Sage 300 Web Screens SDK

View Field Attribute Wizard

July 2022

The MIT License (MIT)

Copyright © 2022 The Sage Group plc or its licensors. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Contents

[1. Overview 4](#_Toc108439263)

[2. Load Wizard Solution 5](#_Toc108439264)

[2.1 Restore Nuget Packages 6](#_Toc108439265)

[2.2 Compile for References and Packages 6](#_Toc108439266)

[3. Module Model References 7](#_Toc108439267)

[3.1 Add References to Project 8](#_Toc108439268)

[3.2 Add Reference (Using) to Code File 9](#_Toc108439269)

[4. Compile the Solution 11](#_Toc108439270)

[5. Run the Wizard 12](#_Toc108439271)

[5.1 Business View Credentials and Source Code Folder 13](#_Toc108439272)

[6. Modify Model Files 14](#_Toc108439273)

[7. Modified Files 15](#_Toc108439274)

[8. Load Your Solution 16](#_Toc108439275)

1. Overview

This document is intended to serve as a guide for explaining, modifying, compiling, and running the Sage 300 View Field Attribute Wizard.

Since this wizard must be modified prior to running, a binary or compiled version of this wizard has not been delivered in the bin folder of the Web SDK.

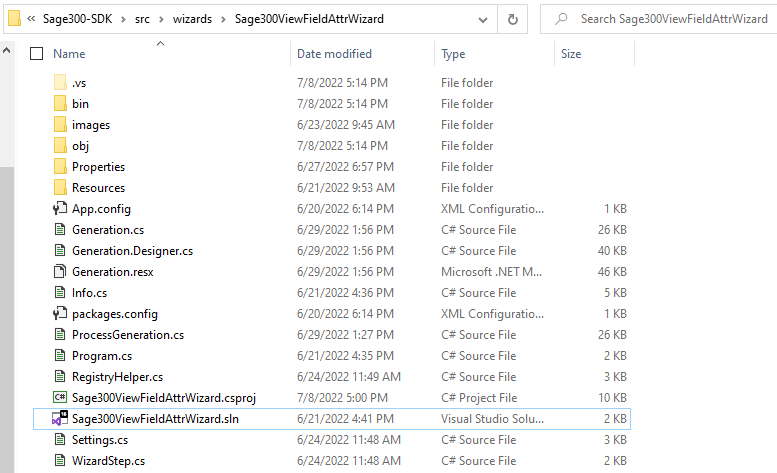
The purpose of this wizard is to modify the MVC models in the Sage/Partner solution’s model project(s) to add the View Field attribute to public properties that belong to the underlying business (Accpac) view.

The View Field attribute was introduced in Sage 300 2022 with our PJC web screens for efficient value setting of entities properties from JavaScript via the controller. Some partners have already implemented their own reflection methods for creating generic methods to do the same. Adding this attribute along with the already established helpers and methods in the Web Screen framework will ensure that partners do not have to re-invent the wheel.

The running of this wizard is optional if the above-described behavior is not required. For Sage 300 2023, all public properties in all module models have been modified to include this new attribute. This will allow support of not only the described behavior but also support for a future enhancement to Razor View Helpers which will reduce the amount of code required in the Sage 300 Razor Views.

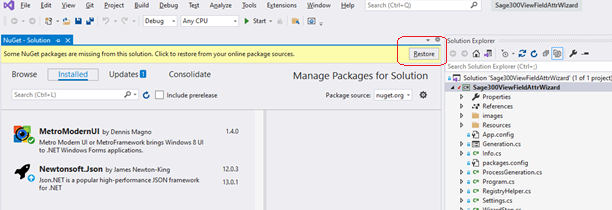
1. Load Wizard Solution

Since the Sage 300 View Field Attribute Wizard requires modification, the wizard must have binary references added for Sage/Partner module(s) model binaries, the first step in the process is to load the **Sage300ViewFieldAttrWizard.sln** file located in the Web SDK’s **src\wizards\Sage300ViewFieldAttrWizard** folder.



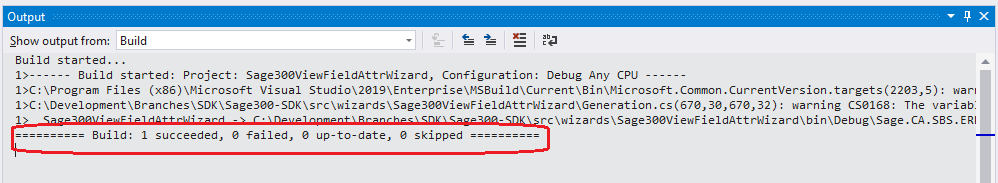
* 1. Restore Nuget Packages

From the **Tools** 🡪 **Nuget Package Manager** 🡪 **Manage Nuget Packages for Solution…** menu item, select the **Restore** button to add Nuget packages to solution.



* 1. Compile for References and Packages

With the references and packages resolved, it is a good idea to compile the solution to ensure all references are resolved by selecting the **Build** 🡪 **Build Solution** menu item.

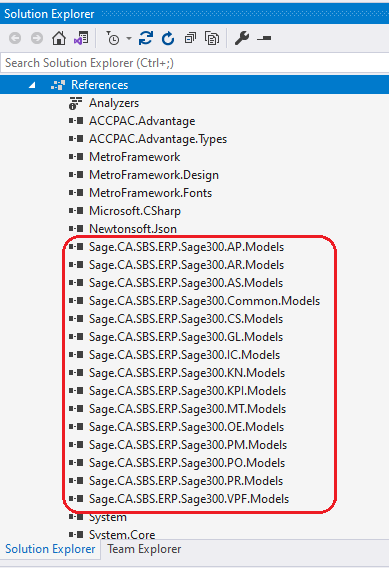


There should not be any missing references, but if there are, they must be resolved. This wizard uses the **SageWebDir** environment variable to determine the location of the locally installed Sage 300 Online Web folder.

1. Module Model References

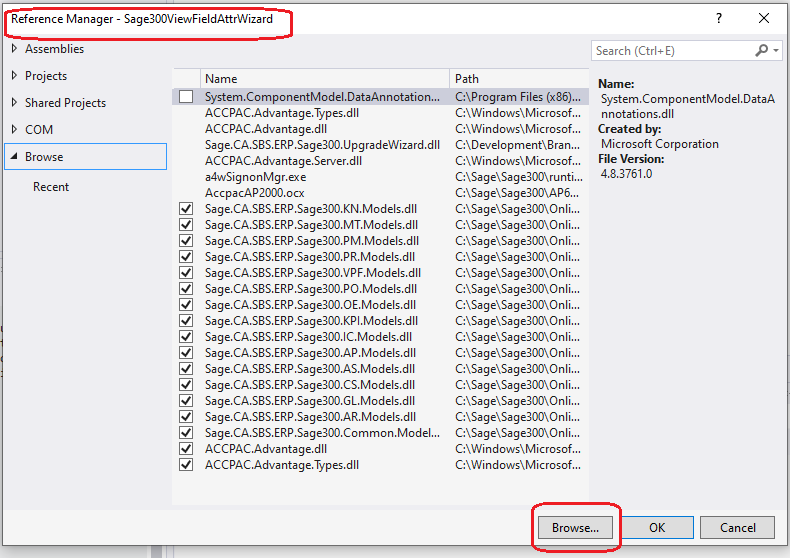
This wizard will use a combination of reflection and source code to modify the models in your module’s model project. Therefore, you will be required to add your module’s model binary reference to the Sage300ViewFieldAttrWizard project.

The screenshot below shows how this wizard already has the Sage 300 module’s model references added.



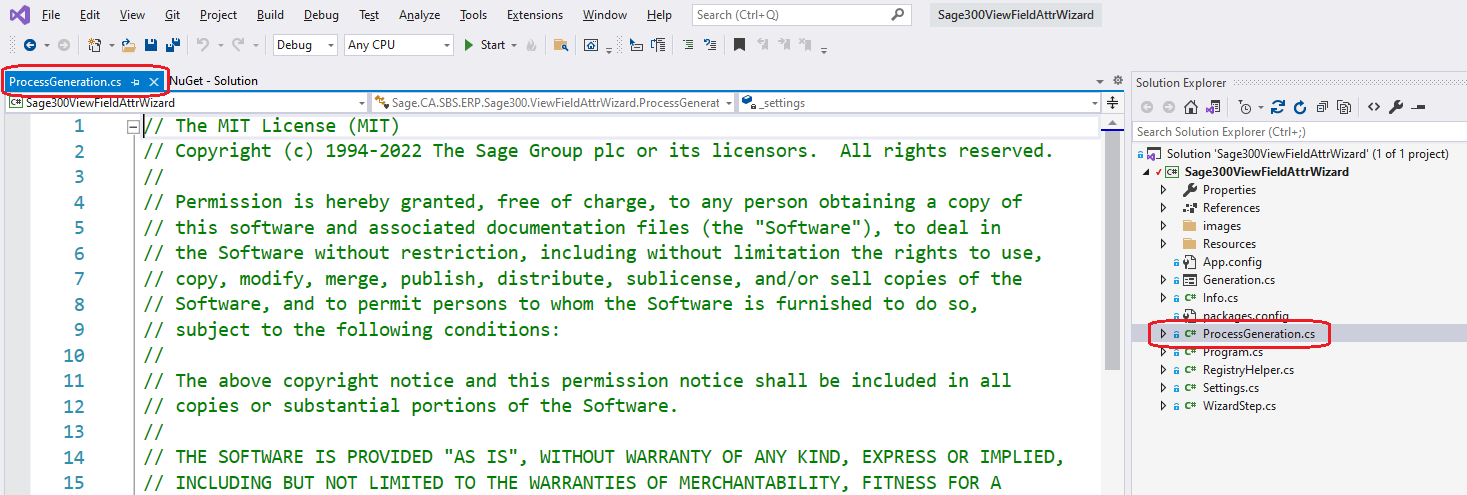
* 1. Add References to Project

In this step, you will need to locate your module’s model binary for it to be added to the project. Right-click on the **References** item in the **Solution Explorer** and select **Add Reference…** to display the **Reference Manager** dialog. Select the **Browse…** button to navigate to the folder containing your binary. Once located, add the binary reference to the project.

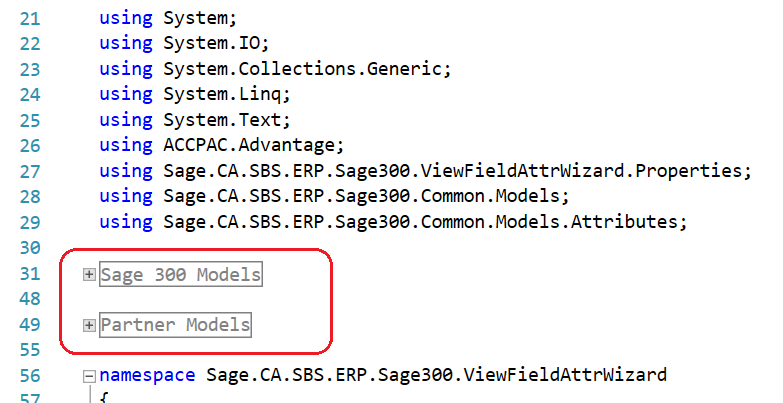


* 1. Add Reference (Using) to Code File

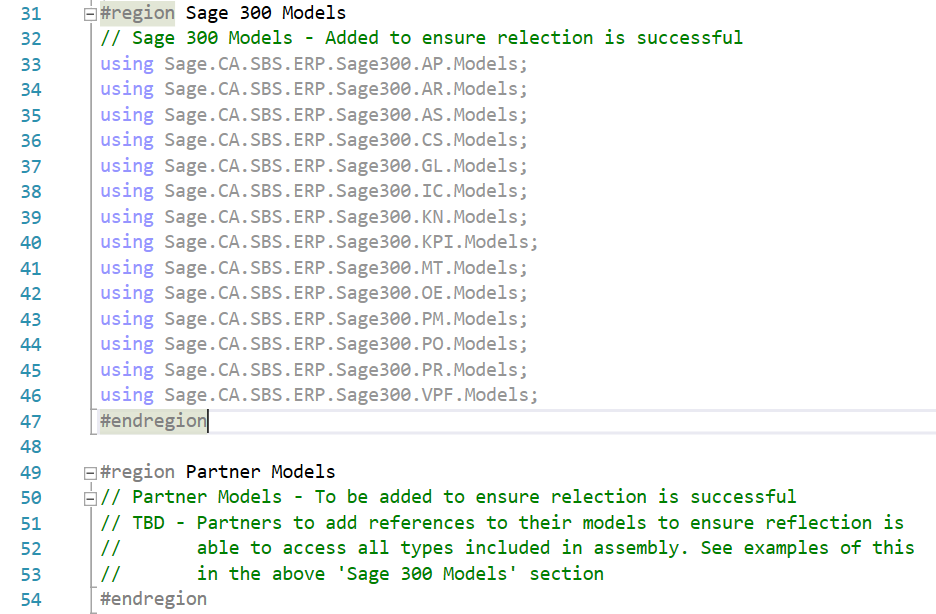
Now that the binary reference (or multiple if there are multiple modules) has been added to the project, open the **ProcessGeneration.cs** file by double-clicking on it in the **Solution Explorer.**



Scroll down the code file to the Sage 300 Models and Partner Models regions.



Expand these regions by clicking on the **+** signs.



In the Partner Models region, you will add a using statement for the binary added in section 3.1.

Reference the Sage 300 Models section for what was added for the Sage 300 modules

1. Compile the Solution

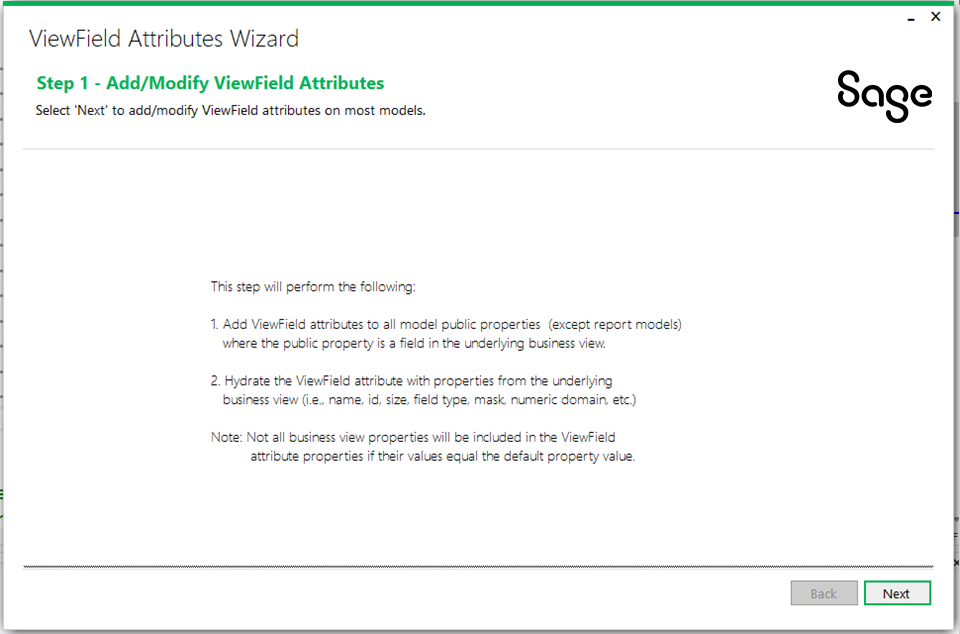
Now that the wizard source has been modified to add a reference (or references if there are multiple modules) it is time to compile the solution. Select **Build** 🡪 **Build Solution** and ensure it compiles successfully.

Graphical user interface, text, application, Word

Description automatically generated

1. Run the Wizard

The wizard can either be run in debug mode with Visual Studio or select **Sage.CA.SBS.ERP.Sage300.ViewFieldAttrWizard.exe** from the projects **bin\Debug** folder.



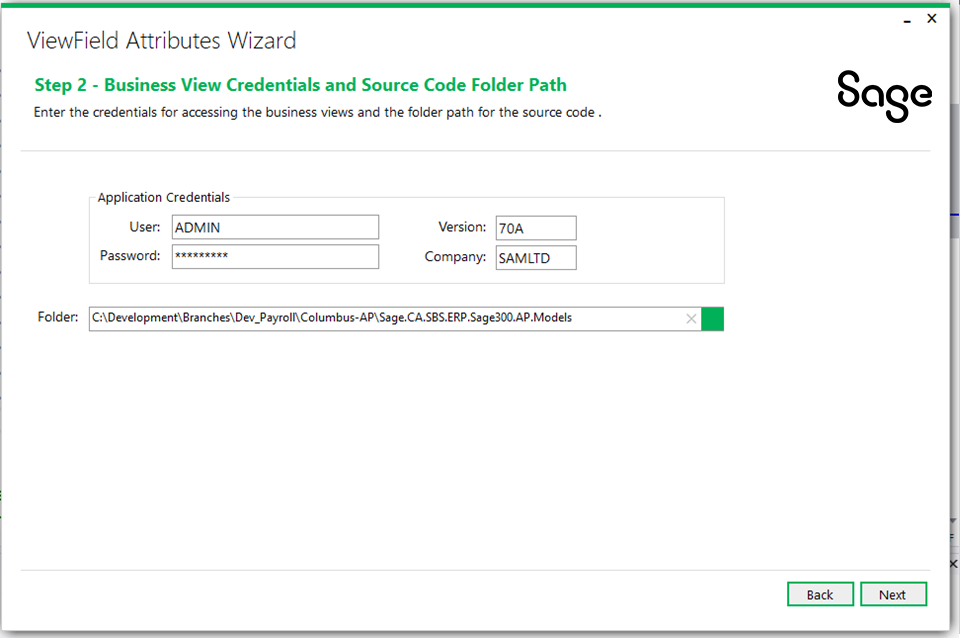
Prior to continuing with the wizard, please ensure that your source code is backed up since this wizard will be modifying source code.

Select **Next** to advance to the next step.

* 1. Business View Credentials and Source Code Folder

Since the View Field Attribute properties will be hydrated from the model’s underlying business view, you are required to enter credentials to connect to a Business Session (Accpac Session).

In this step, you will also indicate where the module’s model source code project is located (**.csproj**). The reason is that the wizard parses the .csproj file to locate all models in the project (i.e., Sage.CA.SBS.ERP.Sage300.AR.Models.csproj).



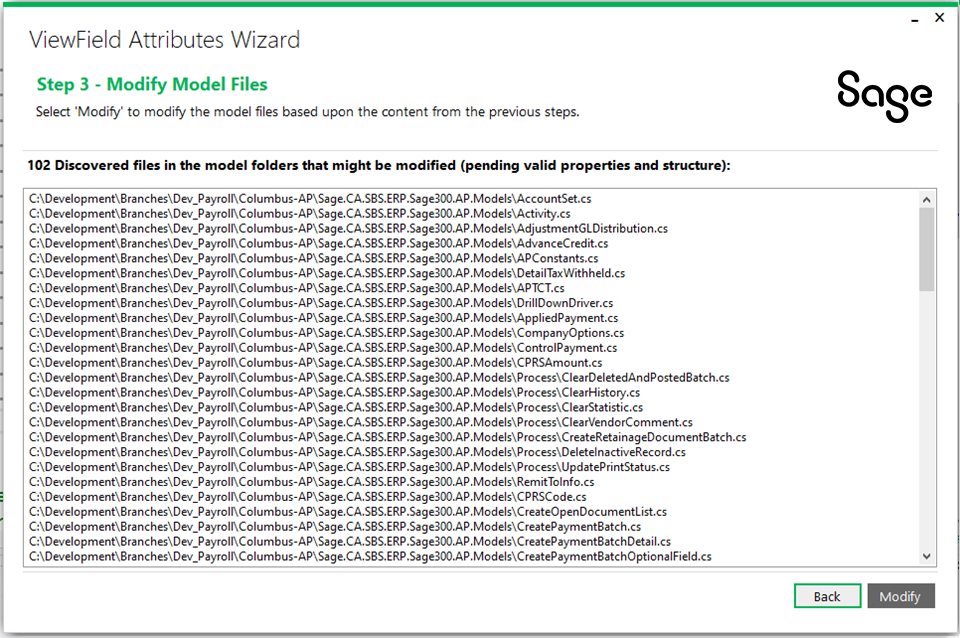
If the credentials are invalid or the folder has not been entered or does not exist, you will be prevented from continuing.

Select **Next** to advance to the next step.

1. Modify Model Files

In this step, once the .csproj file has been located as specified in the previous step, the model code files discovered in the .csproj file will be displayed.

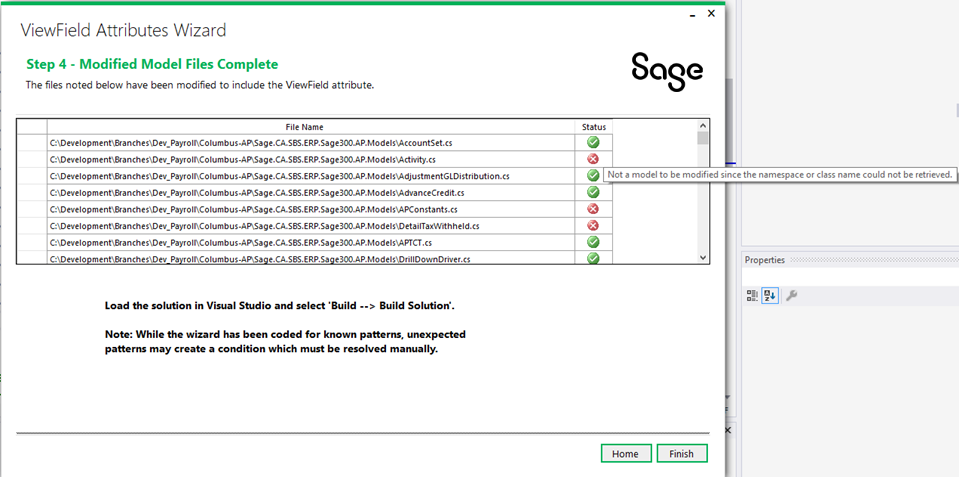
Even though a model class file is displayed, this does not guarantee that it will be modified. The **Modify** step will locate the specified source code file, use reflection to reference its assembly, locate the underlying business view as specified in the model’s Fields class **EntityName** or **ViewName** public constant.



Select **Modify** to begin the modification process.

1. Modified Files

Once the wizard has finished modifying the specified model’s source code, a list of the processed files (modified or not) will be displayed. Any errors or other information can be obtained by hovering the mouse about the red X for the file.



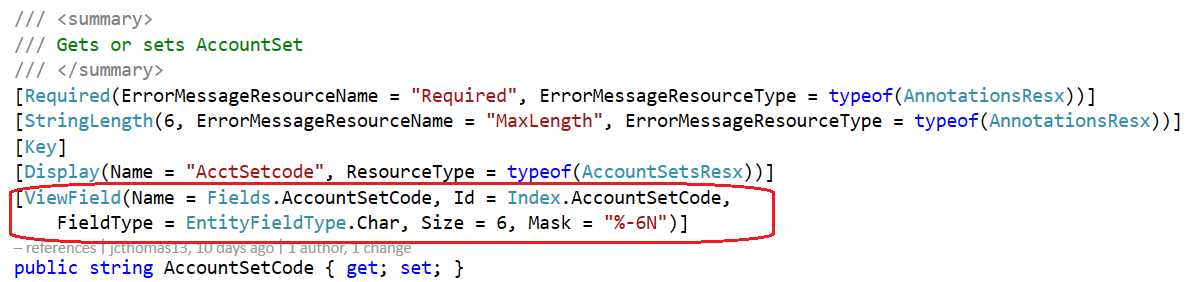
As stated in the previous step, other class files other than models may be present in the model’s folders and therefore are not “valid” models and will not be modified. Only class files that meet the definition of a model class will be processed (valid underlying business view, Fields class implementation, Index class implementation, etc.)

Select **Home** to run the wizard again or **Finish** to exit.

1. Load Your Solution

Now that your model source files have been modified, it is time to load your solution and ensure that it compiles as expected.

To look at the type of changes that have been made, load any model file, and scroll to a public property. You will see the new ViewField attribute that has been added to the public property.



If a public property does not have the ViewField attribute it could be because the model file encountered an error, the public property is not an actual property in the underlying business view (it is merely a worker property or …).

Note in the screenshot above how the ViewField attribute properties have been hydrated from the underlying business view!

The ViewField attribute has numerous properties (Name, Id, FieldType, Size, Precision, Mask, and Number Domain). Currently the ViewFieldHelper function has methods to leverage all these properties. Some properties may not be added to the ViewField attribute if the underlying business view’s property is a default value or has no value (i.e., no size or no mask or no precision or …)