



Conditional Visibility in Forms

Data Aquarium Framework supports declarative conditional visibility of fields and field categories in form views.

CONDITIONAL VISIBILITY OF FIELDS

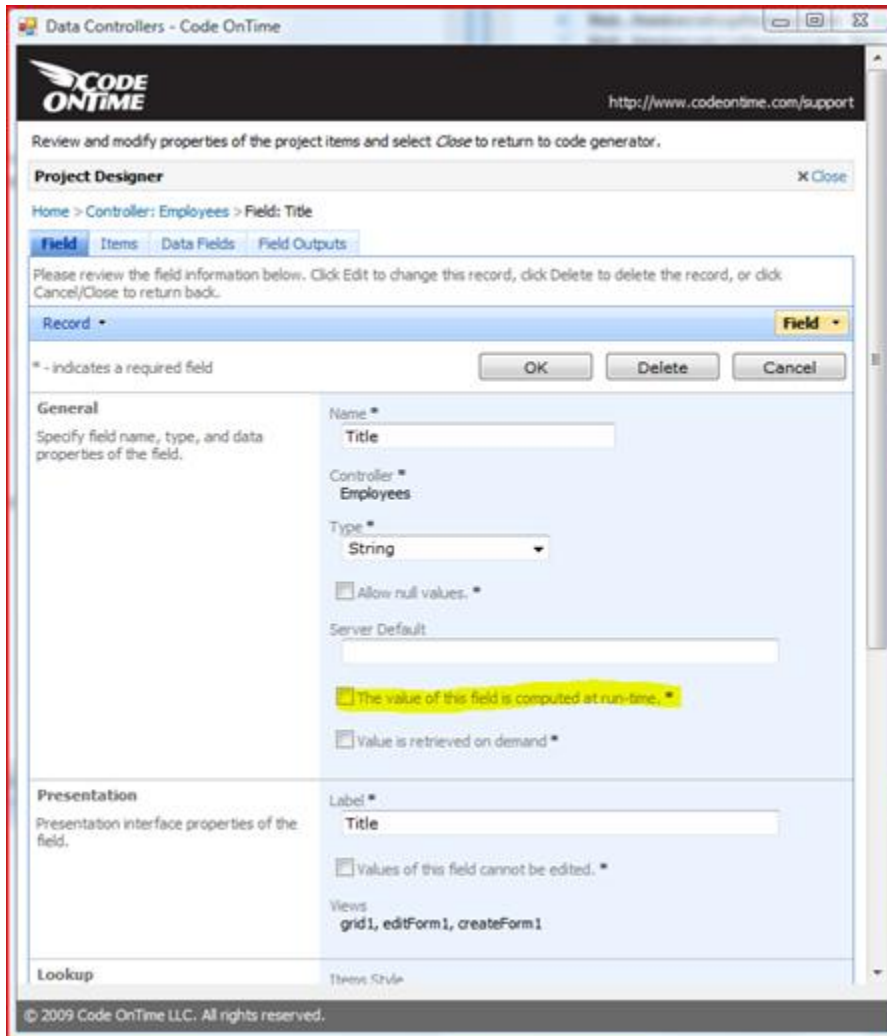
Here is a snippet of the data field controller used in Code OnTime Designer that controls visibility of the Formula field if the field is marked as computed.

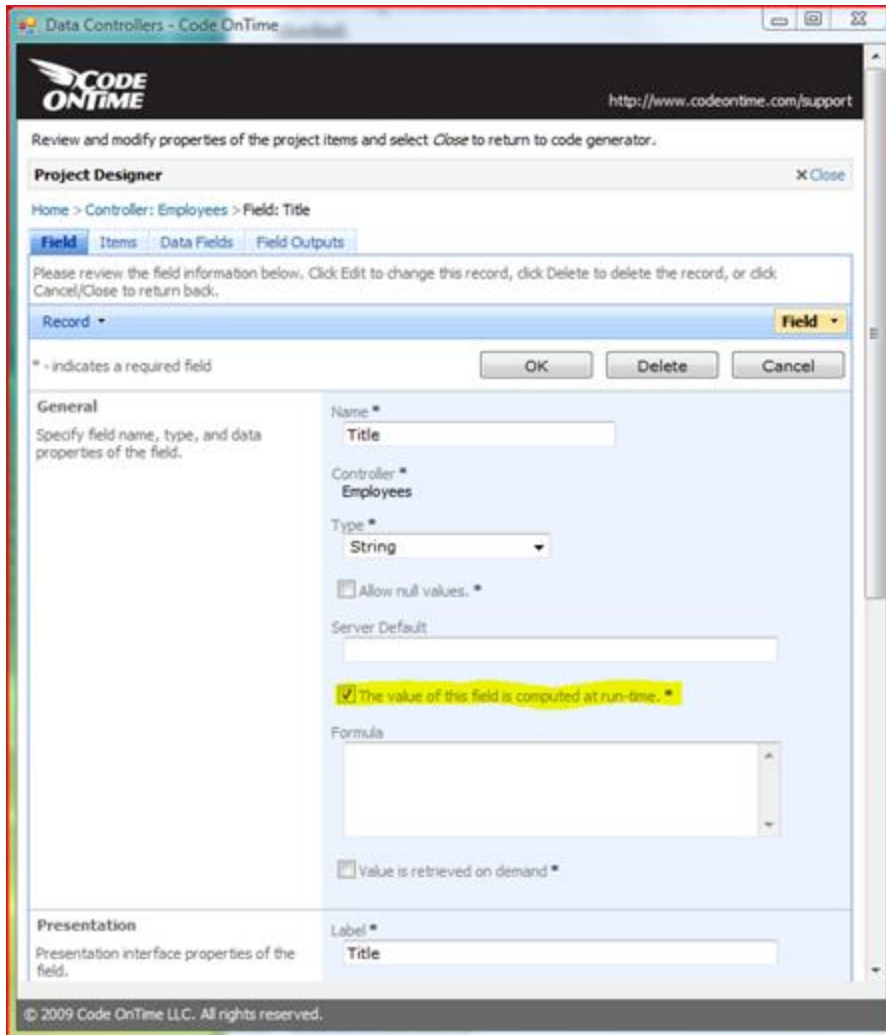
```
<category headerText="General" >
  <description><![CDATA[ Specify field name, type, and data properties of
the field.]]></description>
  <dataFields>
    . . . . .
    <dataField fieldName="Computed">
      <headerText>The value of this field is computed at run-
time.</headerText>
    </dataField>
    <dataField fieldName="Formula" columns="50" rows="5">
      <visibility>
        <expression type="ClientScript" test="[Computed]==true"/>
      </visibility>
    </dataField>
    <dataField fieldName="OnDemand"/>
  </dataFields>
</category>
```

The visibility expression test is written in *JavaScript*. The expression test can refer to the names of the fields defined in the view. The field referenced must be incased in square brackets.

The expression test is evaluated on the client whenever the field values are changed. Evaluation does not cause the server post-back.

The following screenshots show the form before and after the computed field checked.





CONDITIONAL VISIBILITY OF CATEGORIES

The following snippet from the the same data controller shows visibility expression for a field category.

```
<category headerText="Lookup">
  <dataFields>
    <dataField fieldName="ItemsStyle"/>
    <dataField fieldName="ItemsDataController"/>
    <dataField fieldName="ItemsDataValueField"/>
    <dataField fieldName="ItemsDataTextField"/>
    <dataField fieldName="ItemsNewDataView"/>
    <dataField fieldName="ContextFields"/>
```

```

</dataFields>
<visibility>
  <expression type="ClientScript" test="[Type] != 'Byte[]' && [Type] != 'Object' && [OnDemand] != true" />
</visibility>
</category>

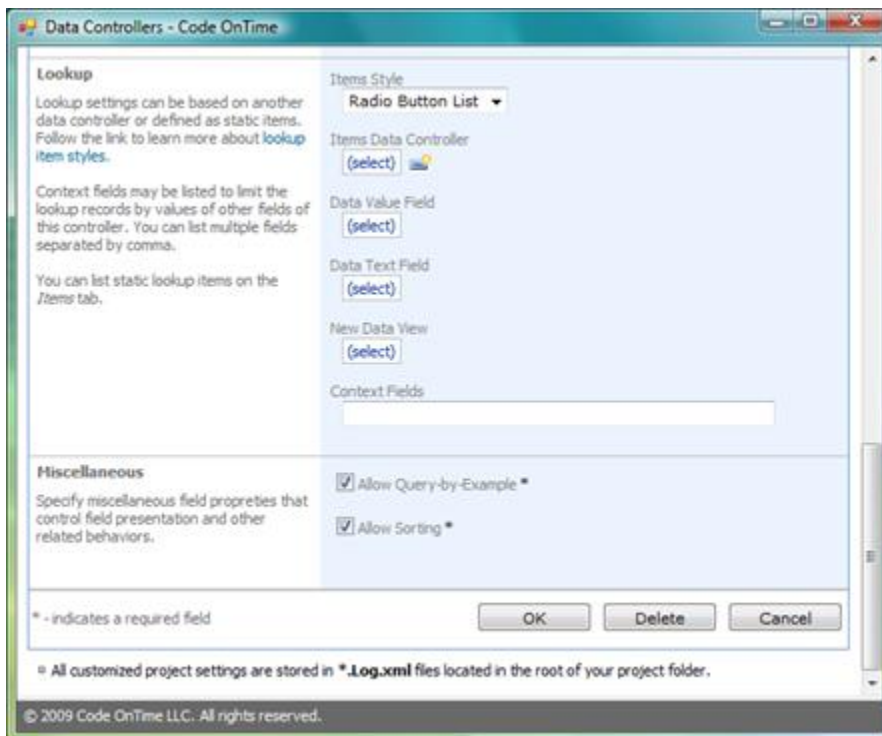
```

The *JavaScript* expression reads as follows:

```
[Type] != 'Byte[]' && [Type] != 'Object' && [OnDemand] != true
```

If the value of field *Type* is not equal to *Byte[]* or *Object* and the field is not *OnDemand* then the category of fields is visible.

The next screenshot shows the visible *Lookup* field category.



CONCLUSION

The visibility expressions can be defined directly in the data controller files. The upcoming update to [Code OnTime Designer](#) will allow easy modifications of the visibility expressions.

Code OnTime LLC

<http://www.codeontime.com>