

# Modeling Data - How-to

## Table of Contents

<b>Outline</b>	<b>2</b>
<b>How to</b>	<b>2</b>
Create Entities	2
Bootstrap Data from Excel	7
Create Static Entities	9

# Outline

In this exercise lab, we will create the data model for the OSMDb application created in the previous exercise.

The data model of this application will exclusively be created in the OSMDb\_Core module, and will consist at this stage of two Entities, **Movie** and **Person**, and two Static Entities, **MovieGenre** and **PersonRole**. These Entities will represent the movies (Movie) in the database and their genres (MovieGenre), as well as the cast and crew (People) and the role they play in the movies (PersonRole).

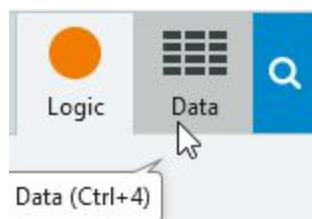
The movie genres we will use in this application should be Comedy, Action, Drama and Horror, while the cast and crew roles used will be Director, Producer, Actor and Crew.

## How to

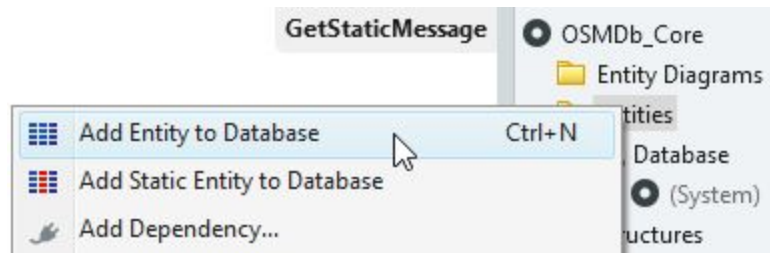
In this document, we will describe, step by step, the exercise 2.4. - *Modeling Data Exercise*.

### Create Entities

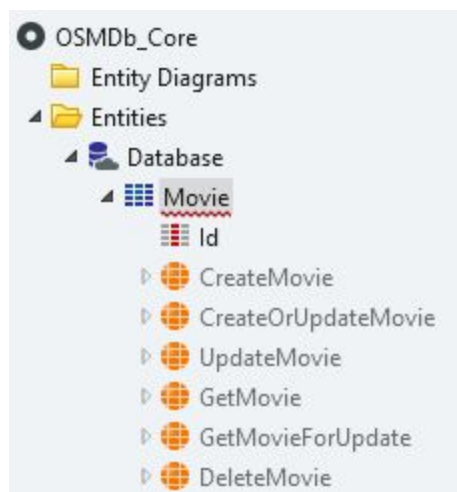
1. Create the *Movie* Entity with the following attributes: *Title*, *Year*, *PlotSummary*, *GrossTakingsAmount* and *IsAvailableOnDvd*, with the first two being **mandatory**. The Entity will also be **Public** and **Exposed as Read Only**.
  - a. Switch to the OSMDb\_Core module.
  - b. Click the **Data** tab in the upper right corner of the workspace, to switch to the Data elements.



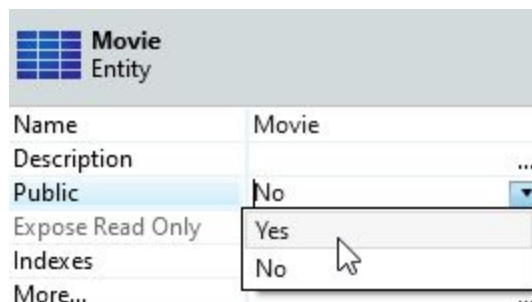
- c. In the elements area, right-click the **Entities** folder and select **Add Entity to Database**.



- d. Type *Movie* for the **Name** of the Entity.
- e. Expand the **Movie** Entity. There is an auto-number *Id* attribute and six Entity Actions to provide typical Create, Read, Update and Delete (CRUD) functionality.



- f. Since we will be using this Entity on our UI module, in the properties editor at the bottom right, change the **Public** property to **Yes**.



- g. Still in the properties of the Entity, leave the **Expose Readonly** to **Yes**, to make the Entity exposed with read only permissions.

- h. Right-click the Movie Entity and select **Add Entity Attribute**.



- i. Enter *Title* for the **Name** of the attribute. Notice that the errors disappear.
- j. Change its **Is Mandatory** property to *Yes*.

Title Entity Attribute	
Name	Title
Description	...
Label	Title
Data Type	Text
Length	50
Is Mandatory	No
Default Value	Yes
	No

- k. Right-click the Movie Entity and select **Add Entity Attribute** again.
- l. Type *Year* for the **Name** of the attribute. Notice the default **Data Type** is *Integer*. Make this attribute **Mandatory**.
- m. Add the *PlotSummary* attribute. Set its **Length** to 500 characters.

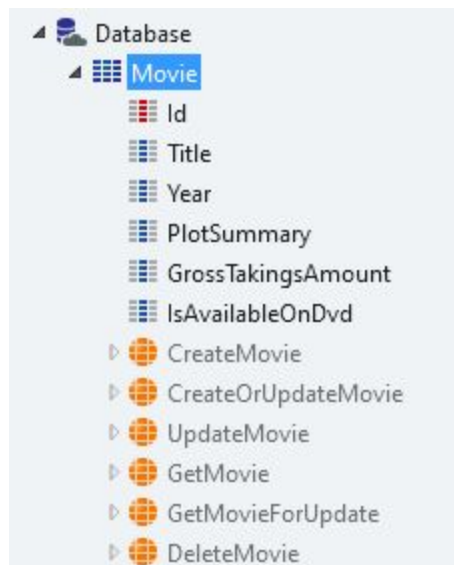
PlotSummary Entity Attribute	
Name	PlotSummary
Description	...
Label	Plot Summary
Data Type	Text
Length	500
Is Mandatory	No
Default Value	

- n. Add the *GrossTakingsAmount* and *IsAvailableOnDVD* attributes. Notice the default **Data Types** are correctly set to *Currency* and *Boolean*.

- o. Select the GrossTakingsAmount attribute and change its **Label** property to *Gross Takings*. This will shorten the default label used in OutSystems, when generating UI, as we will see in a later exercise.

GrossTakingsAmount Entity Attribute	
Name	GrossTakingsAmount
Description	...
Label	Gross Takings
Data Type	Currency ▼
Length	
Decimals	
Is Mandatory	No ▼
Default Value	...

- p. The Movie Entity should look like this

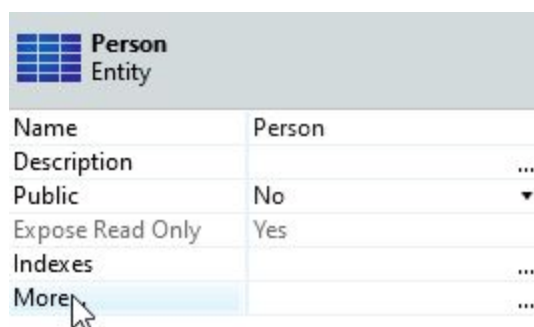


2. Create the *Person* Entity with the following attributes: *Name*, *Surname*, *DateOfBirth* and *DateOfDeath*, with the first three being **mandatory**. The Entity will also be **Public** and **Exposed as Read Only**.
  - a. On the **Data** tab, create a new Entity and set its name to *Person*.
  - b. In the properties editor at the bottom right, change the **Public** property to *Yes* and the leave **Expose Read Only** to *Yes*.
  - c. Add the following attributes to the Entity: *Name*, *Surname*, *DateOfBirth*, and *DateOfDeath*. Did Service Studio guess the Data Types correctly from the attribute name? **Always double-check!**

- d. Set *Name*, *Surname* and *DateOfBirth* as **mandatory**.
- e. The Person Entity should look like this



- f. Select the Person Entity and, in the properties editor, double-click the **More...** property to launch the **Entity Editor**.



- g. In the Entity Editor dialog, expand the **More Options** section and change the **Label (plural)** property to *People* and click the **Close** button.

Person

Name: Person

Description:

**More options**

Label: Person

Label (plural): People

Identifier Attribute: Id

Order By Attribute:

Label Attribute:

Is Active Attribute:

Attributes | Indexes | Example Record | Advanced

New | Delete | Up | Down

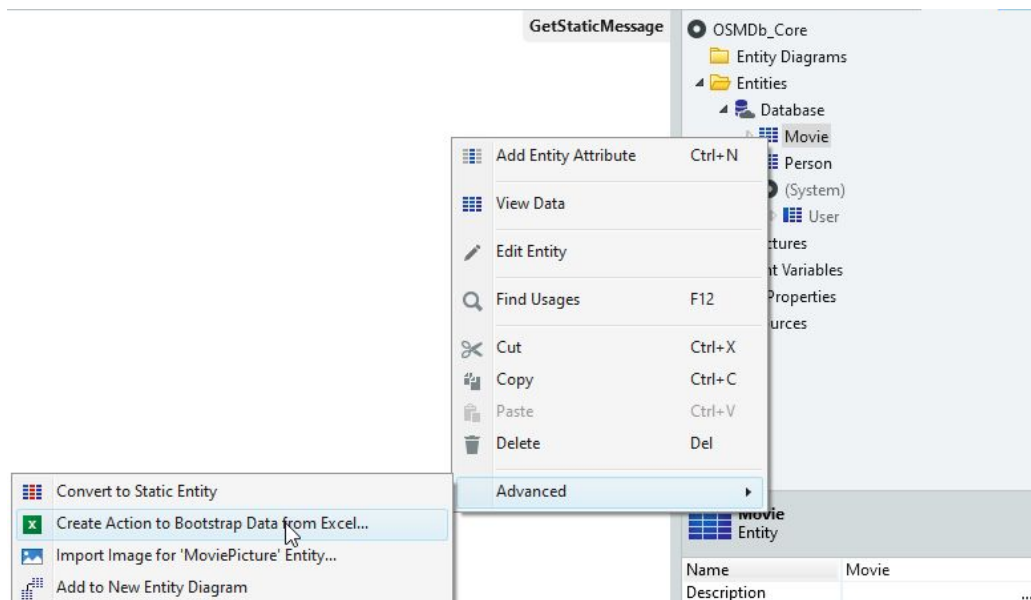
- Id
- Name
- Surname
- DateOfBirth
- DateOfDeath

?

CLOSE

## Bootstrap Data from Excel

1. Populate the Movies Entity with data, by bootstrapping data from the Excel file **Movies.xlsx**.
  - a. Right-click the Movie Entity, select Advanced, and then the option **Create Action to Bootstrap Data from Excel...**



- b. Browse to the Resources folder and select the **Movies.xlsx** Excel file.
- c. In the new dialog, ensure that the column headers names from the Movies.xlsx excel file (**Excel Columns**) match the names of the Movie Entity attributes (**'Movie' Attributes**). If so, click on the Proceed button. Otherwise, click on the Cancel button and fix the names, or data types, of the mismatched Movie Entity attributes. Then, redo the third step again. Your Entity attributes **must match in name and type** what is in the Excel file.

Create Action to Bootstrap Data from Excel

The 'BootstrapMovies' Action will be created to bootstrap data from Excel Sheet 'Movie'.

Excel Columns	'Movie' Attributes
Title	Title
Year	Year
PlotSummary	PlotSummary
GrossTakingsAmount	GrossTakingsAmount
IsAvailableOnDVD	IsAvailableOnDvd

PROCEED

CANCEL

2. Populate the Person Entity with data, by bootstrapping data from the Excel file **People.xlsx**. Don't forget to verify that the values in **Excel Columns** match the values in **'Person' Attributes** before you click Proceed.

Create Action to Bootstrap Data from Excel

The 'BootstrapPeople' Action will be created to bootstrap data from Excel Sheet 'Person'.

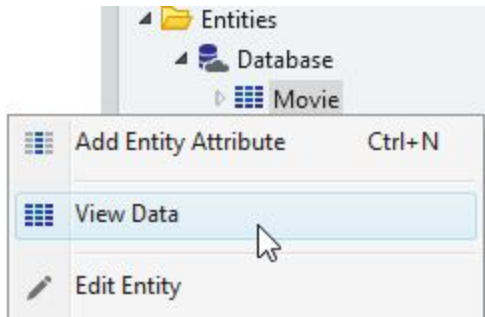
Excel Columns	'Person' Attributes
Name	Name
Surname	Surname
DateOfBirth	DateOfBirth
DateOfDeath	DateOfDeath

PROCEED

CANCEL

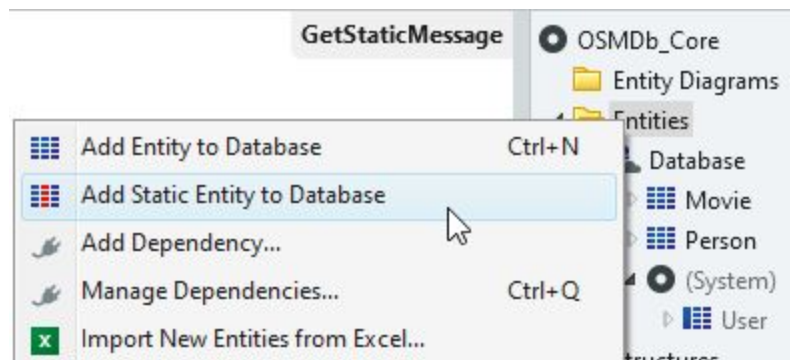


3. Publish the module to save the changes in the server.
4. After a few seconds, right-click on the **Movie** or **Person** Entity, select *View Data*, and make sure that the data is previewed properly.



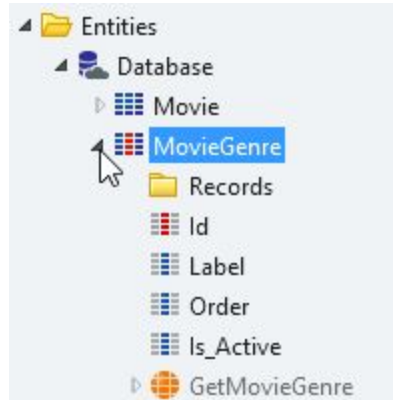
## Create Static Entities

1. Create a Static Entity called *MovieGenre*, with the following records: *Comedy*, *Action*, *Drama* and *Horror*. This Entity should have an additional attribute, called *MinimumAge*, with the following values per each record: 6, 12, 16 and 18. The Static Entity should be set as **Public**.
  - a. Right-click the Entities folder and select **Add Static Entity to Database**.

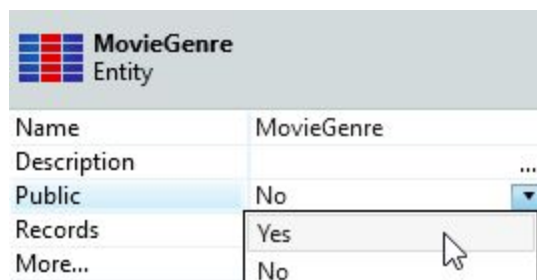


- b. Type *MovieGenre* for the **Name** of the Static Entity.

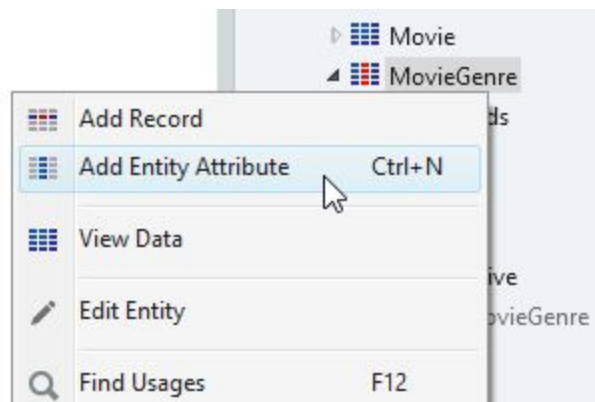
- c. Expand the **MovieGenre** Entity by clicking on the expand symbol.



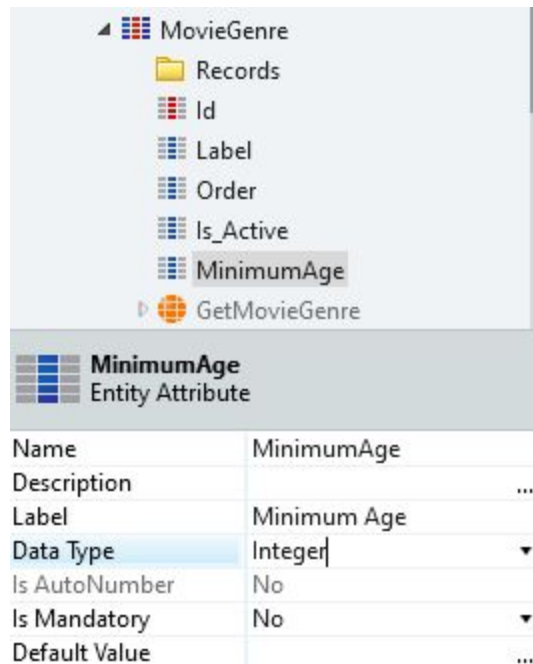
- d. Change the **Public** property to Yes.



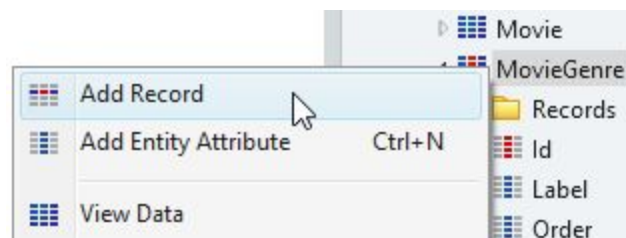
- e. Right-click the MovieGenre Entity and select **Add Entity Attribute**.



- f. Set its **Name** to *MinimumAge* and change the **Data Type** to *Integer*.



- g. Right-click the Static Entity and select **Add Record**.



- h. Enter *Comedy* for the identifier of the Record. In the properties editor, set the value 6 in the **MinimumAge** attribute this Record.

The screenshot shows the 'MovieGenre' entity in the OutSystems Studio. Under the 'Records' folder, the 'Comedy' record is selected. The 'Identifier' is set to 'Comedy'. The 'Icon' is set to 'Default Icon'. The 'Attribute Values' table is as follows:

Attribute Values	
Id	(Auto Number)
Label	"Comedy" ...
Order	1 ...
Is_Active	True ▼
MinimumAge	6 ...

- i. Add 3 more Records: *Action*, *Drama* and *Horror*. Set these Records' **MinimumAge** to 12, 16 and 18, respectively.
- j. The Static Entity should look like this

The screenshot shows the 'MovieGenre' entity in the OutSystems Studio. Under the 'Records' folder, four records are listed: 'Comedy', 'Action', 'Drama', and 'Horror'. The 'Id', 'Label', 'Order', 'Is\_Active', and 'MinimumAge' attributes are visible for each record. The 'GetMovieGenre' service is also shown at the bottom.

2. Create a Static Entity called *PersonRole*, with the following records: *Director*, *Producer*, *Actor* and *Crew*. The Static Entity should be set as **Public**.
  - a. Create a new Static Entity and set its **Name** to *PersonRole*.
  - b. In the properties at the bottom right, change the **Public** property to *Yes*.
  - c. Add four Records to the Static Entity: *Director*, *Producer*, *Actor* and *Crew*.
  - d. The Static Entity should look like this



- e. Click the  **1-Click Publish** button to publish the module.