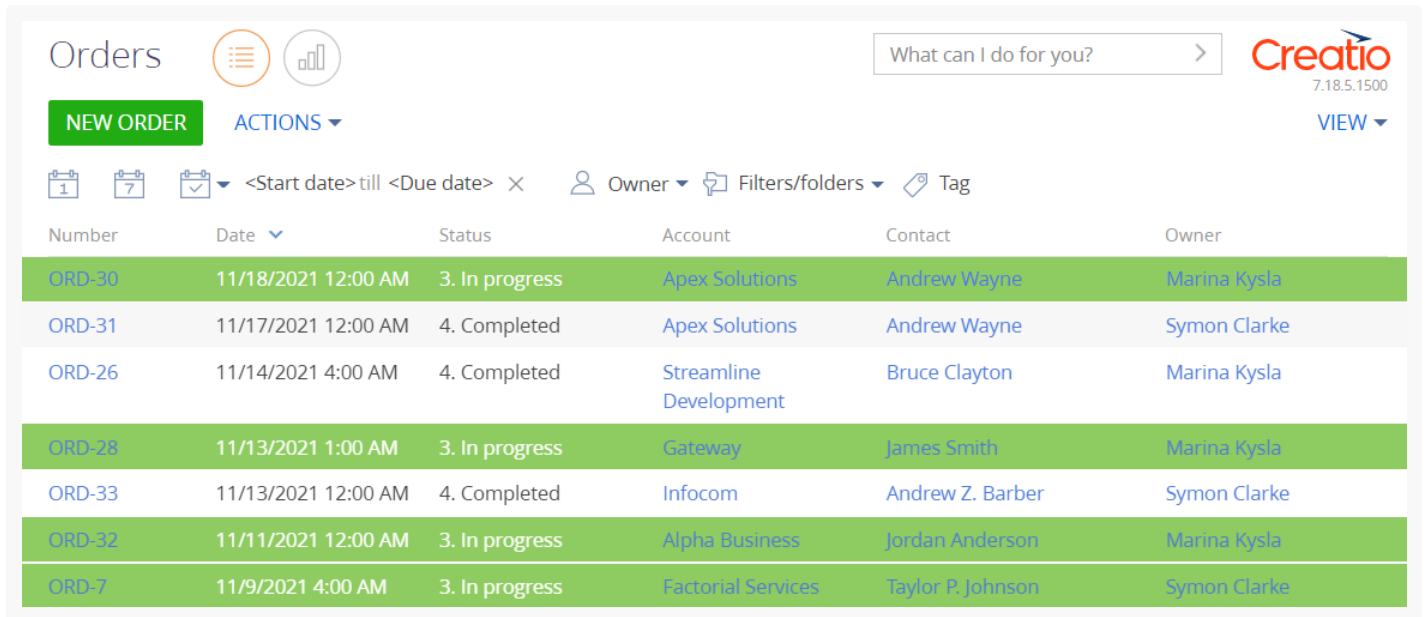
5. На панели инструментов дизайнера нажмите [ *Сохранить* ] ([ *Save* ]).

## Результат выполнения примера

Чтобы **посмотреть результат выполнения примера**:

1. Очистите кэш браузера.
2. Обновите страницу раздела [ *Заказы* ] ([ *Orders* ]).

В результате выполнения примера заказы, которые находятся на стадии [ *Исполнение* ] ([ *In progress* ]), выделяются цветом.



Number	Date	Status	Account	Contact	Owner
ORD-30	11/18/2021 12:00 AM	3. In progress	Apex Solutions	Andrew Wayne	Marina Kysla
ORD-31	11/17/2021 12:00 AM	4. Completed	Apex Solutions	Andrew Wayne	Symon Clarke
ORD-26	11/14/2021 4:00 AM	4. Completed	Streamline Development	Bruce Clayton	Marina Kysla
ORD-28	11/13/2021 1:00 AM	3. In progress	Gateway	James Smith	Marina Kysla
ORD-33	11/13/2021 12:00 AM	4. Completed	Infocom	Andrew Z. Barber	Symon Clarke
ORD-32	11/11/2021 12:00 AM	3. In progress	Alpha Business	Jordan Anderson	Marina Kysla
ORD-7	11/9/2021 4:00 AM	3. In progress	Factorial Services	Taylor P. Johnson	Symon Clarke

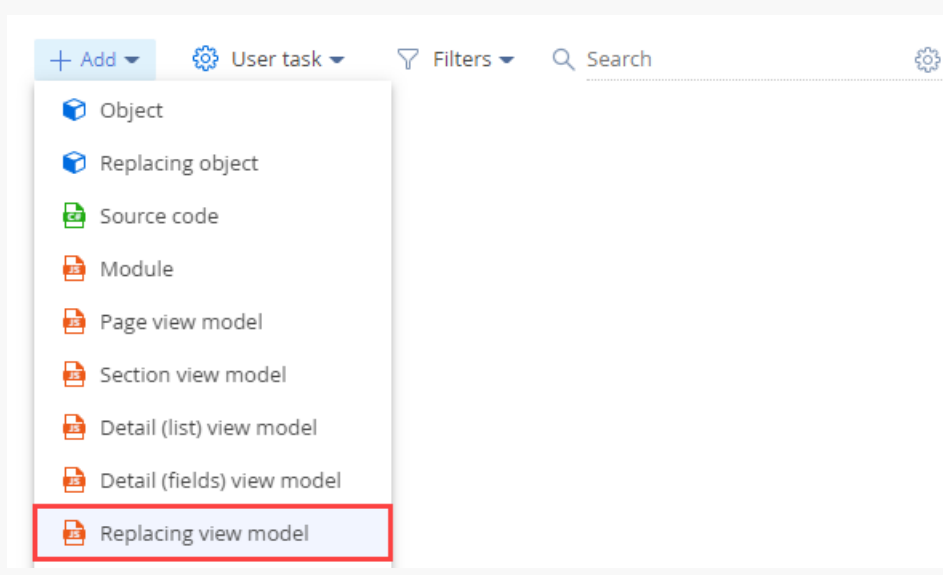
## Настроить блок быстрых фильтров раздела

 Средний

**Пример.** В раздел [ *Договоры* ] ([ *Contracts* ]) добавить блок быстрых фильтров. Договоры фильтруются по дате начала договора и по ответственному.

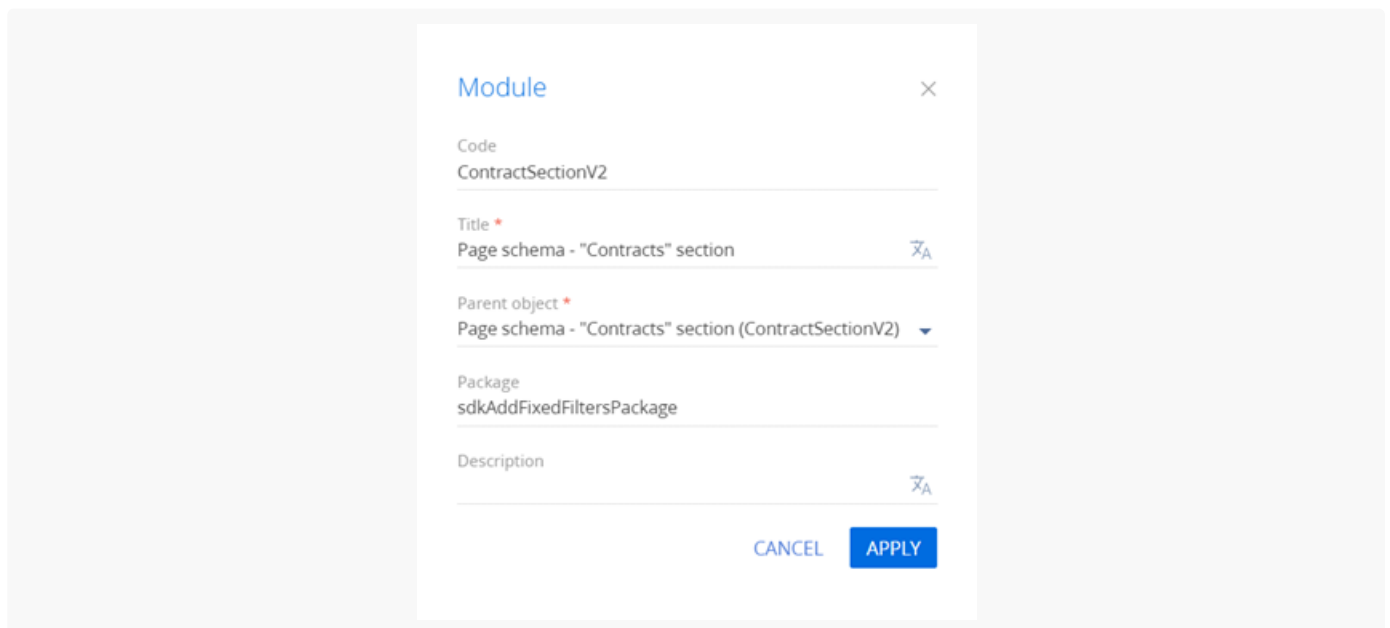
## Создать схему замещающей модели представления раздела

1. [Перейдите в раздел \[ Конфигурация \]](#) ([ *Configuration* ]) и выберите пользовательский [пакет](#), в который будет добавлена схема.
2. На панели инструментов реестра раздела нажмите [ *Добавить* ] —> [ *Замещающая модель представления* ] ([ *Add* ] —> [ *Replacing view model* ]).



### 3. Заполните **свойства схемы**.

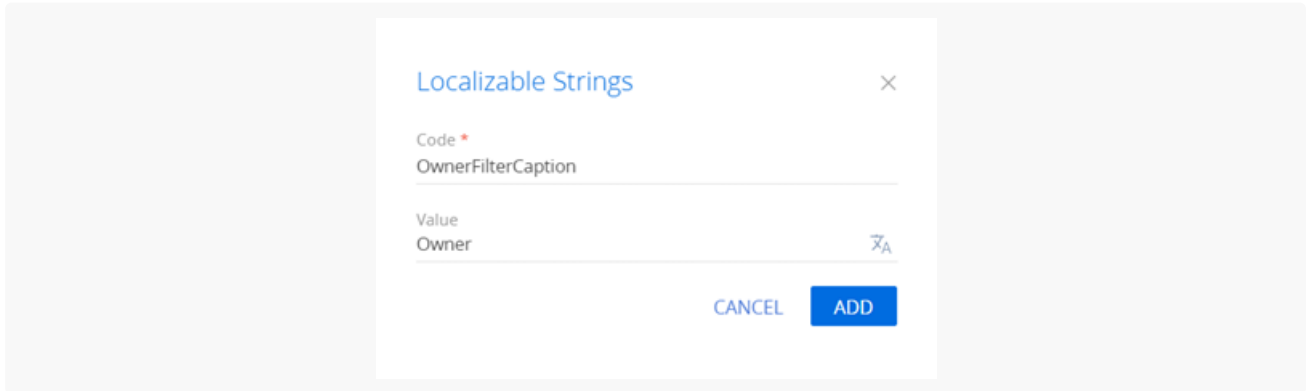
- [ Код ] ([ Code ]) — "ContractSectionV2".
- [ Заголовок ] ([ Title ]) — "Схема страницы раздела "Договоры"" ("Page schema - "Contracts" section").
- [ Родительский объект ] ([ Parent object ]) — выберите "ContractSectionV2".



### 4. Добавьте **локализуемые строки** с названиями фильтров.

- Добавьте локализуемую строку, которая содержит **название фильтра по ответственному сотруднику**.
  - В контекстном меню узла [ Локализуемые строки ] ([ Localizable strings ]) нажмите кнопку **+**.
  - Заполните **свойства локализуемой строки**.
    - [ Код ] ([ Code ]) — "OwnerFilterCaption".

- [ *Значение* ] ([ *Value* ]) — "Ответственный" ("Owner").



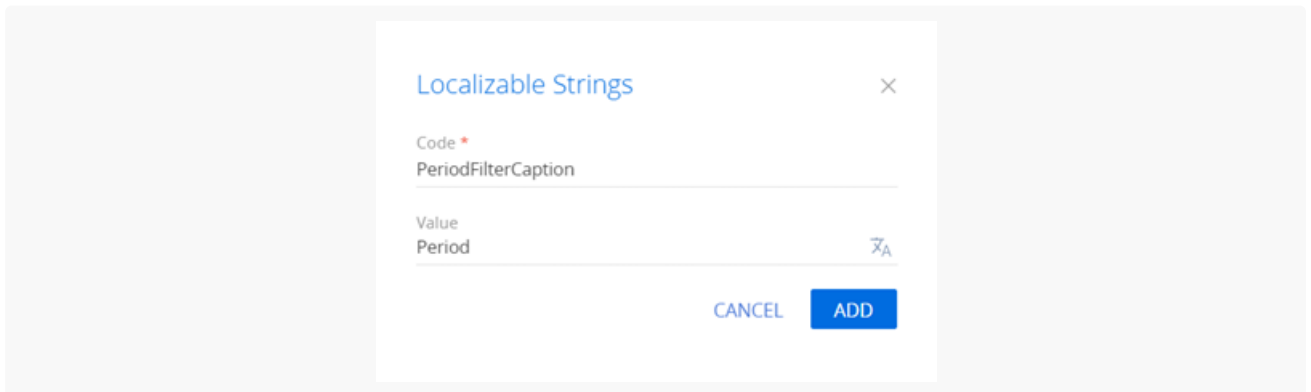
е. Для добавления локализуемой строки нажмите [ *Добавить* ] ([ *Add* ]).

б. Добавьте локализуемую строку, которая содержит **название фильтра по периоду**.

а. В контекстном меню узла [ *Локализуемые строки* ] ([ *Localizable strings* ]) нажмите кнопку **+**.

б. Заполните **свойства локализуемой строки**.

- [ *Код* ] ([ *Code* ]) — "PeriodFilterCaption".
- [ *Значение* ] ([ *Value* ]) — "Период" ("Period").



е. Для добавления локализуемой строки нажмите [ *Добавить* ] ([ *Add* ]).

5. Реализуйте **логику работы фильтрации**. Для этого в свойстве `methods` реализуйте метод `initFixedFiltersConfig()`. В методе создайте конфигурационный объект с массивом фильтров `PeriodFilter` и `OwnerFilter`, присвойте ссылку на объект атрибуту `fixedFiltersConfig` модели представления.

Исходный код схемы замещающей модели представления раздела представлен ниже.

**ContractSectionV2**

```
define("ContractSectionV2", ["BaseFiltersGenerateModule"], function(BaseFiltersGenerateModule) {
    return {
        /* Название схемы объекта раздела. */
        entitySchemaName: "Contract",
```

```

/* Методы модели представления раздела. */
methods: {
    /* Инициализирует фиксированные фильтры. */
    initFixedFiltersConfig: function() {
        /* Создает конфигурационный объект. */
        var fixedFilterConfig = {
            /* В качестве схемы объекта для фиксированных фильтров указывается схема
            entitySchema: this.entitySchema,
            /* Массив фильтров. */
            filters: [
                /* Фильтр периода. */
                {
                    /* Название фильтра. */
                    name: "PeriodFilter",
                    /* Заголовок фильтра. */
                    caption: this.get("Resources.Strings.PeriodFilterCaption"),
                    /* Тип данных – дата. */
                    dataValueType: this.Terrasoft.DataValueType.DATE,
                    /* Дата начала периода фильтрации. */
                    startDate: {
                        /* Фильтруются данные из колонки [Date]. */
                        columnName: "StartDate",
                        /* Значение по умолчанию – начало текущей недели. */
                        defValue: this.Terrasoft.startOfWeek(new Date())
                    },
                    /* Дата завершения периода фильтрации – завершение текущей недели
                    dueDate: {
                        columnName: "StartDate",
                        defValue: this.Terrasoft.endOfWeek(new Date())
                    }
                },
                /* Фильтр ответственного. */
                {
                    /* Название фильтра. */
                    name: "Owner",
                    /* Заголовок фильтра. */
                    caption: this.get("Resources.Strings.OwnerFilterCaption"),
                    /* Фильтрация данных из колонки [Owner]. */
                    columnName: "Owner",
                    /* Значение по умолчанию – контакт текущего пользователя, который
                    defValue: this.Terrasoft.SysValue.CURRENT_USER_CONTACT,
                    /* Тип данных – справочник. */
                    dataValueType: this.Terrasoft.DataValueType.LOOKUP,
                    /* Фильтр. */
                    filter: BaseFiltersGenerateModule.OwnerFilter
                }
            ]
        };
        /* Атрибуту [FixedFilterConfig] присваивается ссылка на созданный конфигураци

```

```

        this.set("FixedFilterConfig", fixedFilterConfig);
    }
}
};
});

```

6. На панели инструментов дизайнера нажмите [ Сохранить ] ([ Save ]).

## Результат выполнения примера

Чтобы **посмотреть результат выполнения примера**, обновите страницу раздела [ *Договоры* ] ([ *Contracts* ]).

В результате выполнения примера в разделе [ *Договоры* ] ([ *Contracts* ]) отображается блок фиксированных фильтров, который позволяет фильтровать договоры как по дате их начала, так и по ответственному сотруднику.

Number	Start date	Type	Account	Owner	Status
201 (sample)	4/11/2021	Contract	Accom (sample)	Supervisor	Draft
322	10/29/2014	Contract	Axiom	Marina Kysla	Signed
324	10/28/2016	Contract	Alpha Business	Marina Kysla	Signed

## Схема BaseSectionV2 JS

### Оснoвы

**BaseSectionV2** — базовая схема раздела. Предоставляет базовую логику раздела. Реализована в пакете **NUI**. Схема является схемой модели представления. Описание свойств схемы содержится в статье [Клиентская схема](#). Все схемы разделов должны наследовать схему **BaseSectionV2**.

## Сообщения



Сообщения базового раздела

Название	Режим	Направление	Описание
Rerender Module	Адресное	Публикация	Повторно отобразить сообщение модуля дашбордов.
ReloadData OnRestore	Широковещательное	Подписка	Указывает на необходимость перезагрузки данных при следующей перезагрузке.
Selected Package Result	Адресное	Подписка	Выбранный результат.

Режимы сообщений представлены перечислением `Terrasoft.core.enums.MessageMode`, а направления сообщений — перечислением `Terrasoft.core.enums.MessageDirectionType`. Перечисление `MessageMode` описано в [Библиотеке JS классов](#). Перечисление `MessageDirectionType` описано в [Библиотеке JS классов](#).

## Атрибуты

ChartEditSchemaName TEXT

Схема редактирования `SchemaName`.

IsEmptyChart BOOLEAN

Признак пустой схемы.

AnalyticsChartActiveRow GUID

Активная строка схемы аналитики.

AnalyticsGridData COLLECTION

Коллекция представлений аналитики реестра.

IsAnalyticsPrintButtonVisible BOOLEAN

Видимость кнопки печати аналитической формы.

AnalyticsData COLLECTION

Коллекция данных аналитики.

---

`IsAnalyticsActionButtonsContainerVisible` `BOOLEAN`

Признак видимости кнопок аналитических действий.

---

`AnalyticsDataViewName` `TEXT`

Посмотреть название раздела аналитики.

---

`IsBindDataActionVisible` `BOOLEAN`

Признак привязки данных.

Типы данных атрибутов представлены перечислением `Terrasoft.core.enums.DataValueType`.

Перечисление `DataValueType` описано в [Библиотеке JS классов](#).

## Методы

---

`checkCanManageAnalytics()`

Проверяет, имеет ли пользователь право выполнять CRUD-операции с расписанием. Права доступа устанавливаются системной операцией [ *Настройка аналитики* ] ([ *Analytics setup* ], код `CanManageAnalytics` ).

---

`onCanManageAnalytics(result)`

Устанавливает атрибут `CanManageAnalytics` в зависимости от значения запрошенных системной настройки [ *Отображать Демо ссылки* ] ([ *Display demo links* ], код `ShowDemoLinks` ) и системной операции [ *Настройка аналитики* ] ([ *Analytics setup* ], код `CanManageAnalytics` ).