Creatio Academy

Client schema structure

Attributes. The "attributes" property

Version 7.18

This documentation is provided under restrictions on use and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this documentation, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

© 2021 Creatio. All rights reserved.

Table of Contents

© 2021 Creatio. All rights reserved.

Attributes. The "attributes" property

Medium

Introduction

`Attributes` is a configuration object property of the view model schema. It contains configuration objects that describe the model attributes. A model column is the attribute. All object schema columns are included in the `attributes` collection automatically upon generation.

Attribute base properties

The schema attributes have the following base properties:

- `dataValueType` – attribute data type. This property is used for generation of views. The available data types are represented by the `Terrasoft.DataValueType` enumeration.
- `type` – column type. Optional parameter used in the `BaseViewModel` internal work. The available column types are represented by the `Terrasoft.ViewModelColumnType` enumeration.
- `value` – the attribute value. The value of this parameter will be set in the view model at its creation.

Attention. You can specify numeric, string and Boolean values in the `value` attribute.

If the attribute type involves the use of a reference type value (array, object, collection, etc.), its initial value must be initialized using methods.

An example of using attribute base properties:

```
attributes: {
  // Attribute name.
  "NameAttribute": {
    // Data type.
    "dataValueType": this.Terrasoft.DataValueType.TEXT,
    // Column type.
    "type": this.Terrasoft.ViewModelColumnType.VIRTUAL_COLUMN,
    // Default value.
    "value": "NameValue"
  }
}
```

Attributes additional properties

Attributes can have the following properties:

© 2021 Creatio. All rights reserved.

- `Caption` – attribute title.
- `isRequired` – indicates whether the attribute is required.
- `Dependencies` – dependency from another model attribute. For example, setting dependencies of an attribute on the value of another attribute. The property is used to create calculated fields. More information about the calculated fields and the uses of this parameter can be found in the "Adding calculated fields" article.
- `lookupListConfig` – property that configures lookup field features. More information about this parameter can be found in the "Using filtration for lookup fields. Examples" article. This is a configuration object that can contain the following optional properties:
    - columns – an array of column names that will be added to the query with the Id column and the primary display column.
    - orders – an array of configuration objects that determine the sorting of displayed data.
    - filter – a method that returns an object of `Terrasoft.BaseFilter` class or its inheritor that will be applied to the query. Can not be used with the “filters” property.
    - filters – filters array (methods that return collections of the `Terrasoft.FilterGroup` class). Can not be used with the filter property.

An example of using attribute additional properties:

```
attributes: {
  // Attribute name.
  "Client": {
    // Attribute header.
    "caption": { "bindTo": "Resources.Strings.Client" },
    // Attribute  is required.
    "isRequired": true
  },

  // Attribute name.
  "ResponsibleDepartment": {
    lookupListConfig: {
      // Additional columns.
      columns: [ "SalesDirector" ],
      // Sort column.
      orders: [ { columnPath: "FromBaseCurrency" } ],
      // Filter definition function.
      filter: function()
      {
        // Returns filter of Type column, which is equal the "Competitor" constant.
        return this.Terrasoft.createColumnFilterWithParameter(
        this.Terrasoft.ComparisonType.EQUAL,
        "Type",
        ConfigurationConstants.AccountType.Competitor);
      }
    }
  },
  // Attribute name.
  "Probability": {
    // Determination of the column dependency.
    "dependencies": [
      {
        // Depends on the "Stage" column.
        "columns": [ "Stage" ],
        // The name of the handler method for the "Stage" column change.
        // setProbabilityByStage() method is defined  in methods property
        // of schema object.
        "methodName": "setProbabilityByStage"
      }
    ]
  }
},
methods: {

  // "Stage" column modification handler method
  setProbabilityByStage: function()
  {
    // Getting the Stage column value.
    var stage = this.get("Stage");
    // The condition for the "Probability" column modification.
    if (stage.value && stage.value ===
        ConfigurationConstants.Opportunity.Stage.RejectedByUs)
    {
      // Setting the "Probability" column value.
      this.set("Probability", 0);
    }
  }
}
```

© 2021 Creatio. All rights reserved.