

# Виды UI элементов и доступные атрибуты

Ниже представлен список возможных UI элементов и доступные для них атрибуты:

- Если атрибут необязательный, после его имени стоит знак вопроса.
- Если в значении атрибута указан текст с разделителем в виде вертикальной черты, это значит, что атрибут может быть равен одному из значений.
- Как работают действия при изменении описано ниже.

## Элемент "string"

```
<element
  name="ExampleField"
  title="Текстовое поле"
  type="string"
  value="Текстовое значение"
  width="6"
  readonly?="true|false"
  invisible?="true|false"
  onChangeAction?="Действие при изменении" />
```

## Элемент "password"

```
<element
  name="ExampleField"
  title="Пароль"
  type="password"
  value=""
  width="6"
  readonly?="true|false"
  invisible?="true|false"
  onChangeAction?="Действие при изменении" />
```

Примечание:

- данные в инпут передаются не зашифрованными, просто отображение для пользователя с закрытыми символами.

## Элемент "multistring"

```
<element
  name="ExampleField"
  title="Многострочное текстовое поле"
  type="string"
  value="Текстовое значение"
  width="6"
  rows?="24"                                <!-- высота элемента в строках -->
  readonly?="true|false"
  invisible?="true|false"
  onChangeAction?="Действие при изменении" />
```

## Элемент "int"

```
<element
  name="ExampleField"
  title="Числовое поле"
  type="int"
  value="0"
  width="6"
  readonly?="true|false"
  invisible?="true|false"
  onChangeAction?="Действие при изменении" />
```

## Элемент "double"

```
<element
  name="ExampleField"
  title="Числовое с плавающей точкой"
  type="double"
  decimalsLimit?="2"
  value="0"
  width="6"
  readonly?="true|false"
```

```
invisible?="true|false"  
onChangeAction?="Действие при изменении" />
```

### Элемент "money"

```
<element  
  name="ExampleField"  
  title="Денежное поле"  
  type="money"  
  currency="USD|EUR|RUB"  
  value="0"  
  width="6"  
  readonly?="true|false"  
  invisible?="true|false"  
  onChangeAction?="Действие при изменении" />
```

### Элемент "bool"

```
<element  
  name="ExampleField"  
  title="Пример чекбокса"  
  type="bool"  
  value=""  
  width="6"  
  readonly?="true|false"  
  invisible?="true|false"  
  onChangeAction?="Действие при изменении" />
```

### Элемент "date"

```
<element  
  name="ExampleField"  
  title="Поле только с датой"  
  type="date"  
  value=""  
  width="6"  
  readonly?="true|false"  
  invisible?="true|false"  
  onChangeAction?="Действие при изменении" />
```

### Элемент "datetime"

```
<element  
  name="ExampleField"  
  title="Поле с датой и временем"  
  type="datetime"  
  value=""  
  width="6"  
  readonly?="true|false"  
  invisible?="true|false"  
  onChangeAction?="Действие при изменении" />
```

### Элемент "link"

```
<element  
  name="CustomerId"  
  title="Клиент"  
  type="link"  
  tableName="Customers"           <!-- обязательный атрибут -->  
  value="0"                      <!-- Id выбранной записи по умолчанию -->  
  filter?="[IsActive] = 1"       <!-- доп. фильтр для выбора -->  
  width="6"  
  readonly?="true|false"  
  invisible?="true|false"  
  onChangeAction?="Действие при изменении" />
```

### Элемент "dictionary" (выпадающие списки)

```
<element  
  name="Priority"  
  title="Приоритет"
```

```

type="dictionary"
dictionary?="Incident Priority"  <!-- имя словаря -->
value="1"                        <!-- значение по умолчанию -->
width="6"
readonly?="true|false"
invisible?="true|false"
onChangeAction?="Действие при изменении">
  <!-- имя словаря -->
  <option value="1" title="Низкий" icon?="ExclamationTriangle" color?="red|orange|green|blue|violet" />
  <option value="2" title="Средний" icon?="ExclamationTriangle" color?="red|orange|green|blue|violet" />
  <option value="3" title="Высокий" icon?="ExclamationTriangle2" color?="red|orange|green|blue|violet" />
</element>

```

## Примечания:

- Можно указать название системного словаря. Например, dictionary="Incident Priority"
- А также указать дополнительные значения (не обязательно). Например, <option value="-1" title="Нет информации" />
- Список доступных иконок можно посмотреть по адресу [http\(s\)://VERSIVM\\_URL/icons](http(s)://VERSIVM_URL/icons)

## Элемент "html"

```

<element
  name="Info"
  title="Информация"
  type="html"
  width="12"
  readonly?="true|false"
  invisible?="true|false">
  <![CDATA[
    <p>Любой HTML-контент, который будет отображён в мастере.</p>
  ]]>
</element>

```

## Примечание:

- Можно создавать динамические html, внутри которых будут срабатывать подстановки из контекста мастера.
- Пример с подстановкой числового значения:

```

<row>
  <element name="intTest" title="Числовой инпут" type="int" width="3" value="99" />
  <element type="html" width="3" name="htmlTest" readonly="0" title="Test" value="">
  <![CDATA[
    <div>Выбранное число:</div>
    <div>${intTest}</div>
  ]]>
  </element>
</row>

```

- Пример с подстановкой произвольного html:

```

<row>
  <element name="htmlStringTest" title="Введите разметку" type="multistring" width="12" value="" />
  <element type="html" width="3" name="htmlTest" readonly="0" title="Test" value="">
  <![CDATA[
    <div>${htmlStringTest}</div>
  ]]>
  </element>
</row>

```

## .Net код, изменяющий верстку:

```

var html =
  """
  <div>
    <span style="color:red">Текст</span>
  </div>
  """;
wizard.SetValue("htmlStringTest", html);

```

## Элемент "table"

```

<element
  name="OrderItems"
  title="Товары заказа"

```

```

type="table"
selectType="none|singleselect|multiselect" <!-- возможность выбора строк в таблице -->
pageLength?="50" <!-- если не указано, берётся из конфигурации -->
orderBy?="Name" <!-- поле сортировки по умолчанию -->
orderType?="asc|desc"
width="12"
readonly?="true|false"
invisible?="true|false">
<column
  name="Name"
  title="Наименование"
  type="string"
  invisible?="true|false"
  editable?="true|false" />
<column
  name="Price"
  title="Цена"
  type="money"
  currency?="USD|EUR|RUB"
  invisible?="true|false"
  editable?="true|false" />
<column
  name="DoubleField"
  title="Поле с плавающей точкой"
  type="double"
  decimalsLimit?="2"
  invisible?="true|false"
  editable?="true|false" />
<column
  name="Quantity"
  title="Кол-во"
  type="int"
  invisible?="true|false"
  editable?="true|false" />
</element>

```

## Примечания:

- Значение элемента с типом таблица это List<Dictionary<string, object>>.
- Причем, Dictionary каждой строки может содержать больше ключей, чем указано в колонках, но отображаться будут только указанные в списке колонок.

## Элемент "button" (с .Net кодом внутри)

```

<element
  name="BtnSave"
  title="Сохранить"
  type="button"
  buttonVariant?="default|destructive|outline|primary_outline|secondary|ghost|link|info|danger"
  width="3"
  invisible?="true|false" >
  <![CDATA[
    // .Net скрипт, который выполнится по нажатию на кнопку
  ]]>
</element>

```

## Элемент "button" (вызывающая переход на следующую страницу)

```

<element
  name="BtnNext"
  title="Дальше"
  type="button"
  buttonVariant?="default|destructive|outline|primary_outline|secondary|ghost|link|info|danger"
  width="3"
  invisible?="true|false"
  buttonType="transition"
  toPage="page2" />

```

## Примечание:

- Данный вид кнопок добавляется если вам по какой-то причине не подходит использовать кнопки перехода в хедере страницы, и вы хотите добавить кнопки перехода непосредственно на страницу мастера

## Элемент "file"

```
<element
  name="Attachment"
  title="Файл"
  type="file"
  width="6"
  readonly?="true|false"
  invisible?="true|false" />
```

#### Примечания:

- Общие атрибуты для большинства типов: name, title или title.xx (локаль. Например, title.en="English title" ), type, width (целое число, например 6; обычно 1..12), readonly, invisible, value (значение по умолчанию, используется там, где применимо), а также onChangeAction.
- Для type="link" обязателен tableName; можно задать filter.
- Для type="dictionary" можно указать словарь из базы данных через dictionary или задать варианты через дочерние теги option.
- Для type="html" контент берётся из тела элемента.
- Для type="button" при buttonType="transition" обязателен атрибут toPage.

#### Атрибут onChangeAction

- Элементы всех типов могут иметь опциональный атрибут onChangeAction. В нем указывается name элемента типа кнопка, код которого будет вызван при изменении значения в элементе.
- Таким образом, можно реализовать более динамичные интерфейсы. Например, выбрав статус из выпадающего списка, тут же скрылись часть полей на странице мастера.
- Для таблиц onChangeAction срабатывает при выборе строки(ок). Для этого атрибут selectType не должен быть равен "none".
- Важное замечание: для элементов "string", "multistring", а также для таблиц с selectType="multiselect" есть таймер в 1 секунду на прекращение ввода. Чтобы пользователь сначала дописал текст в инпут/выбрал необходимые строки в таблице, а потом уже один раз сработал onChangeAction.

#### Пример использования onChangeAction:

```
<row>
  <element name="SomeTextInput" title="Введите название" type="string" width="3" value="" onChangeAction="OnChangeInput" />
  <element name="OnChangeInput" title="" type="button" width="3" invisible="true">
    <![CDATA[
      // Код, который выполнится при изменении SomeTextInput
      // У кнопки указан invisible="true", так как сама кнопка не требуется в мастере
    ]]>
  </element>
</row>
```