

# Кнопка

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>Кнопка</b>	4
Контейнеры кнопок	4
Добавить кнопку	6
Добавить к кнопке всплывающую подсказку	7
<b>Добавить кнопку на панель инструментов раздела</b>	8
Создать схему замещающей модели представления раздела	9
Результат выполнения примера	12
<b>Добавить кнопку на панель инструментов страницы записи в совмещенном режиме</b>	13
Создать схему замещающей модели представления раздела	14
Результат выполнения примера	18
<b>Добавить кнопку на панель инструментов страницы добавления записи</b>	20
1. Изменить способ добавления контрагента	20
2. Создать схему замещающей модели представления страницы контрагента	20
Результат выполнения примера	23
<b>Добавить кнопку выбора цвета на страницу записи</b>	25
1. Создать схему замещающего объекта	25
2. Создать схему замещающей модели представления страницы контрагента	27
Результат выполнения примера	29
<b>Добавить к кнопке всплывающую подсказку</b>	30
Создать схему замещающей модели представления страницы контакта	30
Результат выполнения примера	34
<b>Добавить кнопку в строку записи раздела</b>	34
Создать схему замещающей модели представления раздела	34
Результат выполнения примера	38
<b>Свойство diff объекта кнопки</b>	38
Свойства	38

# Кнопка



## Контейнеры кнопок

**Виды** контейнеров кнопок, которые реализованы в Creatio:

- **Контейнер кнопок действий** — содержит кнопки действий страницы раздела. Содержит контейнеры стандартных кнопок и контейнер кнопок с выпадающим меню.
- **Контейнер стандартных кнопок** — содержит кнопки [ Сохранить ] ([ Save ]), [ Отмена ] ([ Cancel ]), [ Теги ] ([ Tags ]) и выпадающее меню кнопки [ Действия ] ([ Actions ]).
- **Контейнер кнопок с выпадающим меню** — содержит выпадающие меню кнопок [ Печать ] ([ Print ]) и [ Вид ] ([ View ]).

Контейнеры кнопок отличаются для страницы раздела, страницы записи и страницы добавления записи.

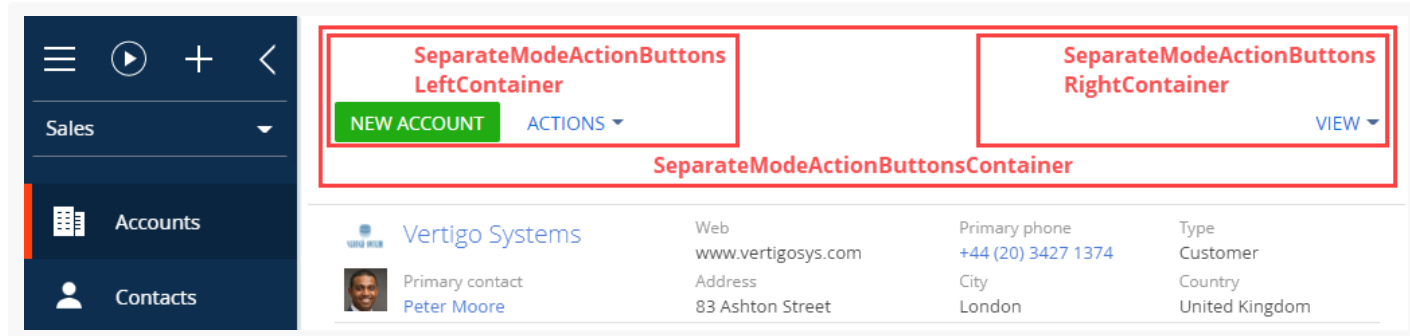
**На заметку.** В приложении используются мета-имена html-контейнеров. На основании мета-имен приложение формирует фактические идентификаторы соответствующих html-элементов страницы записи.

Мета-имена контейнеров кнопок представлены в таблице ниже.

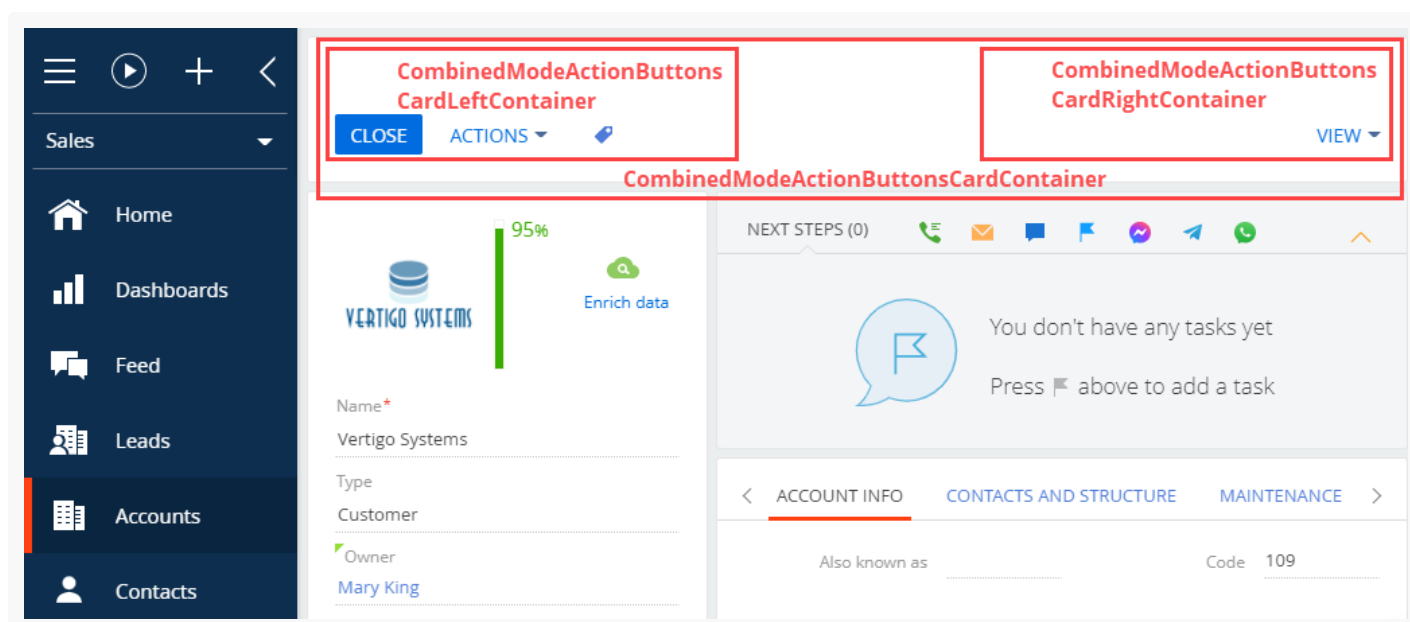
Мета-имена контейнеров кнопок

Название элемента интерфейса	Контейнеры		
	Контейнер кнопок действий	Контейнер стандартных кнопок	Контейнер кнопок с выпадающим меню
Страница раздела	SeparateModeActionButtonsContainer	SeparateModeActionButtonsLeftContainer	SeparateModeActionButtonsRightContainer
Страница записи	CombinedModeActionButtonsCardContainer	CombinedModeActionButtonsCardLeftContainer	CombinedModeActionButtonsCardRightContainer
Страница добавления записи	ActionButtonsContainer	LeftContainer	RightContainer

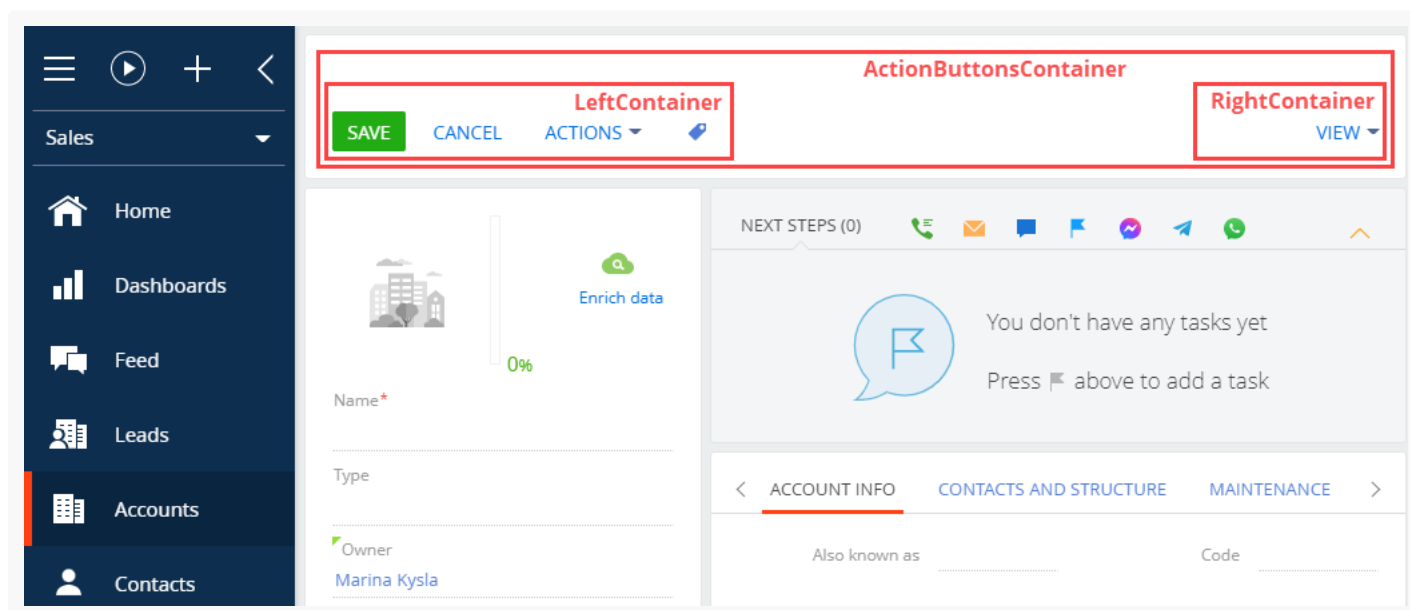
Контейнеры кнопок **страницы раздела** представлены на рисунке ниже.



Контейнеры кнопок **страницы записи** представлены на рисунке ниже.



Контейнеры кнопок **страницы добавления записи** представлены на рисунке ниже.



## Добавить кнопку

Алгоритм, который необходимо использовать для добавления кнопки, зависит от вида кнопки, которую планируется добавить.

**Виды кнопок**, которые реализованы в Creatio:

- Простая кнопка.
- Кнопка выбора цвета.

## Добавить простую кнопку на страницу записи или раздела

1. Создайте схему замещающей модели представления страницы записи или раздела, на которой будет размещена кнопка. Для этого воспользуйтесь инструкцией, которая приведена в статье [Разработка конфигурационных элементов](#).
2. В схему замещающей модели представления добавьте локализуемую строку, которая содержит название кнопки. Для этого воспользуйтесь инструкцией, которая приведена в статье [Операции с локализуемыми ресурсами](#).
3. В схеме замещающей модели представления реализуйте **логику работы кнопки**:
  - a. В свойстве `methods` реализуйте:
    - Метод-обработчика, который вызывается по нажатию на кнопку.
    - Вспомогательные методы, которые необходимы для функционирования элемента управления. Это могут быть методы, которые управляют видимостью или доступностью элемента управления.
  - d. В массив модификаций `diff` добавьте конфигурационный объект с настройками расположения кнопки на странице записи или раздела. В массиве модификаций укажите соответствующий контейнер, в котором планируется разместить кнопку.

**Важно. Совмещенный режим** — режим отображения страницы записи, у которой открыт вертикальный реестр. Чтобы **добавить кнопку на страницу записи в совмещенном режиме**, внесите изменения в схему замещающей модели представления страницы раздела и в схему замещающей модели представления страницы записи.

## Добавить простую кнопку в строку записи раздела

1. Создайте схему замещающей модели представления раздела, в котором будет размещена кнопка. Для этого воспользуйтесь инструкцией, которая приведена в статье [Разработка конфигурационных элементов](#).
2. В схему замещающей модели представления добавьте локализуемую строку, которая содержит название кнопки. Для этого воспользуйтесь инструкцией, которая приведена в статье [Операции с локализуемыми ресурсами](#).
3. В схеме замещающей модели представления реализуйте **логику работы кнопки**:

- a. В свойстве `methods` реализуйте:
  - метод `onActiveRowAction()` — переопределенный базовый метод схемы `BaseDataView` пакета `NUI`, который связывает кнопку с методом-обработчиком.
  - Метод-обработчика, который вызывается по нажатию на кнопку.
- d. В массив модификаций `diff` добавьте конфигурационный объект с настройками расположения кнопки на странице раздела.
  - В свойстве `parentName` укажите контейнер `DataGrid`.
  - в свойстве `propertyName` укажите коллекцию `activeRowActions`.
  - Вместо свойства `itemType` укажите свойство `className`, для которого установите значение `Terrasoft.Button`.
  - Укажите свойство `tag`, которое будет идентифицировать кнопку в методе `onActiveRowAction()`.

## Добавить кнопку выбора цвета

1. Создайте схему замещающего объекта. Для этого воспользуйтесь инструкцией, которая приведена в статье [Разработка конфигурационных элементов](#).
2. В схему замещающего объекта добавьте колонку типа [ *Строка (50 символов)* ] ([ *Text (50 characters)* ]), которая будет хранить информацию о выбранном цвете. Для этого воспользуйтесь инструкцией, которая приведена в статье [Разработка конфигурационных элементов](#).
3. Создайте схему замещающей модели представления страницы записи или раздела, на которой будет размещена кнопка. Для этого воспользуйтесь инструкцией, которая приведена в статье [Разработка конфигурационных элементов](#).
4. В схеме замещающей модели представления реализуйте **логику работы кнопки**:
  - a. В свойстве `methods` реализуйте:
    - Метод-обработчика, который вызывается по нажатию на кнопку.
    - Вспомогательные методы, которые необходимы для функционирования элемента управления. Это могут быть методы, которые управляют видимостью или доступностью элемента управления.
  - d. В массив модификаций `diff` добавьте конфигурационный объект с настройками расположения кнопки на странице записи или раздела.
    - В свойстве `itemType` укажите тип `COLOR_BUTTON`.
    - В свойстве `value` установите привязку к добавленной колонке схемы замещающего объекта.

## Добавить к кнопке всплывающую подсказку

**Всплывающие подсказки** — текстовые сообщения, которые предоставляют пользователю дополнительную информацию о функциональности кнопки. Всплывающая подсказка к кнопке отображается при наведении курсора на кнопку.

Чтобы **добавить к кнопке всплывающую подсказку**:

1. Создайте схему замещающей модели представления страницы записи или раздела, на которой будет размещена кнопка. Для этого воспользуйтесь инструкцией, которая приведена в статье [Разработка конфигурационных элементов](#).
2. В схему замещающей модели представления добавьте локализуемую строку, которая содержит название кнопки. Для этого воспользуйтесь инструкцией, которая приведена в статье [Операции с локализуемыми ресурсами](#).
3. В массиве модификаций `diff` схемы замещающей модели представления реализуйте **всплывающую подсказку**.

**Способы** добавления всплывающей подсказки в конфигурационный объект кнопки:

- Свойство `hint`.
- Свойство `tips`.

Чтобы добавить всплывающую подсказку к кнопке **с использованием свойства** `hint`, в свойство `values` элемента управления добавьте свойство `hint`, которое содержит текст всплывающей подсказки.

Чтобы добавить всплывающую подсказку к кнопке **с использованием свойства** `tips`:

1. В свойство `values` элемента управления добавьте свойство `tips`.
2. В массив `tips` с помощью операции `insert` добавьте конфигурационный объект подсказки.
3. В свойстве `values` конфигурационного объекта подсказки добавьте свойство `content`, которое содержит текст всплывающей подсказки.

Использование свойства `tips` для добавления всплывающей подсказки к кнопке позволяет:

- Изменить стиль отображения.
- Привязать видимость подсказки к событию модели представления.
- Добавить элементы управления и т. д.

**Типы элементов**, которые позволяют использовать свойство `tips`:

- `Terrasoft.ViewItemType.BUTTON`.
- `Terrasoft.ViewItemType.LABEL`.
- `Terrasoft.ViewItemType.COLOR_BUTTON`.
- `Terrasoft.ViewItemType.HYPERLINK`.
- `Terrasoft.ViewItemType.INFORMATION_BUTTON`.
- Элементы, для которых указано свойство `generator`.

## Добавить кнопку на панель инструментов раздела



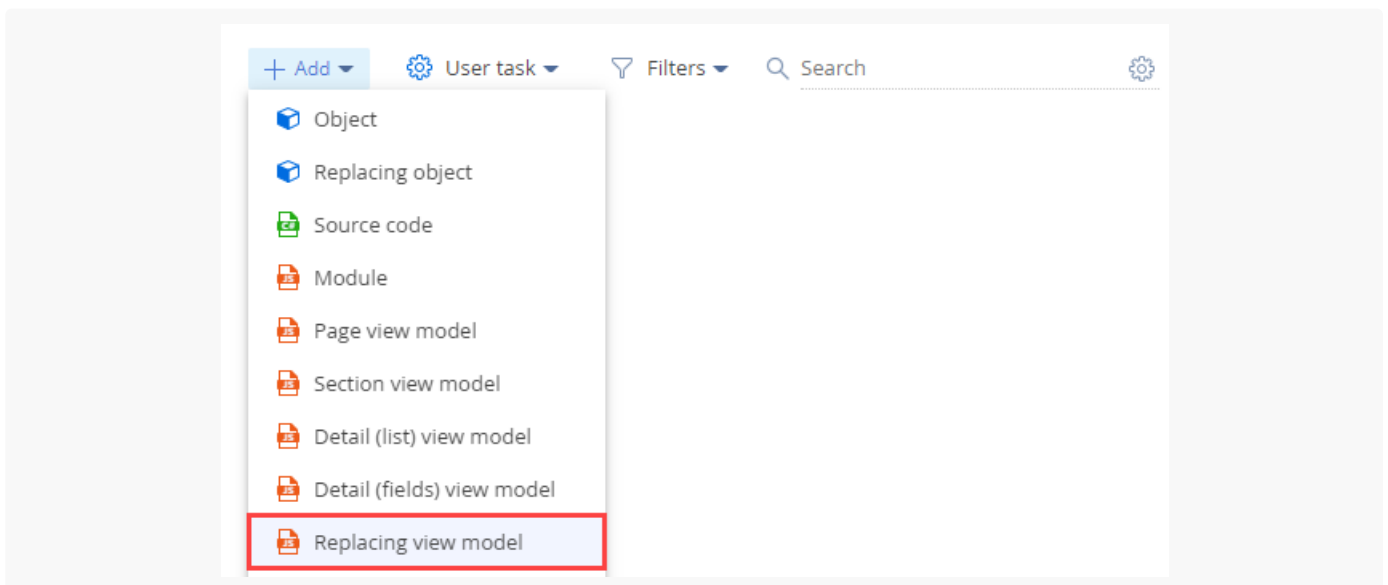
Средний



**Пример.** Добавить кнопку на панель инструментов раздела [ *Контрагенты* ] ([ *Accounts* ]). Кнопка активна при выборе в реестре раздела контрагента, для которого указан основной контакт. При нажатии на кнопку открывается страница основного контакта активного контрагента.

## 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* ]) — "AccountSectionV2".
- [ *Заголовок* ] ([ *Title* ]) — "Раздел контрагенты" ("Account section").
- [ *Родительский объект* ] ([ *Parent object* ]) — выберите "AccountSectionV2".

**Module** [X]

Code  
AccountSectionV2

Title \*  
Accounts section [A]

Parent object \*  
Accounts section (AccountSectionV2) [v]

Package  
sdkAddButtonToSectionPackage

Description [A]

CANCEL APPLY

4. Добавьте **локализуемую строку**.

- a. В контекстном меню узла [ *Локализуемые строки* ] ([ *Localizable strings* ]) нажмите кнопку **+**.
- b. Заполните **свойства локализуемой строки**.
  - [ *Код* ] ([ *Code* ]) — "OpenPrimaryContactButtonCaption".
  - [ *Значение* ] ([ *Value* ]) — "Основной контакт" ("Primary contact").

**Localizable Strings** [X]

Code \*  
OpenPrimaryContactButtonCaption

Value  
Primary contact [A]

CANCEL ADD

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

5. Реализуйте **логику работы кнопки**.

- a. В свойстве `methods` реализуйте **методы**:
  - `isAccountPrimaryContactSet()` — проверяет заполнение поля [ *Основной контакт* ] ([ *Primary contact* ]) страницы.
  - `onOpenPrimaryContactClick()` — метод-обработчик нажатия кнопки. Выполняет переход на страницу основного контакта.
- d. В массив модификаций `diff` добавьте конфигурационный объект с настройками расположения кнопки на странице.

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, например, из коллекции данных списочного представления реестра раздела, которая хранится в свойстве `GridData` модели представления реестра.

Исходный код схемы замещающей модели представления страницы раздела представлен ниже.

#### AccountSectionV2

```
define("AccountSectionV2", [], function() {
    return {
        /* Название схемы объекта раздела. */
        entitySchemaName: "Account",
        /* Методы модели представления раздела. */
        methods: {
            /* Метод-обработчик нажатия кнопки. */
            onOpenPrimaryContactClick: function() {
                /* Получение идентификатора выбранной записи. */
                var activeRow = this.get("ActiveRow");
                if (!activeRow) {
                    return;
                }
                /* Определение идентификатора основного контакта. */
                var primaryId = this.get("GridData").get(activeRow).get("PrimaryContact").val;
                if (!primaryId) {
                    return;
                }
                /* Формирование строки адреса. */
                var requestUrl = "CardModuleV2/ContactPageV2/edit/" + primaryId;
                /* Публикация сообщения о пополнении истории навигации по страницам и переход
                this.sandbox.publish("PushHistoryState", {
                    hash: requestUrl
                });
            },
            /* Проверяет заполнение поля [Основной контакт] выбранного элемента. */
            isAccountPrimaryContactSet: function() {
                var activeRow = this.get("ActiveRow");
                if (!activeRow) {
                    return false;
                }
                var pc = this.get("GridData").get(activeRow).get("PrimaryContact");
                return (pc || pc !== "") ? true : false;
            }
        },
        /* Отображение кнопки в разделе. */
        diff: /**SCHEMA_DIFF*/[
```

```

/* Метаданные для добавления в раздел пользовательской кнопки. */
{
    /* Выполняется операция добавления элемента на страницу. */
    "operation": "insert",
    /* Мета-имя родительского контейнера, в который добавляется кнопка. */
    "parentName": "ActionButtonsContainer",
    /* Кнопка добавляется в коллекцию элементов родительского элемента. */
    "propertyName": "items",
    /* Мета-имя добавляемой кнопки. */
    "name": "MainContactSectionButton",
    /* Свойства, передаваемые в конструктор элемента. */
    "values": {
        /* Тип добавляемого элемента – кнопка. */
        "itemType": Terrasoft.ViewItemType.BUTTON,
        /* Привязка заголовка кнопки к локализуемой строке схемы. */
        "caption": { bindTo: "Resources.Strings.OpenPrimaryContactButtonCaption" },
        /* Привязка метода-обработчика нажатия кнопки. */
        "click": { bindTo: "onOpenPrimaryContactClick" },
        /* Привязка свойства доступности кнопки. */
        "enabled": { bindTo: "isAccountPrimaryContactSet" },
        /* Настройка расположения кнопки. */
        "layout": {
            /* Номер столбца. */
            "column": 1,
            /* Номер строки. */
            "row": 6,
            /* Диапазон занимаемых столбцов. */
            "colSpan": 1
        }
    }
}
]/**SCHEMA_DIFF*/
};
});

```

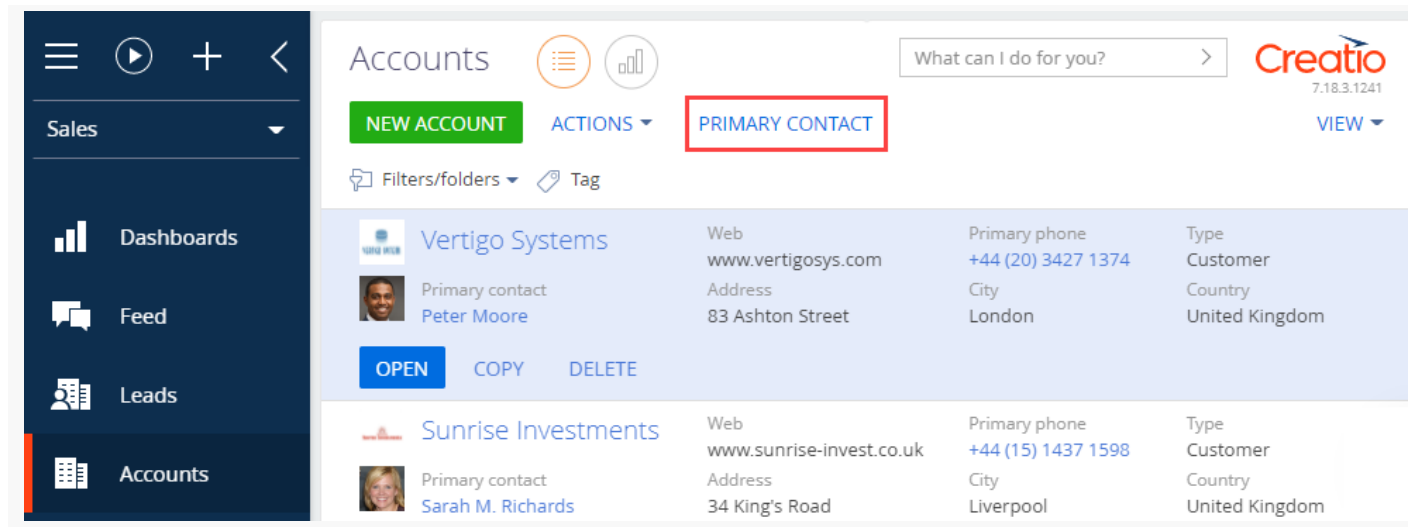
6. На панели инструментов дизайнера нажмите [ *Сохранить* ] ([ *Save* ]).

## Результат выполнения примера

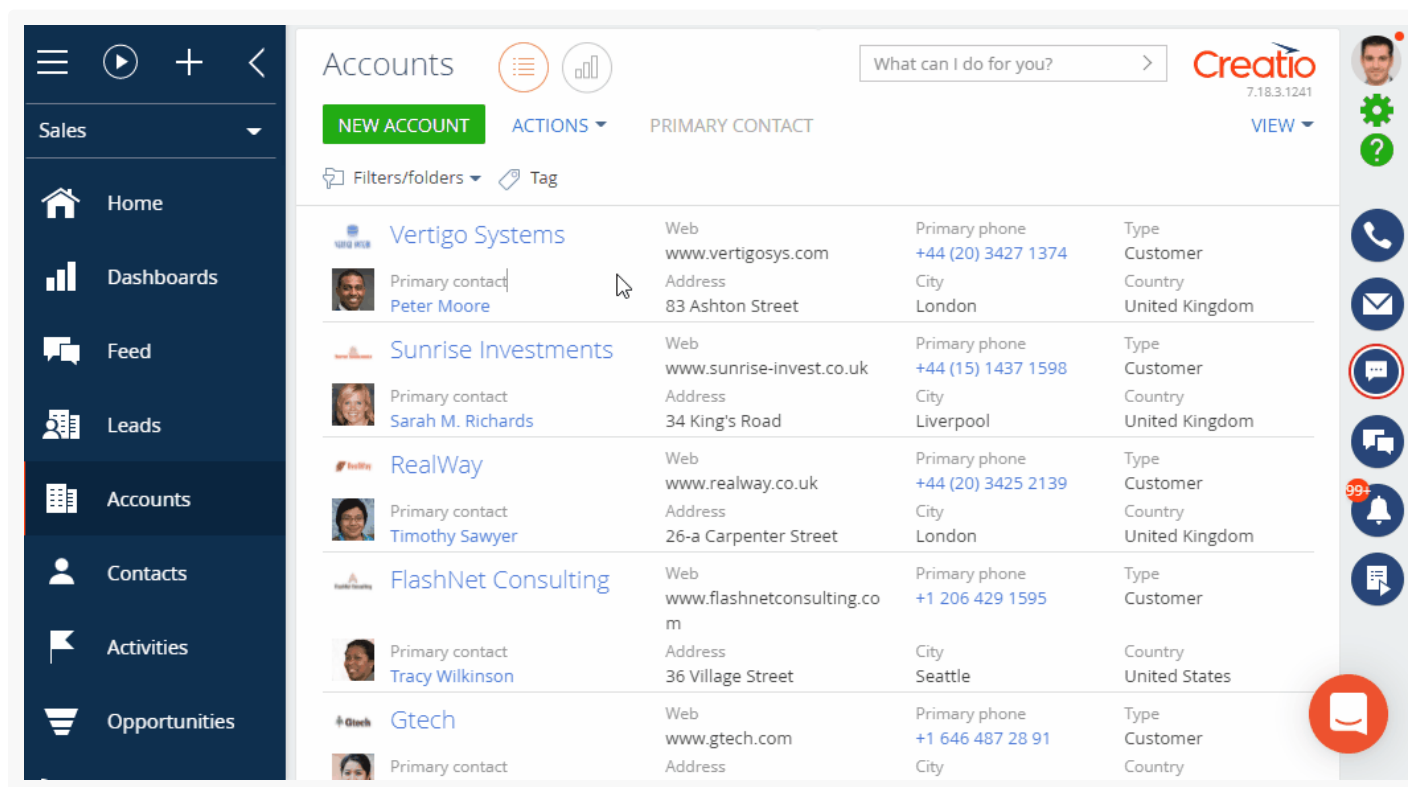
Чтобы **посмотреть результат выполнения примера**:

1. Очистите кэш браузера.
2. Обновите страницу раздела [ *Контрагенты* ] ([ *Accounts* ]).

В результате выполнения примера на панель инструментов раздела [ *Контрагенты* ] ([ *Accounts* ]) добавлена кнопка [ *Основной контакт* ] ([ *Primary contact* ]). Кнопка активна при выборе в реестре раздела контрагента, для которого указан основной контакт.



При нажатии на кнопку [ Основной контакт ] ([ Primary contact ]) открывается страница основного контакта активного контрагента.



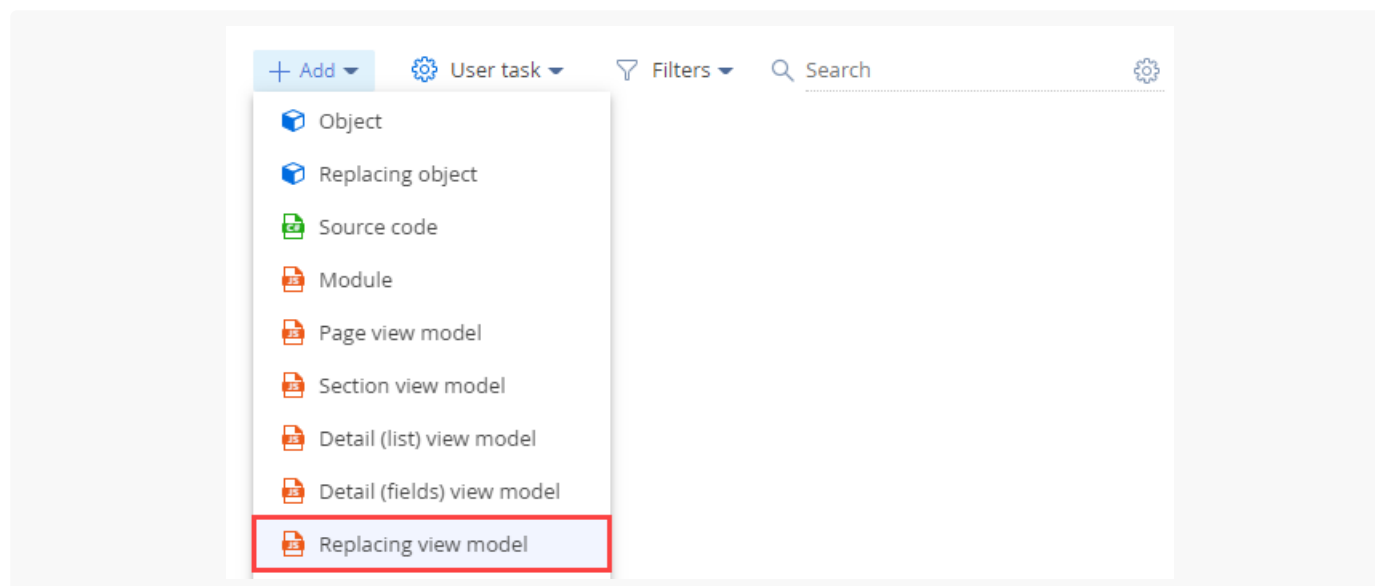
## Добавить кнопку на панель инструментов страницы записи в совмещенном режиме

**Пример.** Добавить кнопку на панель инструментов страницы контрагента, которая отображается в совмещенном режиме.

- Если для контрагента указан основной контакт, то при нажатии на кнопку открывается страница этого контакта.
- Если для контрагента не указан основной контакт, то кнопка неактивна.

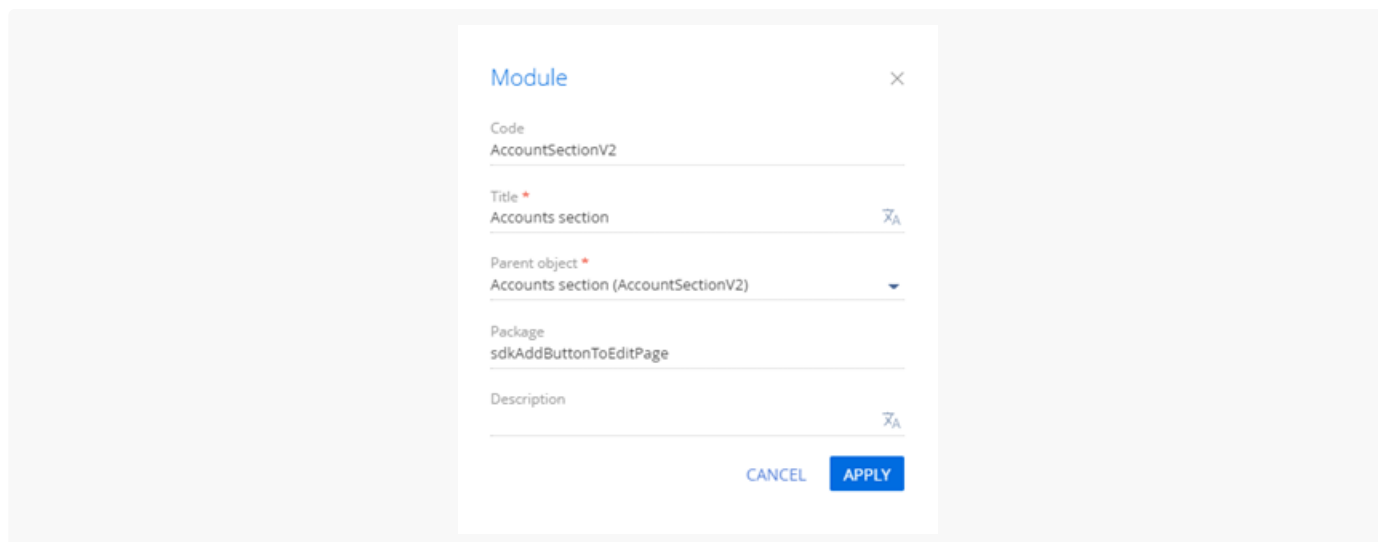
## Создать схему замещающей модели представления раздела

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* ]).



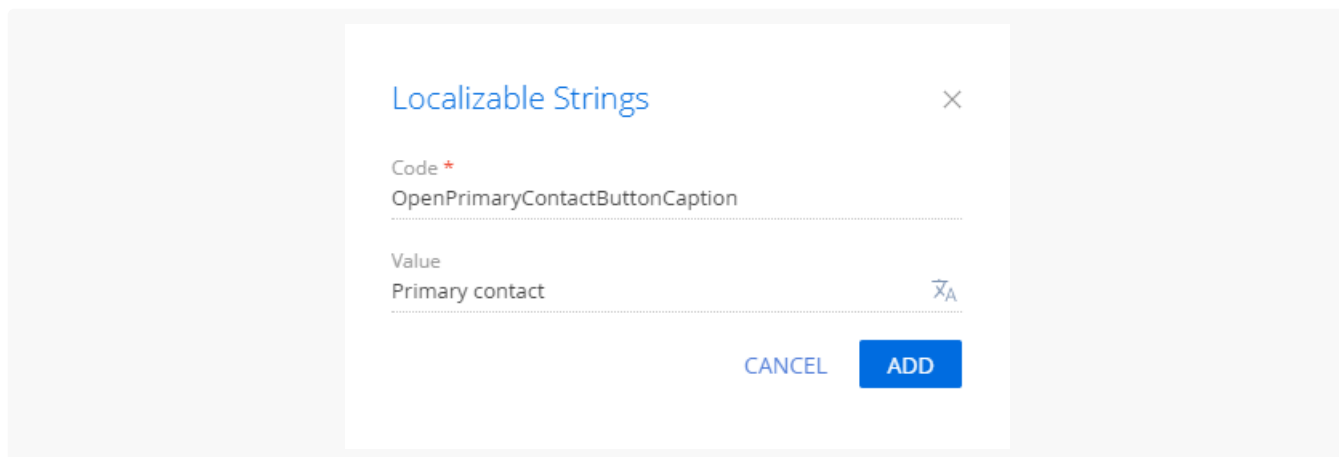
3. Заполните **свойства схемы**.

- [ *Код* ] ([ *Code* ]) — "AccountSectionV2".
- [ *Заголовок* ] ([ *Title* ]) — "Раздел контрагенты" ("Account section").
- [ *Родительский объект* ] ([ *Parent object* ]) — выберите "AccountSectionV2".



4. Добавьте **локализуемую строку**.

- a. В контекстном меню узла [ *Локализуемые строки* ] ([ *Localizable strings* ]) нажмите кнопку **+**.
- b. Заполните **свойства локализуемой строки**.
  - [ *Код* ] ([ *Code* ]) — "OpenPrimaryContactButtonCaption".
  - [ *Значение* ] ([ *Value* ]) — "Основной контакт" ("Primary contact").



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

5. Реализуйте **логику работы кнопки**.

- a. В свойство `attributes` добавьте атрибут `ButtonEnabled` типа `Terrasoft.DataValueType.BOOLEAN`, который сохраняет состояние доступности кнопки.
- b. В свойстве `methods` реализуйте **методы**:
  - `onOpenPrimaryContactClick()` — выполняет переход на страницу основного контакта.
  - `onCardRendered()` — подписка на события загрузки данных в реестр и событие изменения активной записи реестра. Выполняется после отрисовки страницы контрагента в совмещенном режиме.
  - `isPrimaryContactExist()` — определяет наличие основного контакта для активной записи

реестра.

- `setButtonEnabled()` — устанавливает значение атрибута `ButtonEnabled` в зависимости от наличия основного контакта контрагента.

g. В массив модификаций `diff` добавьте конфигурационный объект с настройками расположения кнопки на странице.

Исходный код схемы замещающей модели представления страницы раздела представлен ниже.

#### AccountSectionV2

```
define("AccountSectionV2", [], function() {
    return {
        /* Название схемы объекта раздела. */
        entitySchemaName: "Account",
        attributes: {
            /* Атрибут для хранения состояния доступности кнопки. */
            "ButtonEnabled": {
                "dataValueType": Terrasoft.DataValueType.BOOLEAN,
                "type": Terrasoft.ViewModelColumnType.VIRTUAL_COLUMN,
                "value": false
            }
        },
        /* Методы модели представления раздела. */
        methods: {
            /* Метод-обработчик нажатия кнопки. */
            onOpenPrimaryContactClick: function() {
                /* Определение активной записи вертикального реестра. */
                var activeRow = this.get("ActiveRow");
                if (activeRow) {
                    /* Определение идентификатора основного контакта. */
                    var primaryId = this.get("GridData").get(activeRow).get("PrimaryContact");
                    if (primaryId) {
                        /* Формирование строки адреса. */
                        var requestUrl = "CardModuleV2/ContactPageV2/edit/" + primaryId;
                        /* Публикация сообщения о пополнении истории навигации по страницам и
                        this.sandbox.publish("PushHistoryState", {
                            hash: requestUrl
                        });
                    }
                }
            },
            /* Выполняется после отрисовки страницы контрагента. */
            onCardRendered: function() {
                this.callParent();
                /* Данные реестра. */
                var gridData = this.get("GridData");
                var activeRow = this.get("ActiveRow");
```



```

    if (activeRow)
    {
        this.setButtonEnabled(activeRow, this);
    }
    /* После закрытия страницы основного контакта теряется активная запись. Ее ну
else {
    var historyState = this.sandbox.publish("GetHistoryState");
    var hash = historyState.hash;
    if (hash && hash.valuePairs)
    {
        activeRow = hash.valuePairs[0].name;
        /* Восстановление активной записи. */
        this.set("ActiveRow", activeRow);
        /* Сохранение контекста в локальную переменную. */
        var self = this;
        /* Подписка на событие полной загрузки данных в вертикальный реестр.
        gridData.on("dataloaded", function() {
            self.setButtonEnabled(activeRow, self);
        });
    }
}
/* Подписка на событие изменения активной записи реестра. */
gridData.on("itemchanged", function() {
    this.setButtonEnabled(activeRow, this);
}, this);
},
/* Определяет наличие основного контакта для активной записи реестра. */
isPrimaryContactExist: function(id) {
    var pc = this.get("GridData").get(id).get("PrimaryContact");
    return (pc || pc !== "") ? true : false;
},
/* Устанавливает значение атрибута ButtonEnabled в зависимости от того, определен
setButtonEnabled: function(activeRow, context) {
    if (context.isPrimaryContactExist(activeRow)) {
        context.set("ButtonEnabled", true);
    }
    else {
        context.set("ButtonEnabled", false);
    }
}
},
/* Отображение кнопки на странице записи. */
diff: [
    /* Метаданные для добавления на страницу пользовательской кнопки. */
    {
        /* Выполняется операция добавления элемента на страницу. */
        "operation": "insert",
        /* Мета-имя родительского контейнера, в который добавляется кнопка. */
        "parentName": "CombinedModeActionButtonCardsLeftContainer",

```

```

/* Кнопка добавляется в коллекцию элементов родительского элемента. */
"propertyName": "items",
/* Мета-имя добавляемой кнопки. */
"name": "MainContactButton",
/* Свойства, передаваемые в конструктор элемента. */
"values": {
    /* Тип добавляемого элемента – кнопка. */
    "itemType": Terrasoft.ViewItemType.BUTTON,
    /* Привязка заголовка кнопки к локализуемой строке схемы. */
    "caption": {bindTo: "Resources.Strings.OpenPrimaryContactButtonCaption"},
    /* Привязка метода-обработчика нажатия кнопки. */
    "click": {bindTo: "onOpenPrimaryContactClick"},
    /* Стилль отображения кнопки. */
    "style": Terrasoft.controls.ButtonEnums.style.GREEN,
    /* Привязка свойства доступности кнопки. */
    "enabled": {bindTo: "ButtonEnabled"}
}
}
]
};
});

```

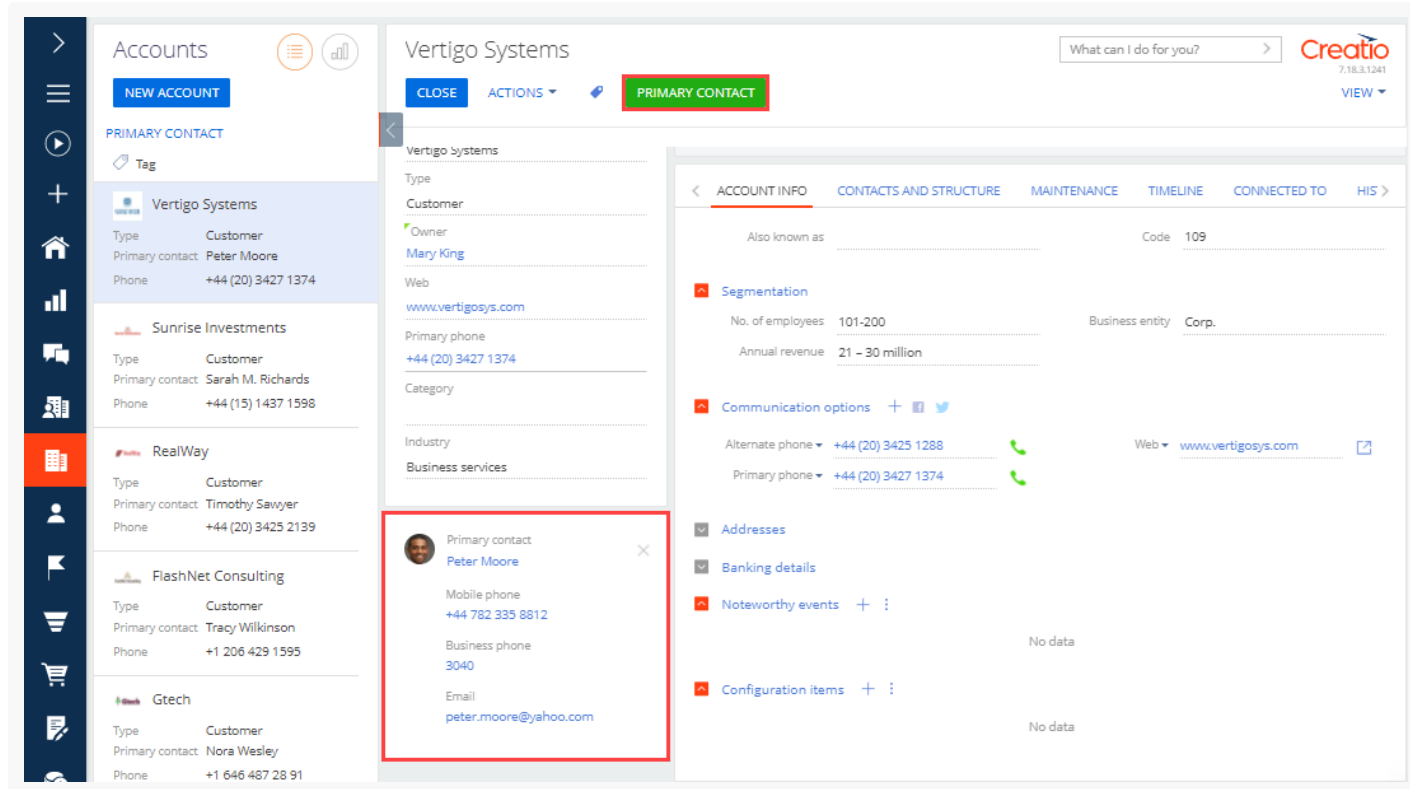
6. На панели инструментов дизайнера нажмите [ *Сохранить* ] ([ *Save* ]).

## Результат выполнения примера

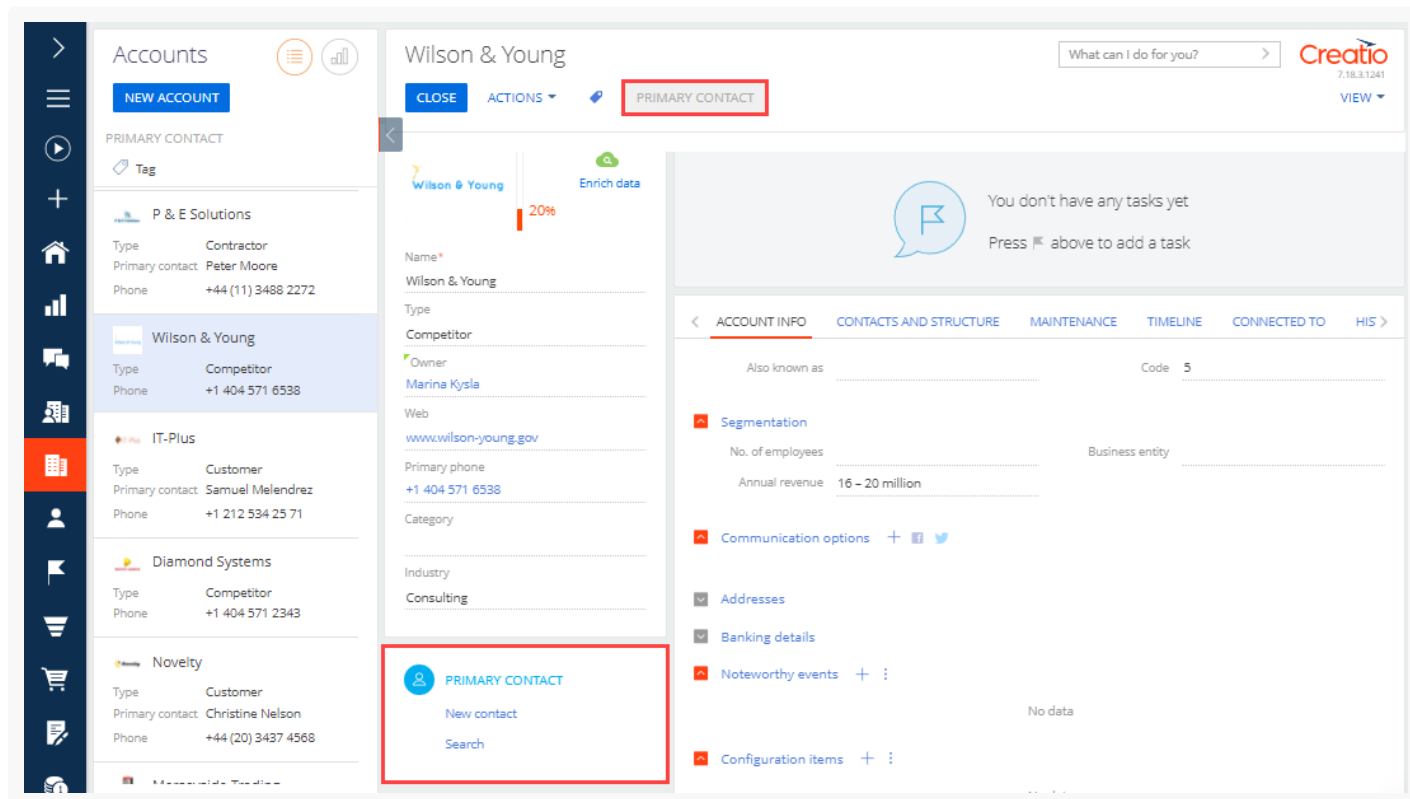
Чтобы **посмотреть результат выполнения примера**, обновите страницу раздела [ *Контрагенты* ] ([ *Accounts* ]).

В результате выполнения примера на панель инструментов раздела [ *Контрагенты* ] ([ *Accounts* ]) добавлена кнопка [ *Основной контакт* ] ([ *Primary contact* ]).

Если для контрагента указан основной контакт, то при нажатии на кнопку [ *Основной контакт* ] ([ *Primary contact* ]) открывается страница этого контакта.



Если для контрагента не указан основной контакт, то кнопка [ *Основной контакт* ] ([ *Primary contact* ]) неактивна.



## Добавить кнопку на панель

# инструментов страницы добавления записи


 Средний

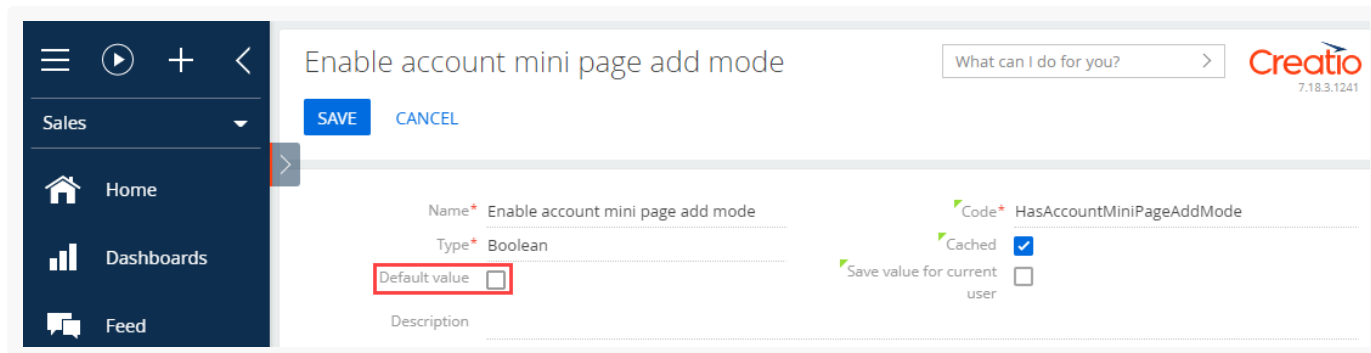
**Пример.** Добавить кнопку на панель инструментов страницы добавления контрагента. Кнопка активна после добавления основного контакта контрагента. При нажатии на кнопку открывается страница основного контакта этого контрагента.

## 1. Изменить способ добавления контрагента


По умолчанию для добавления контрагента используется мини-карточка.

Чтобы для добавления контрагента **использовать страницу добавления**:

1. Перейдите в дизайнер системы по кнопке .
2. В блоке [ *Настройка системы* ] ([ *System setup* ]) перейдите по ссылке [ *Системные настройки* ] ([ *System settings* ]).
3. Выберите настройку [ *Использовать миникарточку добавления контрагента* ] ([ *Enable account mini page add mode* ], код `HasAccountMiniPageAddMode` ).
4. Снимите признак [ *Значение по умолчанию* ] ([ *Default value* ]).



Enable account mini page add mode

What can I do for you? 

SAVE CANCEL

Name\* Enable account mini page add mode

Type\* Boolean

Code\* HasAccountMiniPageAddMode

Cached ☒

Save value for current user ☐

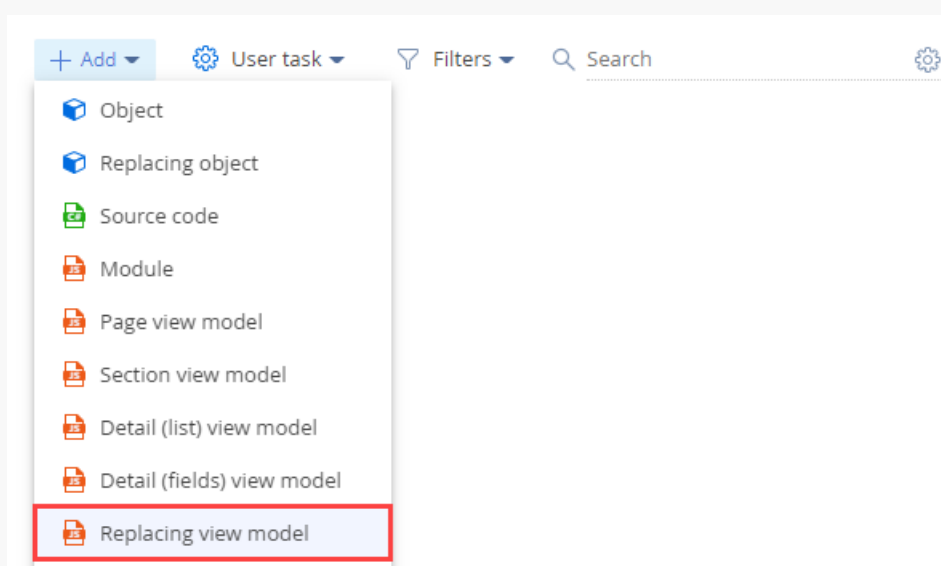
Default value ☐

Description

5. Выполните повторный вход в приложение.

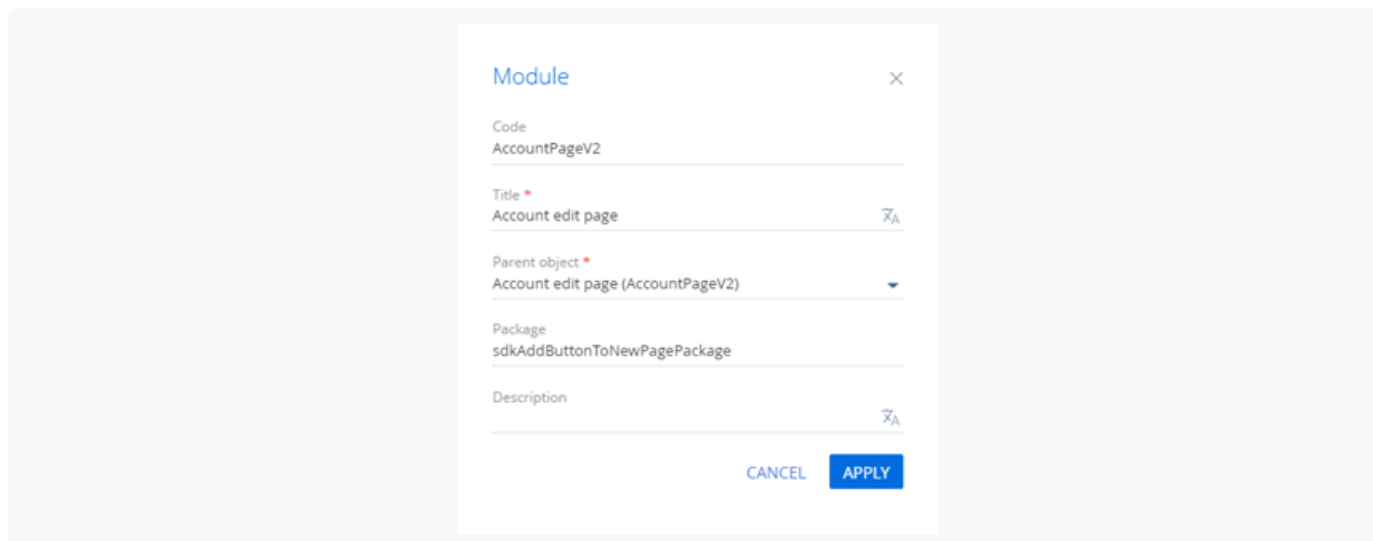
## 2. Создать схему замещающей модели представления страницы контрагента

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* ]).



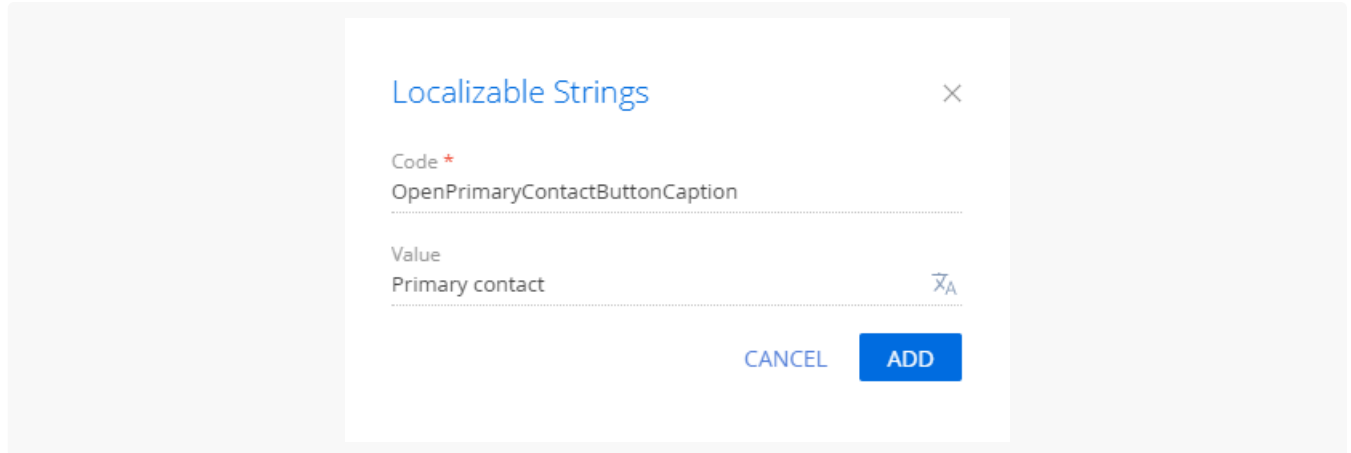
### 3. Заполните **свойства схемы**.

- [ Код ] ([ Code ]) — "AccountPageV2".
- [ Заголовок ] ([ Title ]) — "Страница редактирования контрагента" ("Account edit page").
- [ Родительский объект ] ([ Parent object ]) — выберите "AccountPageV2".



### 4. Добавьте **локализуемую строку**.

- В контекстном меню узла [ Локализуемые строки ] ([ Localizable strings ]) нажмите кнопку **+**.
- Заполните **свойства локализуемой строки**.
  - [ Код ] ([ Code ]) — "OpenPrimaryContactButtonCaption".
  - [ Значение ] ([ Value ]) — "Основной контакт" ("Primary contact").



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

## 5. Реализуйте **логику работы кнопки**.

а. В свойстве `methods` реализуйте **методы**:

- `isAccountPrimaryContactSet()` — проверяет заполнение поля [ *Основной контакт* ] ([ *Primary contact* ]) страницы.
- `onOpenPrimaryContactClick()` — метод-обработчик нажатия кнопки. Выполняет переход на страницу основного контакта.

д. В массив модификаций `diff` добавьте конфигурационный объект с настройками расположения кнопки на странице.

Исходный код схемы замещающей модели представления страницы контрагента представлен ниже.

### AccountPageV2

```
define("AccountPageV2", [], function() {
    return {
        /* Название схемы объекта страницы записи. */
        entitySchemaName: "Account",
        /* Методы модели представления страницы записи. */
        methods: {
            /* Проверяет заполнение поле [Основной контакт] страницы записи. */
            isAccountPrimaryContactSet: function() {
                return this.get("PrimaryContact") ? true : false;
            },
            /* Метод-обработчик нажатия кнопки. */
            onOpenPrimaryContactClick: function() {
                var primaryContactObject = this.get("PrimaryContact");
                if (primaryContactObject) {
                    /* Определение идентификатора основного контакта. */
                    var primaryContactId = primaryContactObject.value;
                    /* Формирование строки адреса. */
                    var requestUrl = "CardModuleV2/ContactPageV2/edit/" + primaryContactId;
                    // Публикация сообщения о пополнении истории навигации по страницам и пер
```

```

        this.sandbox.publish("PushHistoryState", {
            hash: requestUrl
        });
    }
}
},
/* Отображение кнопки на странице записи. */
diff: [
    /* Метаданные для добавления на страницу пользовательской кнопки. */
    {
        /* Выполняется операция добавления элемента на страницу. */
        "operation": "insert",
        /* Мета-имя родительского контейнера, в который добавляется кнопка. */
        "parentName": "LeftContainer",
        /* Кнопка добавляется в коллекцию элементов родительского элемента. */
        "propertyName": "items",
        /* Мета-имя добавляемой кнопки. */
        "name": "PrimaryContactButton",
        /* Свойства, передаваемые в конструктор элемента. */
        "values": {
            /* Тип добавляемого элемента – кнопка. */
            "itemType": Terrasoft.ViewItemType.BUTTON,
            /* Привязка заголовка кнопки к локализуемой строке схемы. */
            "caption": {bindTo: "Resources.Strings.OpenPrimaryContactButtonCaption"},
            /* Привязка метода-обработчика нажатия кнопки. */
            "click": {bindTo: "onOpenPrimaryContactClick"},
            /* Привязка свойства доступности кнопки. */
            "enabled": {bindTo: "isAccountPrimaryContactSet"},
            /* Стиль отображения кнопки. */
            "style": Terrasoft.controls.ButtonEnums.style.BLUE
        }
    }
]
};
});

```

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

## Результат выполнения примера

Чтобы **посмотреть результат выполнения примера**:

1. Обновите страницу раздела [ *Контрагенты* ] ([ *Accounts* ]).
2. На панели инструментов раздела [ *Контрагенты* ] ([ *Accounts* ]) нажмите кнопку [ *Добавить контрагента* ] ([ *New account* ]).

В результате выполнения примера на панель инструментов страницы добавления контрагента добавлена неактивна кнопка [ *Основной контакт* ] ([ *Primary contact* ]).

The screenshot shows the 'New record' page in the Creatio system. The page is designed for adding a new contact. On the left, there is a dark blue sidebar with various icons. The main content area is titled 'New record' and includes a search bar at the top right with the placeholder text 'What can I do for you?'. Below the title, there are buttons for 'SAVE', 'CANCEL', 'ACTIONS', and a 'PRIMARY CONTACT' button, which is highlighted with a red box. The form itself is divided into two columns. The left column contains fields for 'Primary phone', 'Category', and 'Industry'. The right column contains fields for 'No. of employees', 'Business entity', and 'Annual revenue'. Below these fields, there is a section for 'Communication options' with a plus sign and social media icons. This is followed by sections for 'Addresses', 'Banking details', 'Noteworthy events', and 'Configuration items', each with a plus sign and a vertical ellipsis. The bottom of the page shows a 'PRIMARY CONTACT' section with a person icon and the text 'PRIMARY CONTACT'. Below this, there are links for 'New contact' and 'Search'. The bottom right of the page shows 'No data' for some sections.

Кнопка [ *Основной контакт* ] ([ *Primary contact* ]) активируется после указания основного контакта контрагента. При нажатии на кнопку [ *Основной контакт* ] ([ *Primary contact* ]) открывается страница этого контакта.



The screenshot shows the 'New record' page in the Creatio CRM system. The page has a dark blue sidebar with navigation icons. The main content area is titled 'New record' and includes a search bar 'What can I do for you?'. Below the title are buttons: 'SAVE' (green), 'CANCEL' (blue), 'ACTIONS' (dropdown), and 'PRIMARY CONTACT' (blue, highlighted with a red box). The form is divided into two columns. The left column has fields for 'Primary phone', 'Category', and 'Industry'. The right column has a section for 'Annual revenue' and a list of expandable sections: 'Communication options' (with social media icons), 'Addresses', 'Banking details', 'Noteworthy events', and 'Configuration items'. A modal window is open, showing details for 'James Smith', including his job title 'Senior Account Manager', mobile phone '+44 (787) 121 4006', business phone '+44 (15) 1432 4926', and email 'smith@gateway-invest.co.uk'. The modal is also highlighted with a red box.

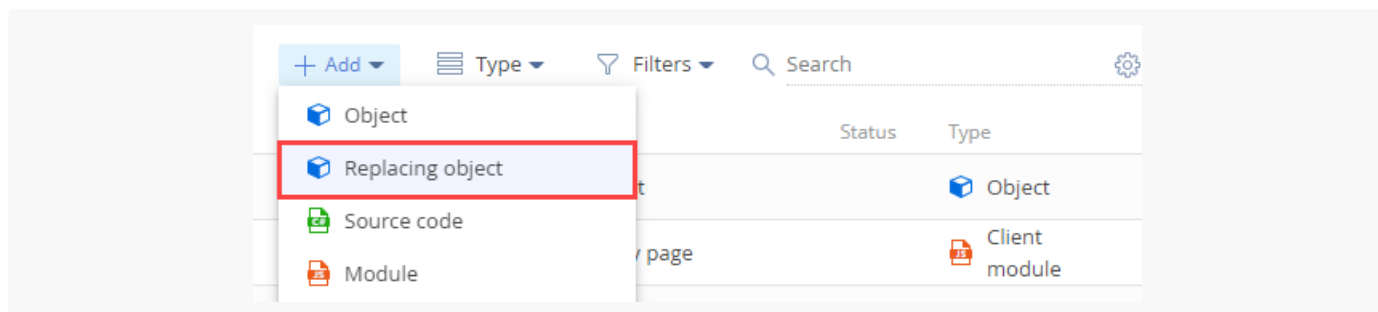
## Добавить кнопку выбора цвета на страницу записи

 Средний

**Пример.** Добавить кнопку выбора цвета на страницу продукта.

### 1. Создать схему замещающего объекта

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

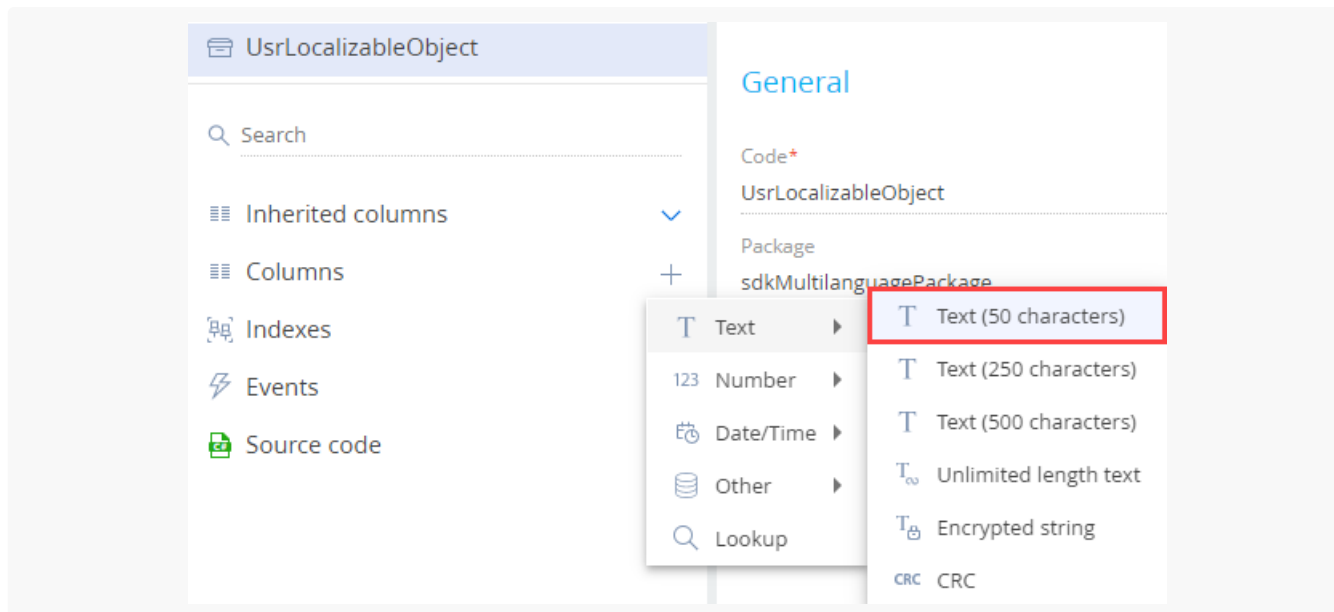


### 3. Заполните **свойства схемы**.

- [ Код ] ([ Code ]) — "Product".
- [ Заголовок ] ([ Title ]) — "Продукт" ("Product").
- [ Родительский объект ] ([ Parent object ]) — выберите "Product".

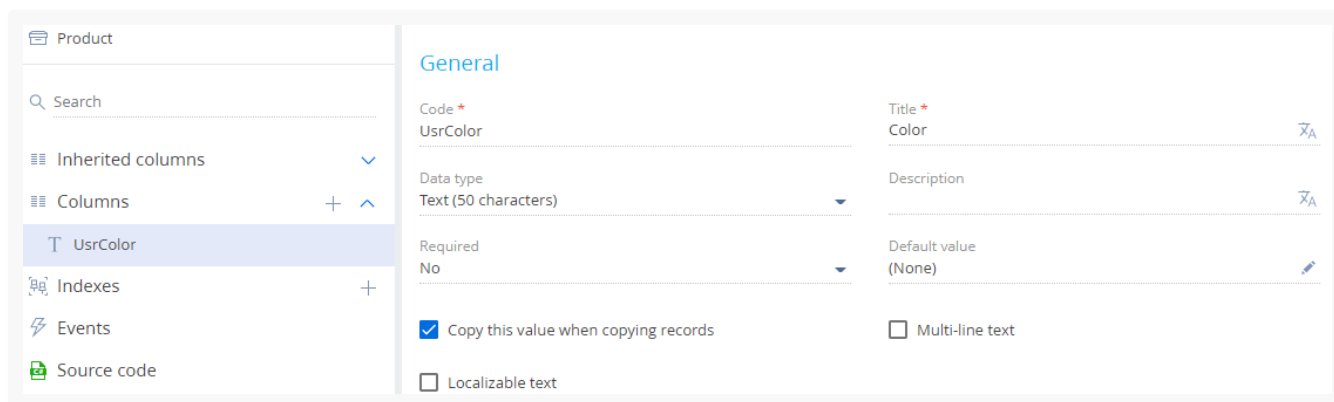
### 4. В схему добавьте **колонку**.

- В контекстном меню узла [ Колонки ] ([ Columns ]) структуры объекта нажмите **+**.
- В выпадающем меню нажмите [ Строка ] —> [ Строка (50 символов) ] ([ Text ] —> [ Text (50 characters) ]).



с. Заполните **свойства добавляемой колонки**.

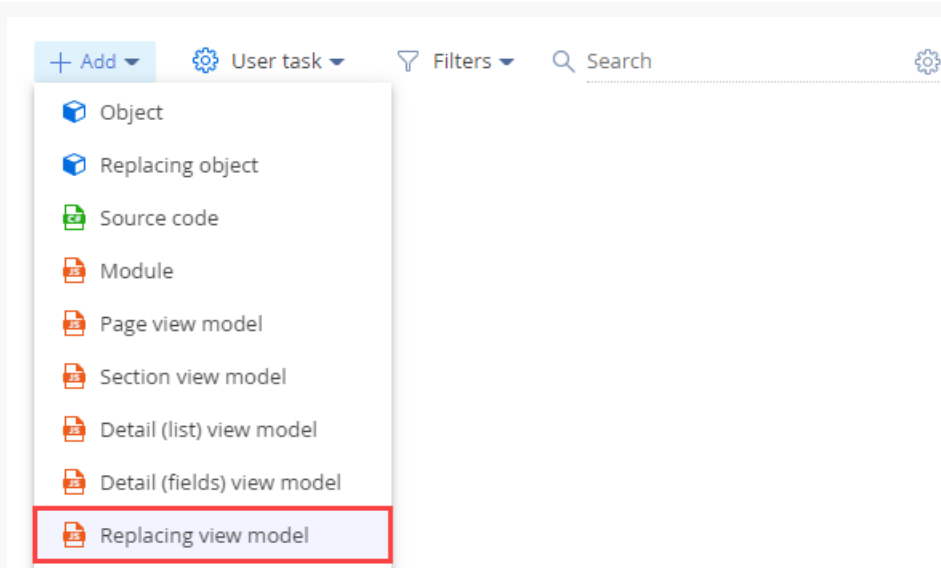
- [ Код ] ([ Code ]) — "UsrColor".
- [ Заголовок ] ([ Title ]) — "Цвет" ("Color").



5. На панели инструментов дизайнера объектов нажмите [ Сохранить ] ([ Save ]), а затем [ Опубликовать ] ([ Publish ]).

## 2. Создать схему замещающей модели представления страницы контрагента

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 ]) — "ProductPageV2".
- [ Заголовок ] ([ Title ]) — "Страница редактирования продукта" ("Edit page - Product").
- [ Родительский объект ] ([ Parent object ]) — выберите "ProductPageV2".

### 4. Настройте **расположение кнопки**. Для этого в массив модификаций `diff` добавьте конфигурационный объект с настройками расположения кнопки на странице.

Исходный код схемы замещающей модели представления страницы продукта представлен ниже.

#### ProductPageV2

```
define("ProductPageV2", [], function() {
    return {
        /* Название схемы объекта страницы записи. */
        entitySchemaName: "Product",
        /* Отображение кнопки на странице записи. */
```

```

diff: /**SCHEMA_DIFF*/[
    /* Кнопка выбора цвета. */
    {
        /* Выполняется операция добавления элемента на страницу. */
        "operation": "insert",
        /* Мета-имя родительского контейнера, в который добавляется кнопка. */
        "parentName": "ProductGeneralInfoBlock",
        /* Кнопка добавляется в коллекцию элементов родительского элемента. */
        "propertyName": "items",
        /* Мета-имя добавляемой кнопки. */
        "name": "ColorButton",
        /* Свойства, передаваемые в конструктор элемента. */
        "values": {
            /* Тип добавляемого элемента – кнопка выбора цвета. */
            "itemType": this.Terrasoft.ViewItemType.COLOR_BUTTON,
            /* Привязка значения элемента управления к колонке модели представления.
            "value": { "bindTo": "UsrColor" },
            /* Настройка расположения кнопки. */
            "layout": {
                /* Номер столбца. */
                "column": 5,
                /* Номер строки. */
                "row": 6,
                /* Диапазон занимаемых столбцов. */
                "colSpan": 12
            }
        }
    }
]/**SCHEMA_DIFF*/
};
});

```

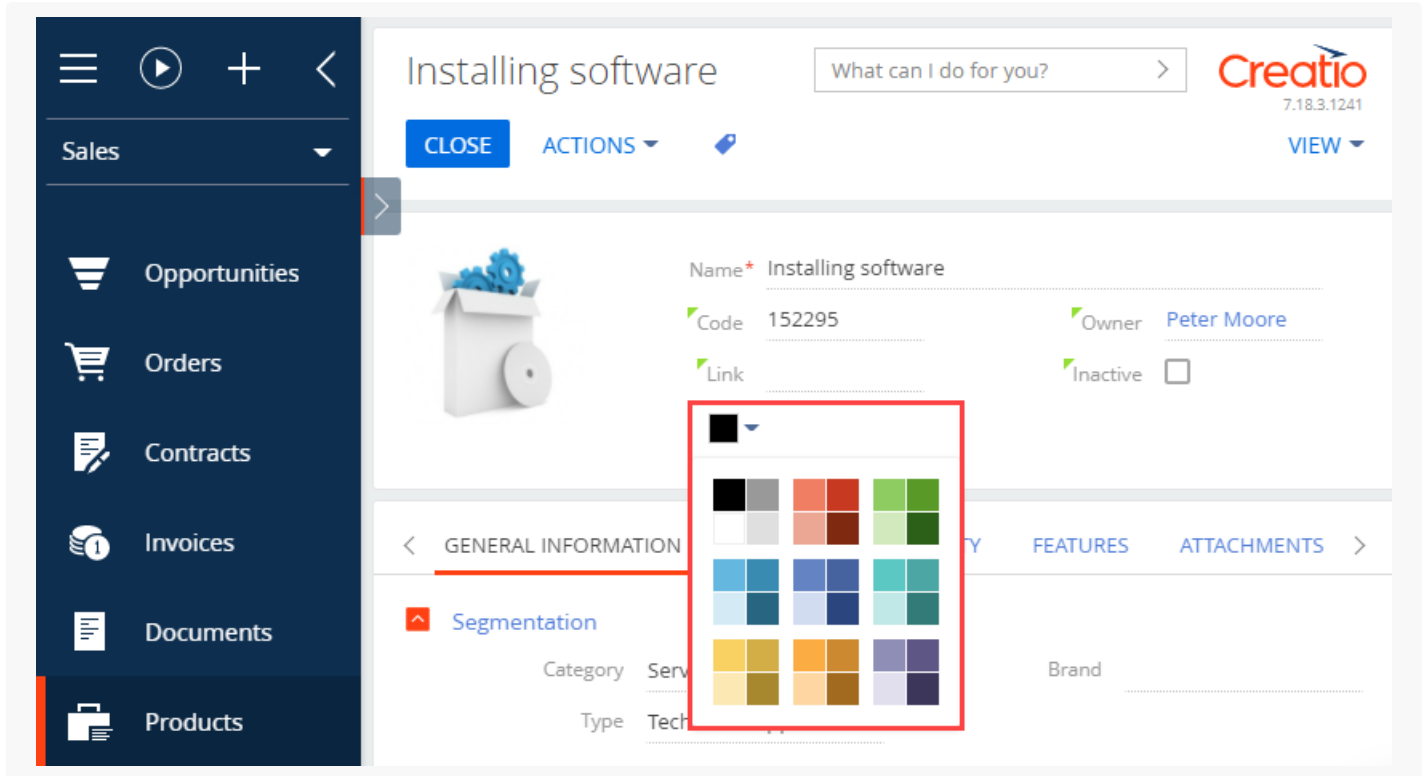
5. На панели инструментов дизайнера нажмите [ *Сохранить* ] ([ *Save* ]).

## Результат выполнения примера

Чтобы **посмотреть результат выполнения примера**:

1. Обновите страницу раздела [ *Продукты* ] ([ *Products* ]).
2. Откройте страницу продукта.

В результате выполнения примера на страницу продукта добавлена кнопка выбора цвета.



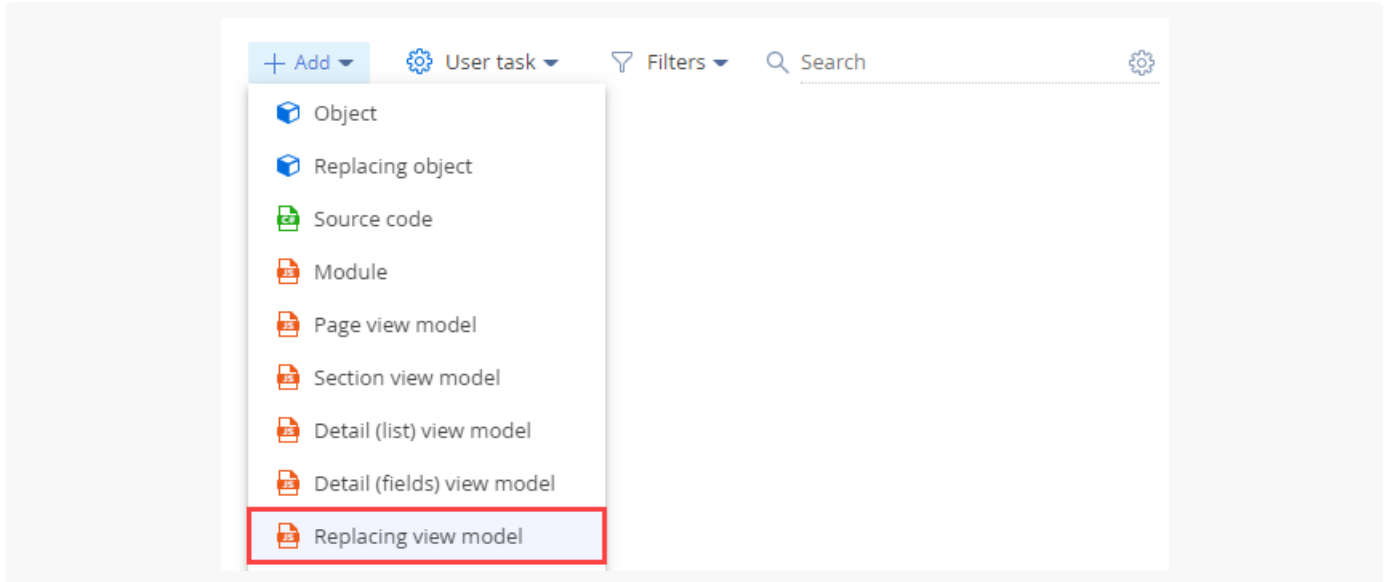
## Добавить к кнопке всплывающую подсказку

 Сложный

**Пример.** Добавить всплывающую подсказку к кнопке [ Сохранить ] ([ Save ]) страницы контакта.

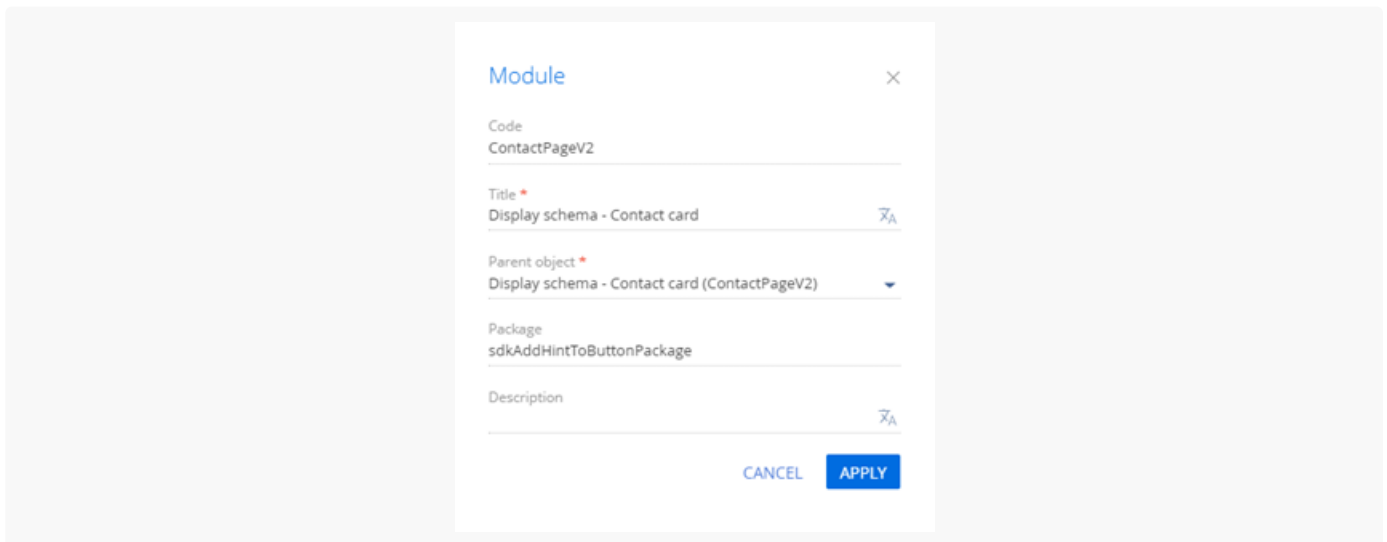
## Создать схему замещающей модели представления страницы контакта

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 ]).



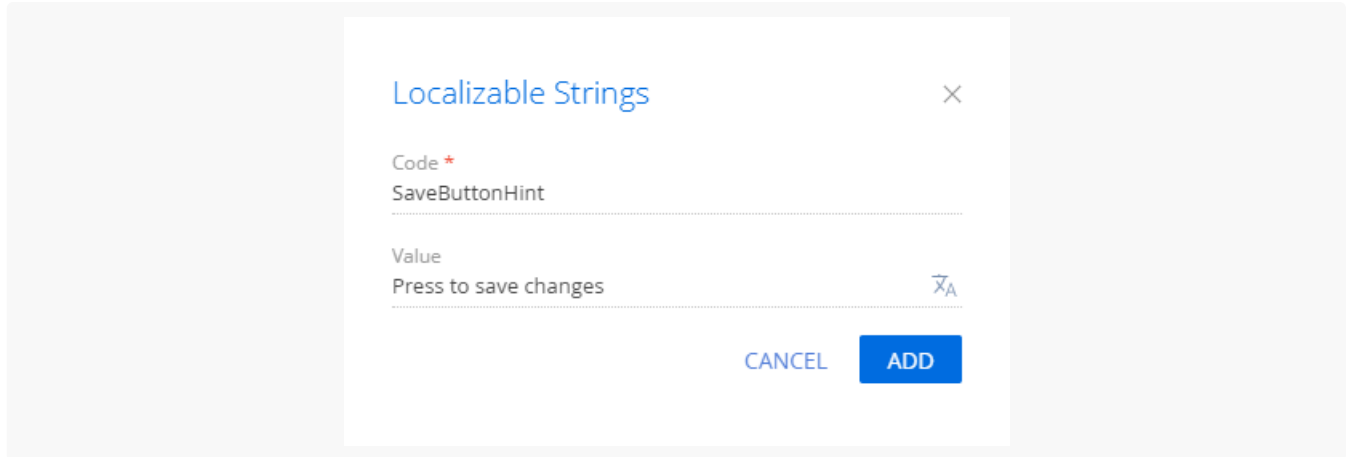
### 3. Заполните **свойства схемы**.

- [ Код ] ([ Code ]) — "ContactPageV2".
- [ Заголовок ] ([ Title ]) — "Схема отображения карточки контакта" ("Display schema – Contact card").
- [ Родительский объект ] ([ Parent object ]) — выберите "ContactPageV2".



### 4. Добавьте **локализуемую строку**, которая содержит текст подсказки.

- В контекстном меню узла [ Локализуемые строки ] ([ Localizable strings ]) нажмите кнопку **+**.
- Заполните **свойства локализуемой строки**.
  - [ Код ] ([ Code ]) — "SaveButtonHint".
  - [ Значение ] ([ Value ]) — "Нажмите, чтобы сохранить изменения" ("Press to save changes").



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

5. Настройте **всплывающую подсказку к кнопке** [ *Сохранить* ] ([ *Save* ]) страницы контакта. Для этого в массив модификаций `diff` добавьте конфигурационный объект кнопки на странице.

Исходный код схемы замещающей модели представления страницы контакта представлен ниже. Используются разные способы добавления всплывающей подсказки в конфигурационный объект кнопки.

#### ContactPageV2 (вариант 1)

```
define("ContactPageV2", [], function () {
    return {
        /* Название схемы объекта страницы записи. */
        entitySchemaName: "Contact",
        /* Отображение всплывающей подсказки. */
        diff: /**SCHEMA_DIFF*/[
            /* Метаданные для добавления к кнопке всплывающей подсказки. */
            {
                /* Выполняется операция изменения существующего элемента. */
                "operation": "merge",
                /* Мета-имя родительского контейнера, в котором изменяется кнопка. */
                "parentName": "LeftContainer",
                /* Кнопка изменяется в коллекции элементов родительского элемента. */
                "propertyName": "items",
                /* Мета-имя изменяемой кнопки. */
                "name": "SaveButton",
                /* Свойства, передаваемые в конструктор элемента. */
                "values": {
                    /* Всплывающая подсказка для кнопки. */
                    "hint": { "bindTo": "Resources.Strings.SaveButtonHint" }
                }
            }
        ]/**SCHEMA_DIFF*/
    };
});
```



## ContactPageV2 (вариант 2)

```

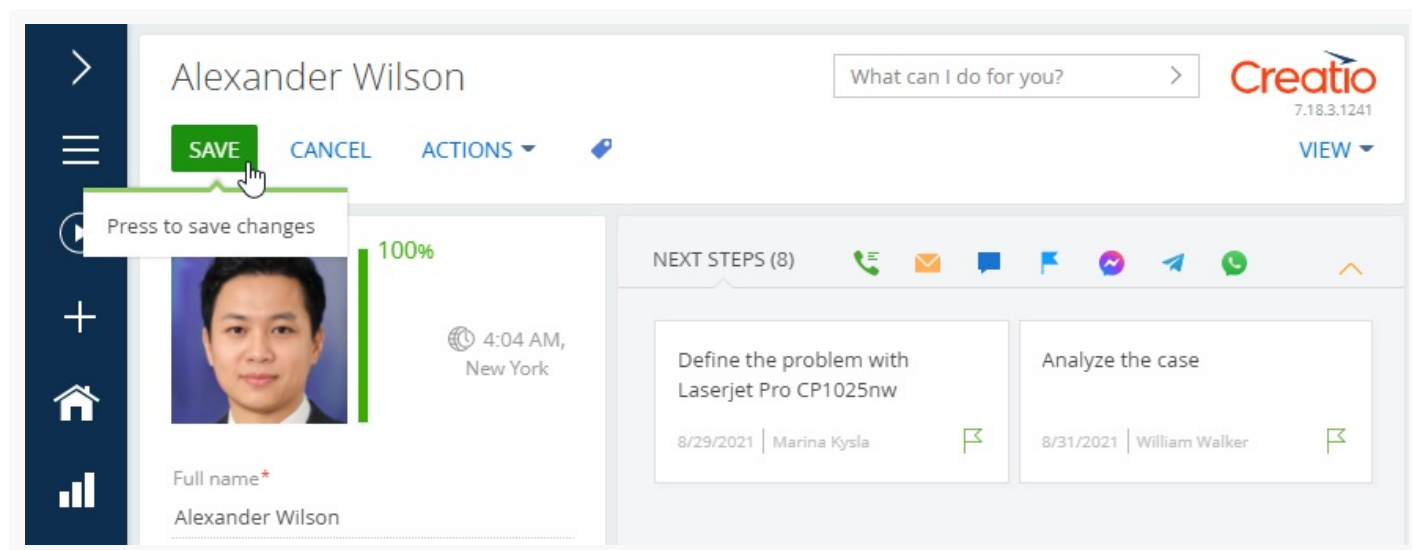
define("ContactPageV2", [], function () {
    return {
        /* Название схемы объекта страницы записи. */
        entitySchemaName: "Contact",
        /* Отображение всплывающей подсказки. */
        diff: /**SCHEMA_DIFF*/[
            /* Метаданные для добавления к кнопке всплывающей подсказки. */
            {
                /* Выполняется операция изменения существующего элемента. */
                "operation": "merge",
                /* Мета-имя родительского контейнера, в котором изменяется кнопка. */
                "parentName": "LeftContainer",
                /* Кнопка изменяется в коллекции элементов родительского элемента. */
                "propertyName": "items",
                /* Мета-имя изменяемой кнопки. */
                "name": "SaveButton",
                /* Свойства, передаваемые в конструктор элемента. */
                "values": {
                    /* Массив подсказок для кнопки. */
                    "tips": []
                }
            },
            /* Конфигурационный объект простой подсказки. */
            {
                /* Выполняется операция добавления элемента. */
                "operation": "insert",
                /* Мета-имя добавляемой кнопки. */
                "parentName": "SaveButton",
                /* Подсказка добавляется в коллекцию элементов родительского элемента. */
                "propertyName": "tips",
                /* Мета-имя добавляемой подсказки. */
                "name": "CustomShowedTip",
                /* Свойства, передаваемые в конструктор элемента. */
                "values": {
                    /* Текст подсказки. */
                    "content": {"bindTo": "Resources.Strings.SaveButtonHint"}
                    /* Здесь можно дополнительно настроить другие параметры отображения и раб
                }
            },
        ]/**SCHEMA_DIFF*/
    };
});

```

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

## Результат выполнения примера

В результате выполнения примера к кнопке [ Сохранить ] ([ Save ]) страницы контакта добавлена всплывающая подсказка.



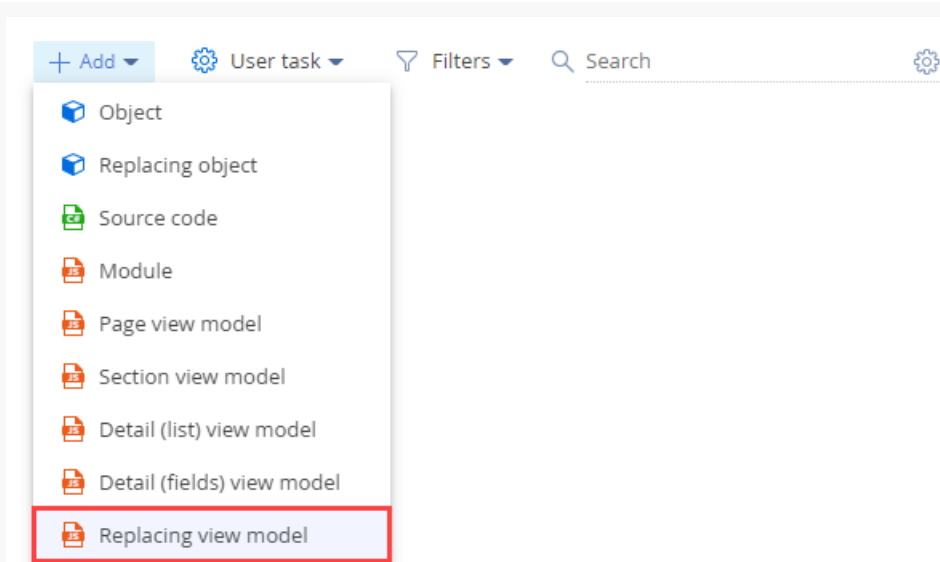
## Добавить кнопку в строку записи раздела

 Средний

**Пример.** Добавить кнопку [ Показать возраст ] ([ Show age ]) в строку активной записи раздела [ Контакты ] ([ Contacts ]). При нажатии на кнопку во всплывающем окне будет отображаться возраст выбранного контакта.

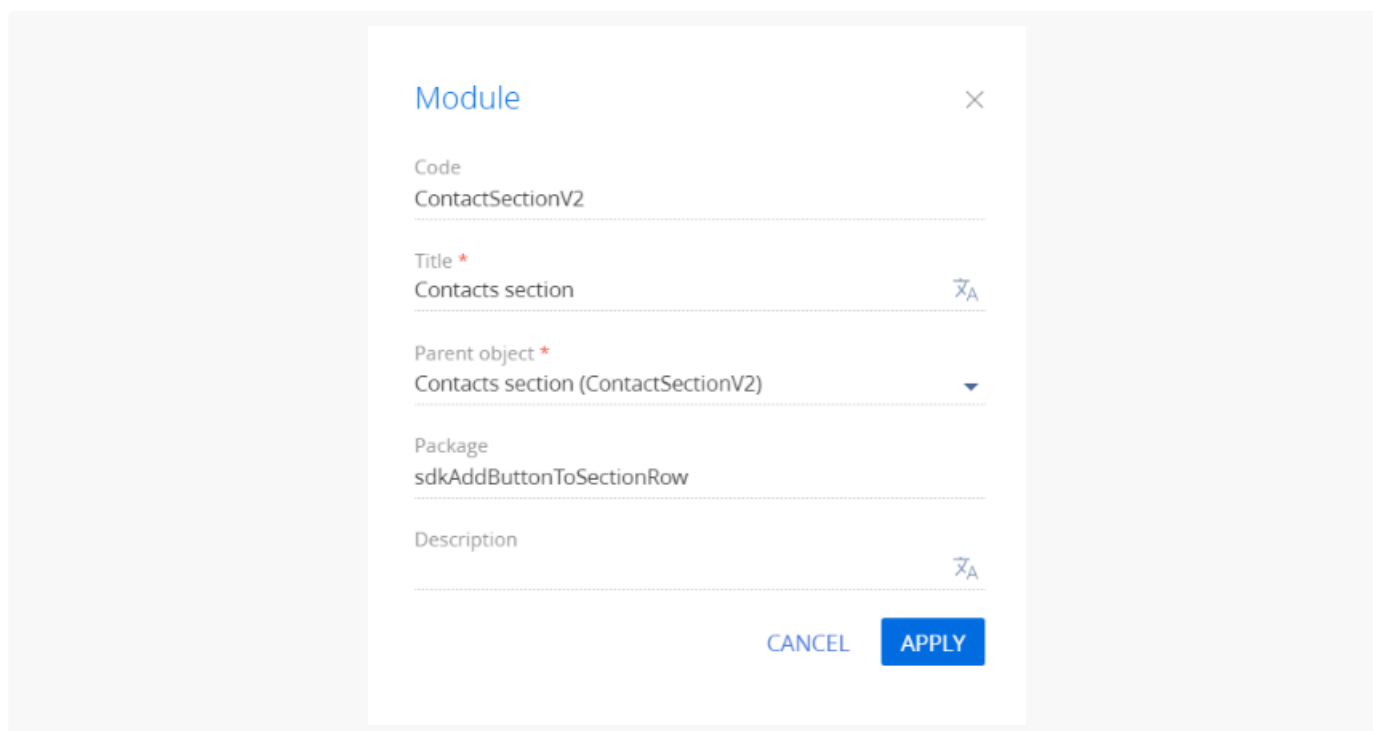
## Создать схему замещающей модели представления раздела

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 ]).



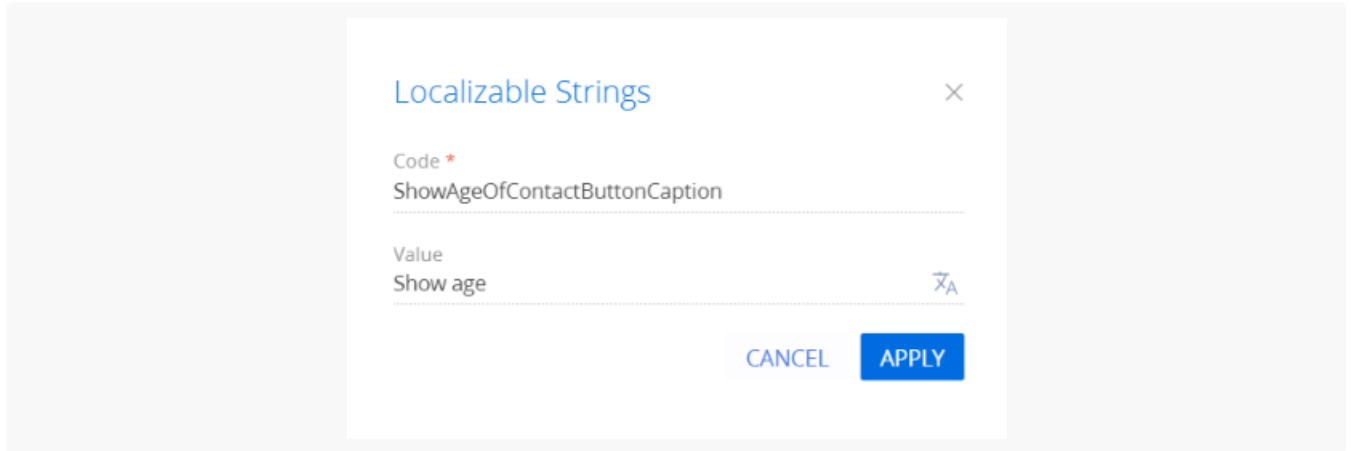
### 3. Заполните **свойства схемы**.

- [ Код ] ([ Code ]) — "ContactSectionV2".
- [ Заголовок ] ([ Title ]) — "Раздел контакты" ("Contact section").
- [ Родительский объект ] ([ Parent object ]) — выберите "ContactSectionV2".



### 4. Добавьте **локализуемую строку**.

- В контекстном меню узла [ Локализуемые строки ] ([ Localizable strings ]) нажмите кнопку **+**.
- Заполните **свойства локализуемой строки**.
  - [ Код ] ([ Code ]) — "ShowAgeOfContactButtonCaption".
  - [ Значение ] ([ Value ]) — "Показать возраст" ("Show age").



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

## 5. Реализуйте **логику работы кнопки**.

а. В свойстве `methods` реализуйте методы:

- `onActiveRowAction()` — присваивает метод-обработчик кнопке, расположенной в строке активной записи раздела.
- `onShowAgeButtonClicked()` — метод-обработчик нажатия кнопки. Возвращает возраст контакта во всплывающем окне.

д. В массив модификаций `diff` добавьте конфигурационный объект с настройками расположения кнопки на странице.

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 страницы раздела представлен ниже.

### ContactSectionV2

```
define("ContactSectionV2", ["ContactSectionV2Resources"], function (resources) {
    return {
        /* Название схемы объекта раздела. */
        entitySchemaName: "Contact",
        /* Методы модели представления раздела. */
        methods: {
            onActiveRowAction: function (buttonTag) {
                switch (buttonTag) {
                    case "showAgeButton":
                        this.onShowAgeButtonClicked();
                        break;
                    default:
                        this.callParent(arguments);
                        break;
                }
            }
        }
    };
});
```

```

    }
  },
  /* Метод-обработчик нажатия кнопки. */
  onShowAgeButtonClicked: function () {
    var message = "";
    var activeRow = this.getActiveRow();
    var recordId = activeRow.get("Id");
    /* Создаем экземпляр класса Terrasoft.EntitySchemaQuery с корневой схемой Con
    var esq = this.Ext.create("Terrasoft.EntitySchemaQuery", {
      rootSchemaName: "Contact"
    });
    /* Добавление колонки с возрастом. */
    esq.addColumn("Age", "Age");
    /* Получение записи из выборки по Id объекта. */
    esq.getEntity(recordId, function(result) {
      if (!result.success) {
        this.showInformationDialog("Error");
        return;
      }
      message += "Age of contact is " + result.entity.get("Age");
      this.showInformationDialog(message);
    }, this);
  },
  /* Отображение кнопки в разделе. */
  diff: /**SCHEMA_DIFF*/[
    /* Метаданные для добавления в раздел пользовательской кнопки. */
    {
      /* Выполняется операция добавления элемента на страницу. */
      "operation": "insert",
      /* Мета-имя добавляемой кнопки. */
      "name": "DataGridActiveRowShowAgeButton",
      /* Мета-имя родительского контейнера, в который добавляется кнопка. */
      "parentName": "DataGrid",
      /* Кнопка добавляется в коллекцию элементов родительского элемента. */
      "propertyName": "activeRowActions",
      /* Свойства, передаваемые в конструктор элемента. */
      "values": {
        "className": "Terrasoft.Button",
        "style": Terrasoft.controls.ButtonEnums.style.GREEN,
        /* Привязка заголовка кнопки к локализуемой строке схемы. */
        "caption": resources.localizableStrings.ShowAgeOfContactButtonCaption,
        "tag": "showAgeButton"
      }
    }
  ]/**SCHEMA_DIFF*/
};
});

```

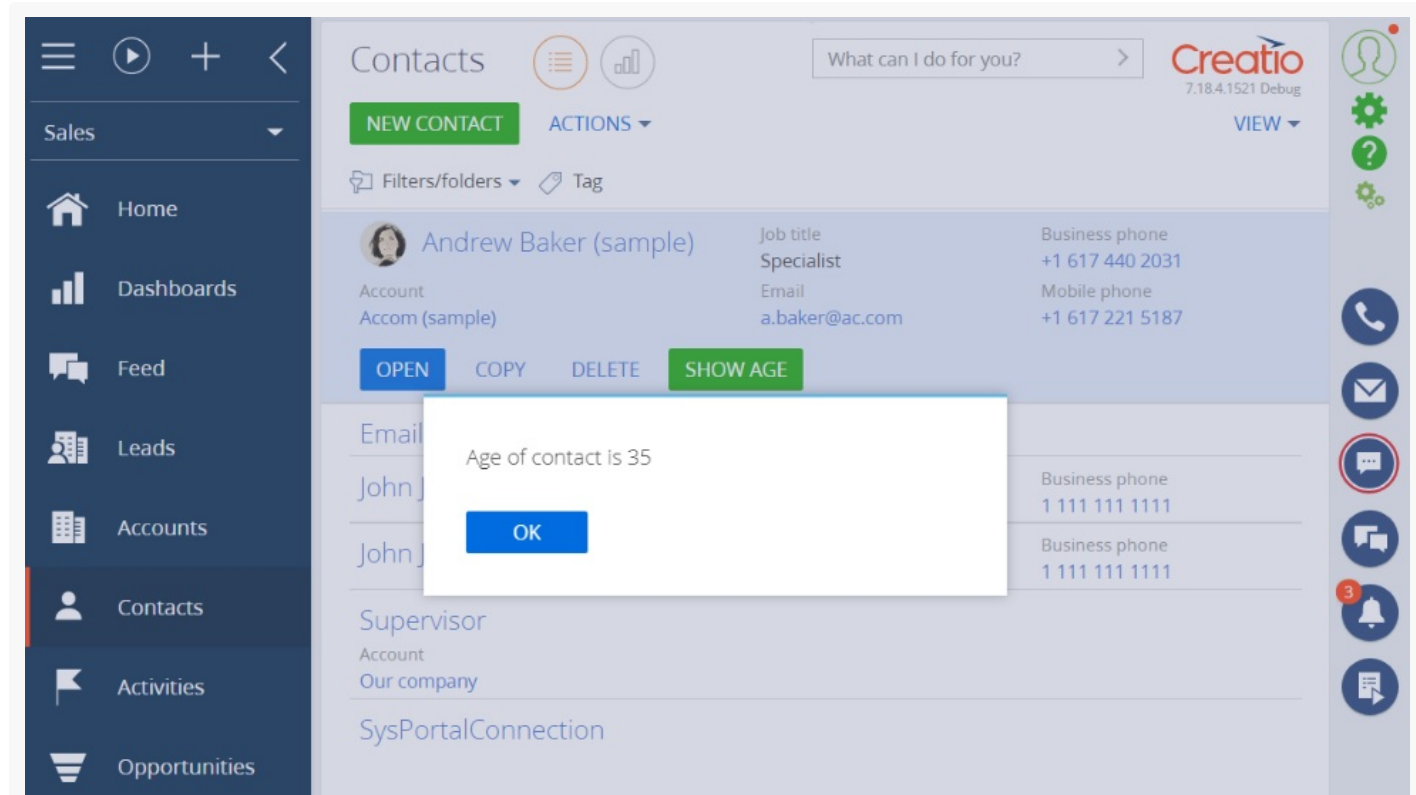
6. На панели инструментов дизайнера нажмите [ Сохранить ] ([ Save ]).

## Результат выполнения примера

Чтобы **посмотреть результат выполнения примера**:

1. Очистите кэш браузера.
2. Обновите страницу раздела [ Контакты ] ([ Contacts ]).

В результате выполнения примера в строку активной записи раздела [ Контакты ] ([ Contacts ]) добавлена кнопка [ Показать возраст ] ([ Show Age ]). При нажатии на кнопку [ Показать возраст ] ([ Show Age ]) отображается всплывающее окно с информацией о возрасте контакта.



## Свойство diff объекта кнопки JS

 Легкий

**Назначение** массива модификаций `diff` объекта кнопки — настройка расположения кнопки на странице записи путем использования конфигурационного объекта. Настройка выполняется в схеме модели представления страницы записи. Описание свойств схемы содержится в статье [Клиентская схема](#).

## Свойства

operation

Операция с кнопкой.

#### Возможные значения

set	Значение кнопки устанавливается значением параметра <code>values</code> .
merge	Значения из родительских, замещаемых и замещающих схем сливаются вместе, при этом свойства из значения параметра <code>values</code> последнего наследника имеют приоритет.
remove	Кнопка удаляется из схемы.
move	Кнопка перемещается в другой родительский элемент.
insert	Кнопка добавляется в схему.

parentName

Мета-имя родительского элемента управления, в который помещается кнопка. Если это функциональная кнопка, то в качестве родительских контейнеров могут выступать `LeftContainer` и `RightContainer`.

propertyName

Имя параметра родительского элемента. Для кнопки указывается значение `items`.

name

Мета-имя добавляемой кнопки.

values

Конфигурационный объект с настройками дополнительных свойств кнопки.

#### Свойства конфигурационного объекта

itemType	Тип элемента. Задается значением перечисления <code>Terrasoft.ViewItemType</code> . Для кнопки используется значение <code>BUTTON</code> .
caption	Заголовок кнопки. Рекомендуется задавать значения заголовков через привязку к локализуемой строке схемы.
click	Привязка метода-обработчика кнопки.
layout	Объект настроек расположения кнопки в сетке.
enabled	Регулирует доступность (активность) кнопки.
visible	Регулирует видимость кнопки.
style	<p>Стиль компонента. Задается значением перечисления <code>Terrasoft.controls.ButtonEnums.style</code>.</p> <p><b>Возможные значения</b> (<code>Terrasoft.controls.ButtonEnums.style</code>)</p> <hr/> <p><b>DEFAULT</b></p> <p>Стиль по умолчанию.</p> <hr/> <p><b>GREEN</b></p> <p>Цвет кнопки — зеленый.</p> <hr/> <p><b>RED</b></p> <p>Цвет кнопки — красный.</p> <hr/> <p><b>BLUE</b></p> <p>Цвет кнопки — синий.</p> <hr/> <p><b>GREY</b></p> <p>Цвет кнопки — серый.</p> <hr/> <p><b>TRANSPARENT</b></p> <p>Прозрачная кнопка. Значение из предыдущих версий Creatio.</p>