

Practice Exercises

GeneXus Advanced Course

GeneXus[™] 17

April 2021

© GeneXus. All rights reserved. This document may not be reproduced by any means without the express permission of GeneXus[™]. The information contained herein is for personal use only.

Registered Trademark:

GeneXus is a trademark or registered trademark of GeneXus S.A. All other trademarks mentioned herein are the property of their respective owners.

CONTENTS

PROBLEM.....	5
NEW PROJECT, NEW KNOWLEDGE BASE	5
TRANSACTIONS	5
“Movie” Transaction.....	7
“Room” Transaction	7
“Schedule” Transaction	8
“Ticket” Transaction	9
RULES	10
Rule Review.....	10
Client Side Validation Behavior [suggested].....	11
Adding New Rules.....	11
TRIGGERING EVENTS.....	12
Navigation Tree.....	12
Triggering Moments.....	13
CONTROLS IN COMPOUND FK.....	15
ADDING NEW ATTRIBUTES.....	15
PARALLEL TRANSACTIONS [SUGGESTED].....	16
SUBTYPES.....	16
Adding a country to the movie	17
Recursive Subtypes.....	17

FORMULAS	18
DYNAMIC TRANSACTIONS	20
EVENTS IN TRANSACTIONS	20
PROCEDURES AND LISTS	21
Subroutines	22
Unique Clause	23
Data Selector	25
PAGING CLAUSES	27
Paging clauses in procedures	27
BUSINESS COMPONENT	27
Updating Employees in Charge	28
Working with Movies and their Actors	28
UPDATING THE DATABASE WITH A PROCEDURE	29
WEB PANELS	30
Working with the Pattern	30
Creating interactive screens	30
Multiple grids	31
Grid without base table	33
Loading Collections	33
Saving context information	34
Special screen for ticket sales	34
MULTI-CHANNEL SCREENS: PANELS	36
Testing the application on smart devices	38

PROGRESSIVE WEB APPLICATIONS [SUGGESTED].....	39
WEB SERVICES [SUGGESTED]	44

PROBLEM

A company that manages a Cinema hires you to develop a system to store, handle, and display the information it works with. Suppose that the system is made up of these modules:

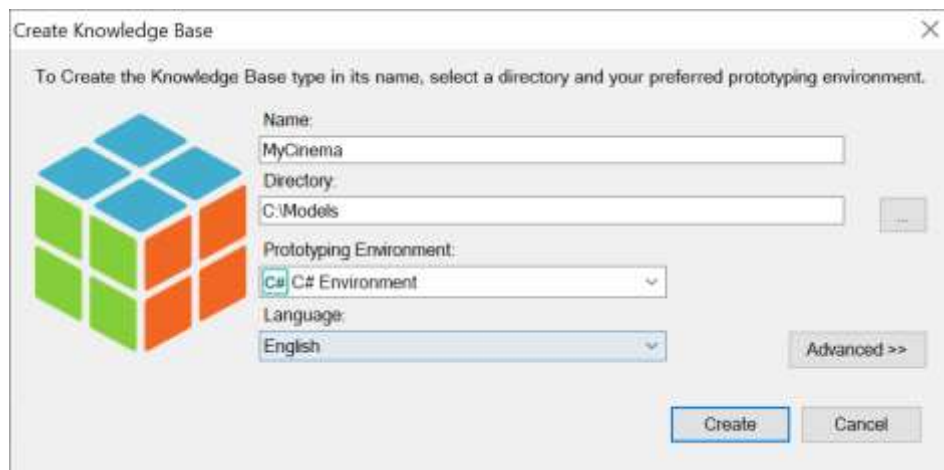
- **Back end:** part of the application that will run on a web server, so that the company's employees can handle the information from any location with an Internet connection.
- **Front end:** part of the application that will run on a web server, in such a way that customers can see the information in an aesthetically pleasing way.

NEW PROJECT, NEW KNOWLEDGE BASE

Open GeneXus and create a knowledge base called *MyCinema* to start developing the application.

Tips:

- Select C# as a development environment. Make sure everything you need is installed (including SQL Server). If you're using GeneXus Trial, the generation environment with C# and SQL Server is already predefined, prototyping in the Amazon cloud.
- Do not create the knowledge base in "My Documents" folder or any other folder below "Documents and Settings" because these folders have special permissions granted by Windows.



NOTE: This practice document contains two types of exercises:

1. Exercises to be done in the GeneXus IDE
2. Exercises that can be excluded from the application; however, there is a space in the practice document for the student to enter the answers. (These exercises can also be replicated in the IDE.)

TRANSACTIONS

Import the file MyCinemaTransactions.xpz located in the Practice folder, from the menu options Knowledge Manager → Import.

Look at the imported transactions below the *Trn Folder* and navigate through their structure:

- Actor
- Category
- Country
- Customer
- FilmDirector
- Movie

In the Category, Country and Film Director transactions, check that the *Data Provider* property is set to *True* and that there is a Data Provider below the transaction that will load the recorded data automatically.

The same has to be done for the following transactions: Actor and Customer.

Note: More data can be added at runtime or in the indicated data provider if desired.

For the Actor transaction, how would you add the Actor's photo automatically? You can use the images attached in the practice folder.

Customer:

- 1, Anne Parker, anneparker@gmail.com, false
- 2, Lee Wu, leewu@hotmail.com, true
- 3, Ben Smith, smithb@gmail.com, false
- 4, Marie Bell, bellmarie@gmail.com, true

Actor

- 1, Tom Hanks, M, United States
- 2, Leonardo DiCaprio, M, United States
- 3, Denny Brechner, M, Uruguay
- 4, Daisy Ridley, F, United Kingdom
- 5, Emma Watson, F, United Kingdom
- 6, Robert Downey Jr, M, United States
- 7, Chris Hemsworth, M, Australia

Remember that:

- We can use the Find formula to navigate a table and find information according to a condition.
- In the KB Explorer, from the menu options Customization → Images, we can upload images that will be available in the context of the KB, so we could save them in the database with the link() function.
- By typing the name of an enumerated domain and the period key (.) we can access the list of enumerated values.

We are requested that it shouldn't be possible to save two Clients or two Actors with the same name. **What control should we add to implement this?** Make this change in GeneXus.

“MOVIE” TRANSACTION

Open the Movie transaction and note that we have the following data for it: Identifier, Name, Year, Poster and a brief description.

In addition, the following information should be recorded: Movie's Country, Category, and Director.

A Movie can have 1 or more actors associated with it; the following information about the Actor should be displayed: Identifier and Name.

It is also necessary to know whether the actor being registered for the movie is the lead actor. To this end, add a new attribute called MovieActorStarring of Boolean type.

To make the application user-friendly, apply the WorkWithForWeb pattern to all transactions created so far.

Run the application and add several movies to test the transaction's functionality.

“ROOM” TRANSACTION

The cinema has several viewing rooms, which in turn are of different types (Normal, VIP, 3D, etc.). Each viewing room has a fixed number of seats that we must also save.

It is recommended that Type be an enumerated value.

How to create an enumerated domain? Make this change in the IDE.

Name	Description	Value
Normal	Normal	Normal
VIP	VIP	VIP
in3D	3D	3D

Run the application and add 3 regular rooms, 1 VIP and 1 3D.

About enumerated domains:

Names starting with numbers are not allowed in the "Name" column. In case of 3D rooms, the name of the room must **begin with a letter**. The Description and Value columns can have the 3D value.

"SCHEDULE" TRANSACTION

The schedule is made up of the viewing Room, the date and time in which the movies will be shown (only at these times: 11:00 AM, 2:00 PM, 5:00 PM and 8:00 PM).

Tip:

Use a MovieTime domain of Character type that allows selecting the time:

Name	Description	Value
EarlyMorning	11 AM	11AM
Afternoon	2 PM	2PM
Evening	5 PM	5PM
Night	8 PM	8PM

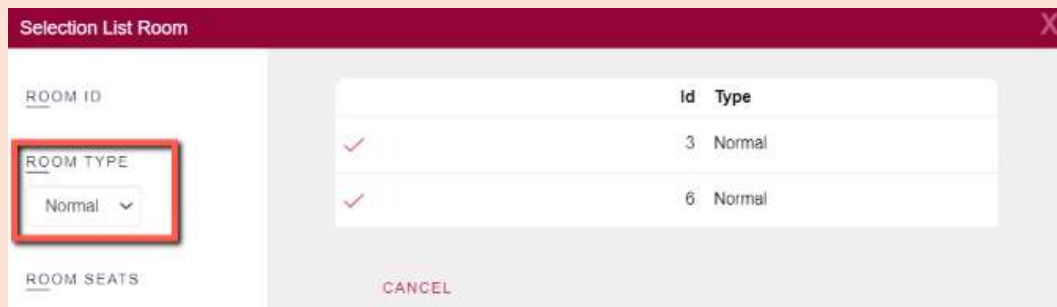
To make sure that no viewing Room, date and time are repeated, the viewing Room identifier, date and time should be set as compound keys. The movie can be repeated because the same movie can be shown in different rooms, dates, and times, so we don't include it in the control.

Name	Type	Description	Formula	Nullable
Schedule	Schedule	Schedule		
RoomId	Id	Room Id		No
ScheduleMovieDate	Date	Schedule Movie Date		No
ScheduleMovieTime	MovieTime	Schedule Movie Time		No
MovieId	Id	Movie Id		No
MovieName	Name	Movie Name		
MoviePoster	Image	Movie Poster		
RoomSeats	Numeric(4,0)	Room Seats		

Run the application and make some insertions to the schedule.

Tip about enumerated domains:

Note that when trying to enter a viewing Room, not all the records appear in their entirety in the selection web panel because there is a domain enumerated by Room type and the user must filter by that domain to choose the corresponding room number.



To avoid the above go to the Room transaction and in the RoomType attribute change the EmptyItem property to True.

Do the same for the ActorGenre and ScheduleMovieTime attributes.

“TICKET” TRANSACTION

For the ticket, we will place its identifier, the date of purchase, whether the purchase is assigned to a customer or not, the corresponding movie with its schedule, the date of the movie and its viewing room, as well as the number of seats sold, the cost of the ticket and the total price (calculate it according to the number and cost of the tickets).

Name	Type	Description	Formula	Nullable
Ticket	Ticket	Ticket		
TicketId	Id	Ticket Id		No
TicketDate	Date	Ticket Date		No
CustomerId	Numeric(4.0)	Customer Id		Yes
CustomerName	Name	Customer Name		
ScheduleMovieTime	MovieTime	Schedule Movie Time		No
ScheduleMovieDate	Date	Schedule Movie Date		No
RoomId	Id	Room Id		No
RoomType	Type	Room Type		
MovieId	Id	Movie Id		
MovieName	Name	Movie Name		
TicketSeatsQuantity	Numeric(4.0)	Ticket Seats Quantity		No
TicketPrice	Price	Ticket Price		No
TicketTotalPrice	Price	Ticket Total Price	TicketSeatsQuantity * TicketPrice...	

Run the application and add a couple of ticket sales.

RULES

Look at the rules of the Category and Country transactions.

If you try to leave a Category or Country without a name, an error rule will be triggered preventing the field from being left empty.

RULE REVIEW

The following rules have to be stated in these transactions:

Customer

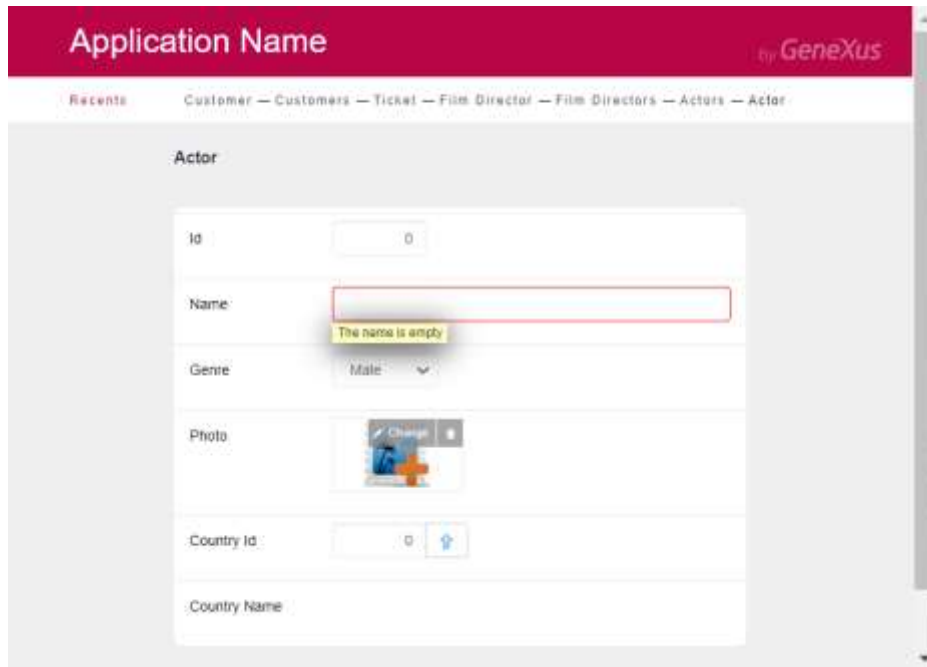
- It should not be possible to save if the name is empty.
- A warning message should be sent if the Email is empty.

Ticket

- By default, the current date should be suggested.
- The Ticket's sale date must be equal to or greater than the current date. It must not be possible to save past dates.
- The date of the movie must not be earlier than the date of purchase. This only applies in case of insertion.

CLIENT SIDE VALIDATION BEHAVIOR [SUGGESTED]

In all transactions, I should not be allowed to continue to the next field when an error rule is triggered. I'm also interested in having all rule messages displayed at the bottom of the field.



ADDING NEW RULES

When tickets are sold for a movie, if the sale is associated with a customer, the points generated for that purchase must be added to the customer's account; each ticket purchase generates points (10% of the total). Write below what you need to program:

Customers Transaction: Add a new CustomerPoints attribute to save the points generated per purchase.

Name	Type	Description	Formula	Nullable
Customer	Customer	Customer		
CustomerId	Numeric(4.0)	Customer Id		No
CustomerName	Name	Customer Name		No
CustomerEmail	Email, GeneXus	Customer Email		No
CustomerVIP	Boolean	Customer VIP		No
CustomerPoints	Numeric(4.0)	Customer Points		No

Ticket Transaction: Add to the transaction definition the attribute that stores the customers' points generated per purchase.

● TicketPrice	Price	Ticket Price	
🔗 TicketTotalPrice	Price	Ticket Total Price	TicketSeatsQuantity * TicketPrice
🔗 TicketPoints	Numeric(4.0)	Ticket Points	

Rules of the Ticket transaction: How should the Add rule be programmed?

```
add( _____, _____);
```

Make the change in GeneXus to test this behavior.

Another requirement is to be able to record the rewards that customers can redeem. These rewards belong exclusively to the client and must have an identifier, a brief description and their cost in points.

Every time a new reward is registered, it is necessary to evaluate whether the customer has enough points to allow redeeming the reward, and otherwise send a message.

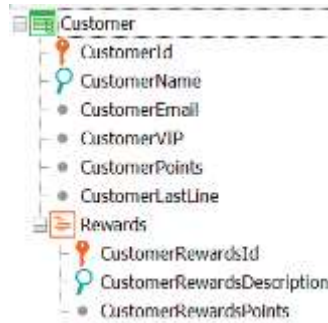
NOTE: It is recommended that a serial rule be implemented and that the CustomerRewardsId field be disabled.

Name	Type	Description	Formula	Nullable
Customer	Customer	Customer		
CustomerId	Numeric(4.0)	Customer Id		No
CustomerName	Name	Customer Name		No
CustomerEmail	Email, GeneXus	Customer Email		No
CustomerVIP	Boolean	Customer VIP		No
CustomerPoints	Numeric(4.0)	Customer Points		No
CustomerLastLine	Numeric(4.0)	Customer Last Line		No
Rewards	Rewards	Rewards		
CustomerRewardsId	Id	Customer Rewards Id		No
CustomerRewardsDescription	Character(50)	Customer Rewards Descript...		No
CustomerRewardsPoints	Numeric(4.0)	Customer Rewards Points		No

TRIGGERING EVENTS

NAVIGATION TREE

Considering the following structure for the Customer transaction:



Below is a list of the rules programmed in this transaction.

Note: it is not necessary to replicate these rules in the Rules section.

Place the correct order of execution in the parentheses:

```

( )error('The name is empty') if CustomerName.IsEmpty();
( )msg('The email is empty') if CustomerEmail.IsEmpty();
( )pPrintCustomerList() on afterComplete;
( )Subtract(CustomerRewardsPoints, CustomerPoints);
( )error("Not enough points") if CustomerPoints < 0;
( )pPrintRewards(CustomerId, CustomerRewardsId) on AfterInsert;
( )serial(CustomerRewardsId, CustomerLastLine, 1);
( )noaccept(CustomerRewardsId);
( )pCheckpoints(CustomerId) if update on AfterInsert;
  
```

TRIGGERING MOMENTS

When saving a new Schedule, it is required that a summary of the new record be printed:



We are also asked to check that when adding a movie at least two actors are registered; otherwise, the application should not allow us to save the movie.

Suppose that the cinema wants to delete a Customer; however, remember that we have customers associated with one or more ticket purchases. In order to solve this situation, the following is proposed:

Customer Rules:

pDeleteTicket(CustomerId) if delete on AfterValidate;

DeleteTicket Procedure Rules:

parm(CustomerId);

DeleteTicket Procedure Source:

```
For each Ticket
    delete
endfor
```

Is the rule in the Customer transaction correctly defined or what changes need to be implemented to make it work correctly?

CONTROLS IN COMPOUND FK

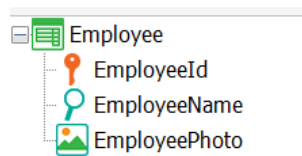
Sometimes it is possible to sell “open” movie tickets that allow the customer to choose when to see the movie. That is, a ticket can be sold for a certain viewing room today, but the customer can go to the cinema at a time of his/her choosing.

What change should be made in the ticket transaction to disregard the referential integrity check in the Schedule table between RoomId, ScheduleMovieDate and ScheduleMovieTime?

Explain it below:

ADDING NEW ATTRIBUTES

Another requirement is to add an employee catalog. Add the following transaction to the KB.



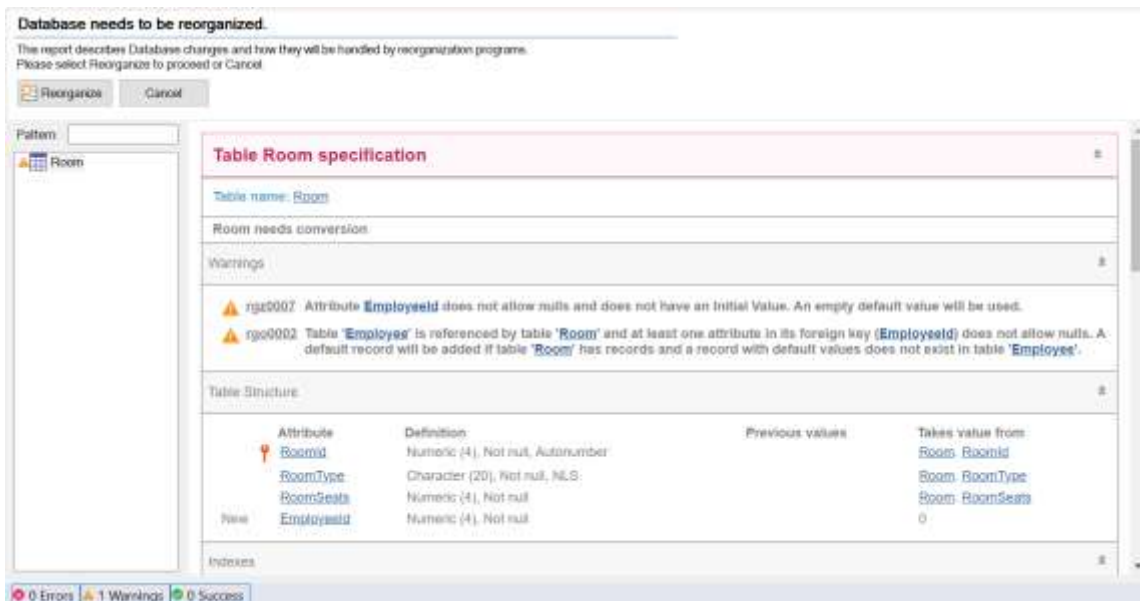
Run the application and add a couple of records for the employee.

The following change is requested: each room is required to have one employee in charge and each employee can be in charge of one or more rooms. **How would you model this in GeneXus taking into account the Employee and Room transactions?**

Write the answer: _____

Replicate in GeneXus what you have just analyzed.

The impact analysis indicates the following. What would happen in the Employee table if we select Reorganize?



If we don't want this to happen, what can we do to correct it?

PARALLEL TRANSACTIONS [SUGGESTED]

Film companies share certain preview information with all Cinemas. The following is shared: The name of the movie, year, poster, director, country and category; information about the description and actors is kept confidential until some time later, so it is not entered into the system before it is included in the Schedule.

To record the initial data of the movie and not see the other fields, a parallel transaction is required where only the information that is available is recorded.

SUBTYPES

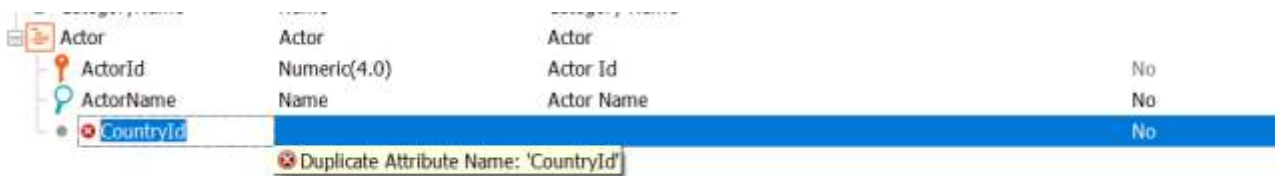
The following application enhancements are required:

ADDING A COUNTRY TO THE MOVIE

Information about the country of the movie is to be added. Make this change.

Remember that a Movie can have 1 or more actors associated with it. In addition, add information about the actor's country.

We are faced with the following problem: We can't repeat two attribute names in the same transaction. **What should we change in our transaction structure to save the Country of the Movie and also make reference to the Actor's Country?** Make this change in the IDE.



RECURSIVE SUBTYPES

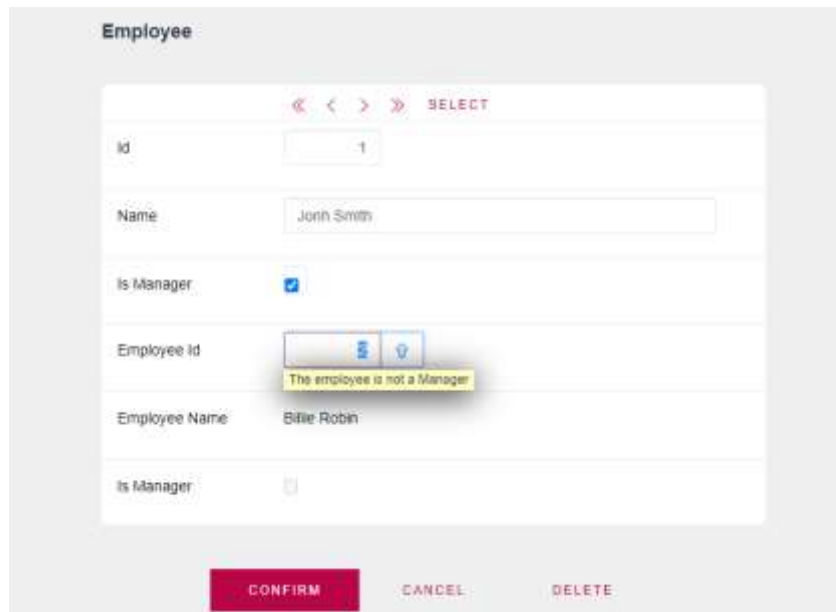
A transaction is requested where we can view information about the cinema's Employees and their manager. How should it be modeled if we take into account that the manager is also an employee?

The Employee transaction should give an error if we try to assign as manager an employee who doesn't hold this position. In addition, it shouldn't be possible to save the information if the manager and the employee are the same person.

How should these rules be programmed? Also write the rules in GeneXus to test this behavior.

```
error("The employee and the Manager can't be the same")
  if _____ = _____;
```

```
error("The employee is not a Manager")
  if _____ and _____;
```



FORMULAS

Previously, the price of the tickets was defined at the time of purchase; however, now the price will be defined according to the viewing room selected. For this, a price history must be saved for the viewing room and the date on which the price was saved (NOTE: it must not be possible to repeat the date). The value that will be used to calculate the ticket will be that of the latest date.

According to the above, which would be the correct option to model the requirement?



Which aggregation formula would you use to return the price value according to the latest date?

RoomUpdatedPrice = _____

Once the Room transaction has been updated, what changes should you make to Ticket? Is Ticket Price still valid?

Do what is necessary in GeneXus and try it at runtime.

Within the Movie transaction, a field is to be added that allows knowing whether the movie is Premium.

For Premium movies, a 7% surcharge must be added at the time of purchase. If the movie doesn't have this field specified, then the normal price must be applied.

Room Type	VIP
Movie id	1
Movie Name	Avengers End Game
Seats Quantity	<input type="text" value="1"/>
Room Updated Price	135.00
Total Price	144.45
Points	14

For each Schedule record, the tickets sold and seats available should be known.

Room Id	Movie Date	Movie Time	Movie Id	Movie Name	Movie Poster	Room Seats	Seats Available	Seats Sell
1	002020	11 AM	1	Avengers End Game		25	25	0 UPDATE DELETE
1	002020	2 PM	1	Avengers End Game		25	25	0 UPDATE DELETE
1	002020	5 PM	6	Harry Potter and the Prisoner of Azkaban		25	25	0 UPDATE DELETE
2	002020	11 AM	7	Wonder Woman		80	80	0 UPDATE DELETE
2	002020	2 PM	7	Wonder Woman		80	80	0 UPDATE DELETE
1	003020	11 AM	2	Call Me if you can		25	25	0 UPDATE DELETE
1	003020	11 AM	3	Forsell Gang		25	23	2 UPDATE DELETE
2	003020	5 PM	2	Call Me if you can		80	80	0 UPDATE DELETE
2	003020	8 PM	5	The Santa Man		80	79	1 UPDATE DELETE

In case all the tickets for that room have already been sold, the system should not allow saving the purchase.

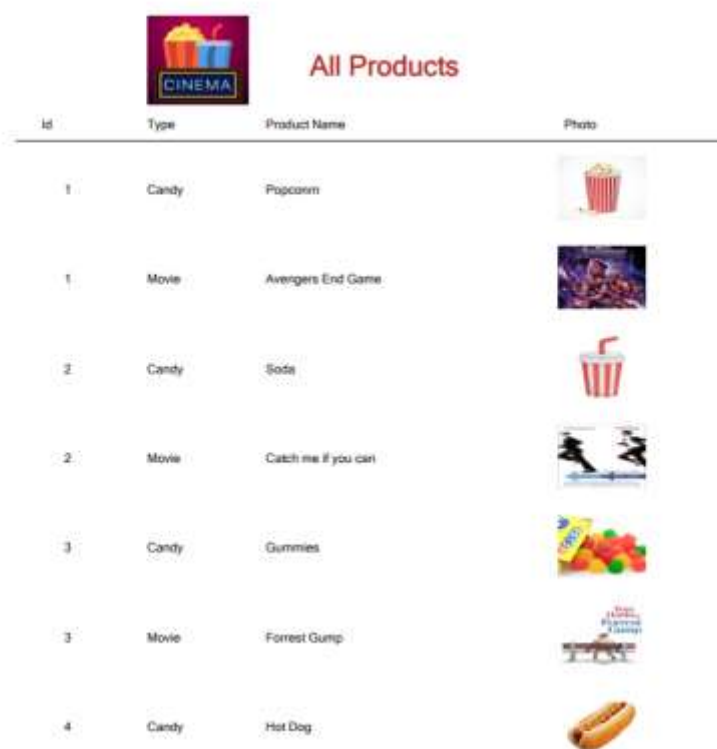
DYNAMIC TRANSACTIONS

Another requirement of the Cinema system is to have a catalog of the products sold at the candy stand. The product identifier, its name, an image and its price will be saved.






In order to know all the products offered by the Cinema, a report listing all the movies and candy products is required. The following are required: their identifier, name, photo and type (whether it is a movie or candy item).

Tip

Use a dynamic transaction to retrieve the information from both catalogs and be able to list this information in the previously requested report.

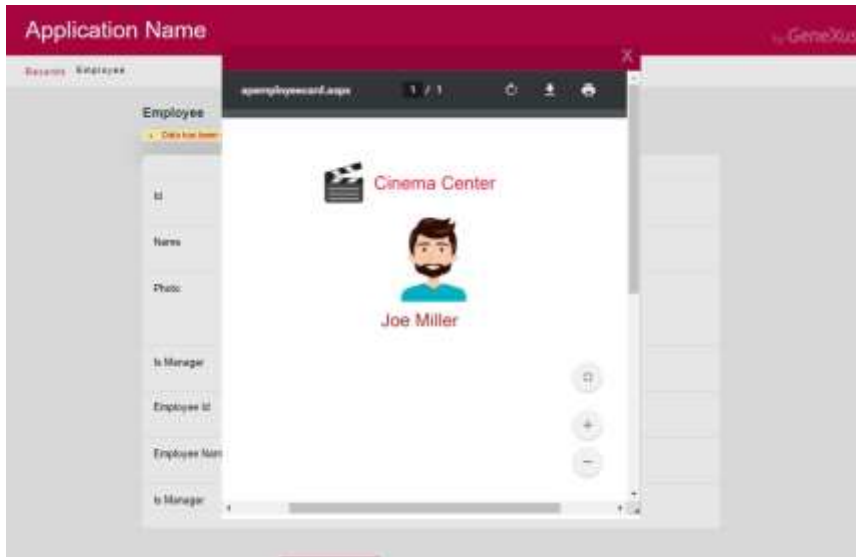


The screenshot shows a report titled "All Products" with a "CINEMA" logo. The report contains a table with the following data:

Id	Type	Product Name	Photo
1	Candy	Popcorn	
1	Movie	Avengers End Game	
2	Candy	Soda	
2	Movie	Catch me if you can	
3	Candy	Gummies	
3	Movie	Forrest Gump	
4	Candy	Hot Dog	

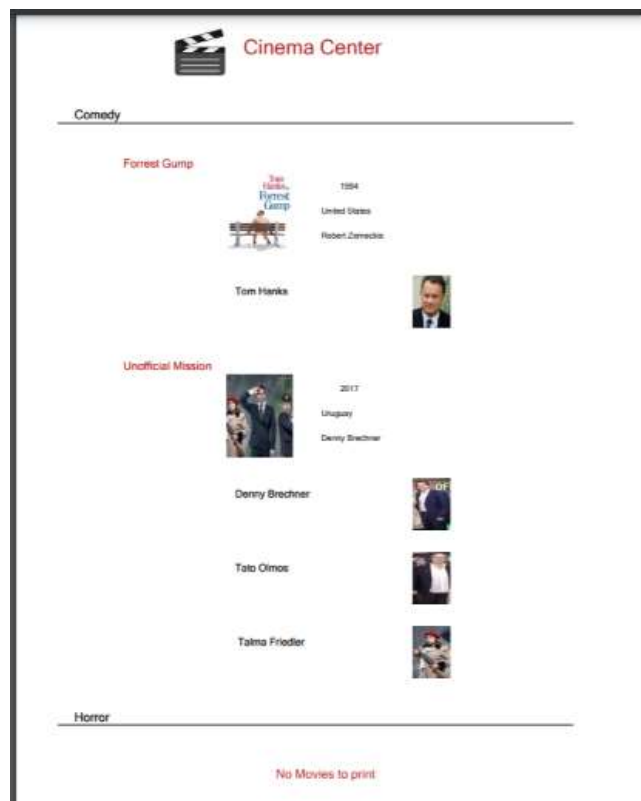
EVENTS IN TRANSACTIONS

The cinema system requires that when a new employee is saved, a popup should appear on the screen with his or her details:



PROCEDURES AND LISTS

We are asked for a report in the following format. Place the categories and their related movies, as well as the actors who appear in those movies. If there are no movies associated with the category, a message should be displayed to inform that there is no data:

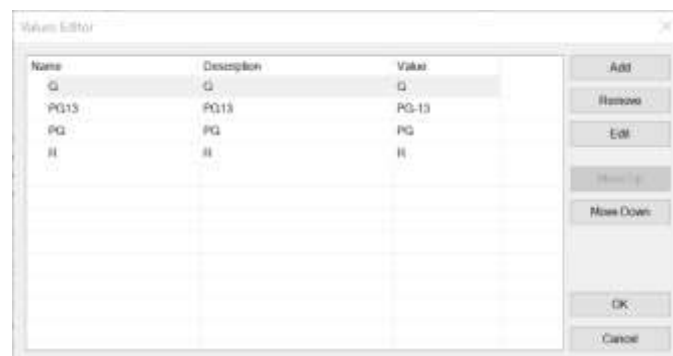


We are asked for a list with the same format as the previous one with the restriction that only the categories with associated movies be listed. How would you achieve the above? Write the code below with the changes to be made.

Do a Save As of the first procedure and make the appropriate modifications to be able to receive a Category as a parameter from a web panel. The procedure must show the information of the selected category and, if no category is selected, all the information of the Categories with their corresponding Movies and Actors must be shown.

SUBROUTINES

In the Movie transaction, add a new attribute called MovieRating where we will store the age rating for each movie (G, PG, PG-13, R). For this, create an enumerated domain called Rating that will contain the following data:



The screenshot shows a 'Values Editor' dialog box with a table containing the following data:

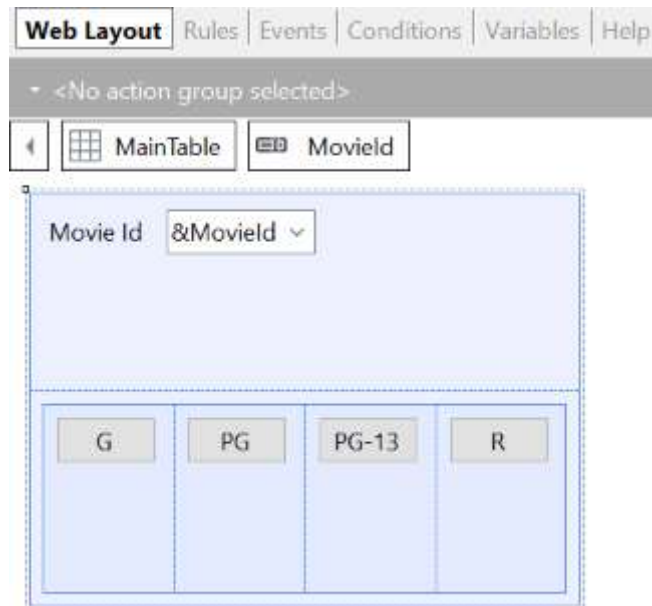
Name	Description	Value
G	G	G
PG13	PG13	PG-13
PG	PG	PG
R	R	R

On the right side of the dialog, there are buttons for 'Add', 'Remove', 'Edit', 'Move Up', 'Move Down', 'OK', and 'Cancel'.

About enumerated domains:

Names containing hyphens or other special characters are not allowed in the "Name" and "Description" columns. To enter the PG-13 rating classification, do so without the hyphen.

Next, create the following web panel where we will select a movie and click a button to update its rating according to the selected value.



Note that certain code will need to be repeated for each button, so we can resort to subroutines to use the same code but update as selected.

```

1 Event 'G'
2     &Rating = Rating.G
3     do "UpdateRating"
4 Endevent

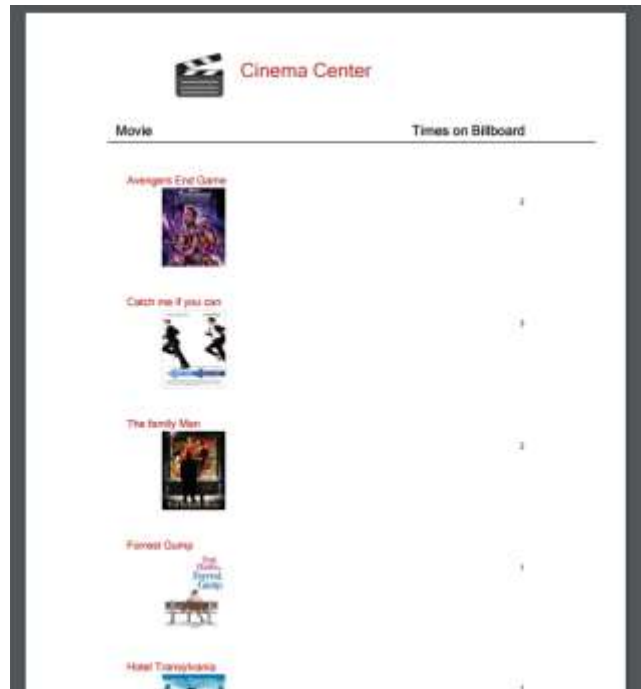
```

Tip

Use a responsive table to place the buttons, and change the class of the buttons to one that matches the appearance of the program, for example: BTN Cancel.

UNIQUE CLAUSE

Another requirement of the Cinema is to view a report of the movies that appear on the schedule, without repetitions and with the number of times they have appeared on the schedule.



A new requirement is to be able to view the above information in a web panel. Place a button where you can query the above list as a popup.

Movies Schedule	
Movie Name	qty
Avengers End Game	4
Catch me if you can	3
Forrest Gump	1
Hotel Transylvania	2
The family Man	2
Harry Potter and the Prisoner of Azkaban	3
Unofficial Mission	3
Inception	2

[VIEW REPORT](#)

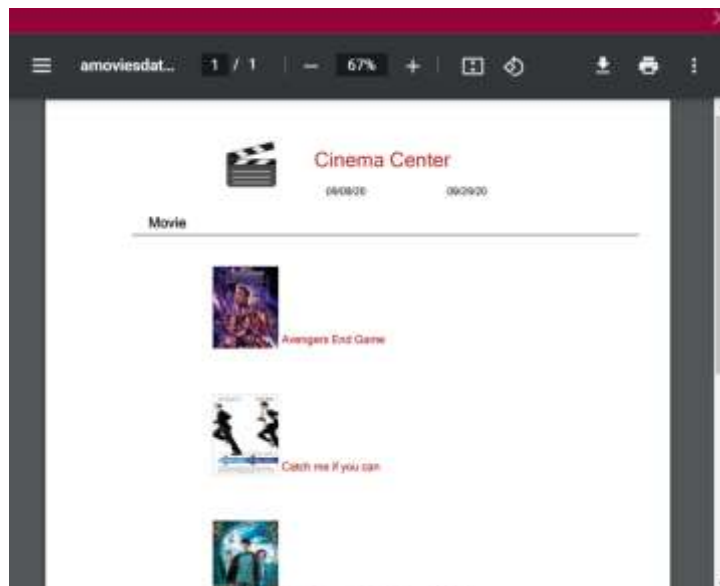
Tip

Change the Grid classes to WorkWith and the Button class to Btn Cancel to enhance the application's design.

DATA SELECTOR

A web screen is required that allows selecting a date range. Then, then by pressing a button it should show as a popup the list of all the movies that appear on the schedule, but only those movies that are in the selected date range.

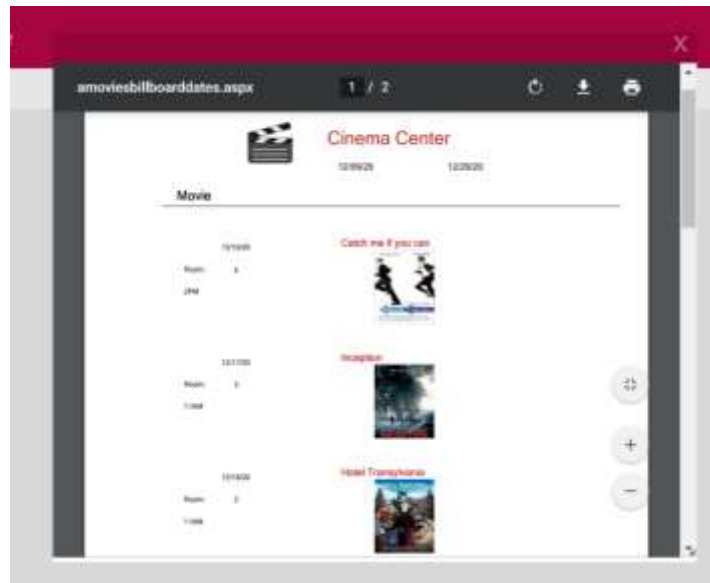
What is the resulting navigation? Explain it...



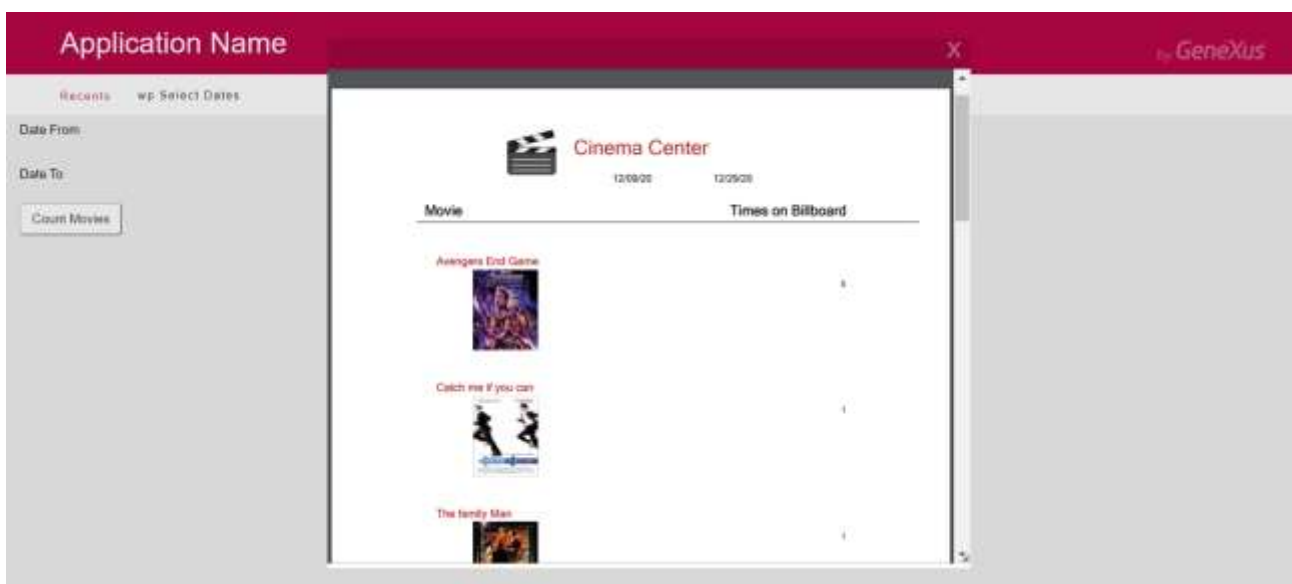
NOTE: Since this is a query that will be frequently used in the Cinemas system, generate a new data selector object for later use.

In the same web screen, another popup list is to be generated to query the information of the movies that are going to be shown, as well as their time, room and date, within a certain date range.

What is the difference between this resulting navigation and the previous one? Explain it...



Additionally, it should allow viewing a popup list with the movies on the schedule, without repetitions and counting only the number of times they have appeared on the schedule in that date range.



What is the difference between this resulting navigation and the previous ones? Explain it...

PAGING CLAUSES

In the Work With screen associated with the Schedule, I am only interested in viewing 5 records at a time, and I want the output format to be an infinite scroll. Apply the requested change.

PAGING CLAUSES IN PROCEDURES

A screen is required where the user can call a list with the actors' information. In particular, the screen must allow selecting how many records to skip and how many to display later. Example: "I want to skip the first 3 records and be able to access the next 5 to work with that data."



ID	Name	Country	Photo
4	Daisy Ridley	United Kingdom	
5	Emma Watson	United Kingdom	
6	Robert Downey Jr	United States	
7	Chris Hemsworth	Australia	
8	Adam Sandler	United States	

BUSINESS COMPONENT

We will perform a series of operations on the database through Business Components.

UPDATING EMPLOYEES IN CHARGE

A specialized screen is required to select a room type (VIP, 3D, etc.) and select an employee. The employee will be in charge of all the rooms of that type.

The screenshot shows a web application interface. At the top, there is a dark red header with the text "Application Name". Below the header, there are two tabs: "Recents" and "Customer — wp Employee Room". The main content area is a light gray box containing two dropdown menus. The first dropdown is labeled "Room Type" and has "Normal" selected. The second dropdown is labeled "Employee Id" and has "Billie Robin" selected. Below these dropdowns is a button labeled "Update Employee".

WORKING WITH MOVIES AND THEIR ACTORS

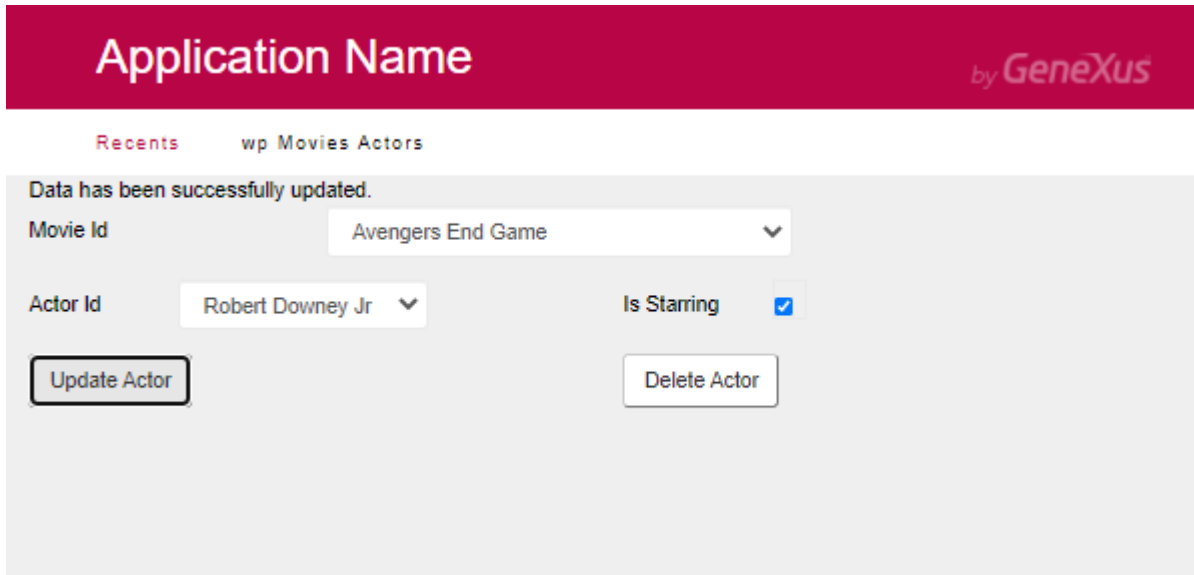
A screen with a special format that allows selecting a movie to work with its actors is requested. Two buttons will be offered:

- One that allows us to modify the actor's role to lead or to remove the lead role.
- Another one that allows deleting the actor from the selected movie.

Tip

Use the &Messages variable of GeneXus to program the system's messages in case of a failed or successful update.

What could you do so that the variable associated with the actor only shows the actors who are in the selected movie?



UPDATING THE DATABASE WITH A PROCEDURE

In the same screen where you made the update of the employees in charge of the rooms according to their type, implement a button that performs the same update mentioned above but now using a procedure.

How could you delete the movies and their associated actors through a procedure?

Insert a new movie (any movie) with its corresponding actors.

Program a button on the screen you created earlier to work with movies and all their actors; this button should delete a selected movie and delete the associated actors. Delete the Movie you have just created.

What happens when you try to delete Movies that are related to other tables?

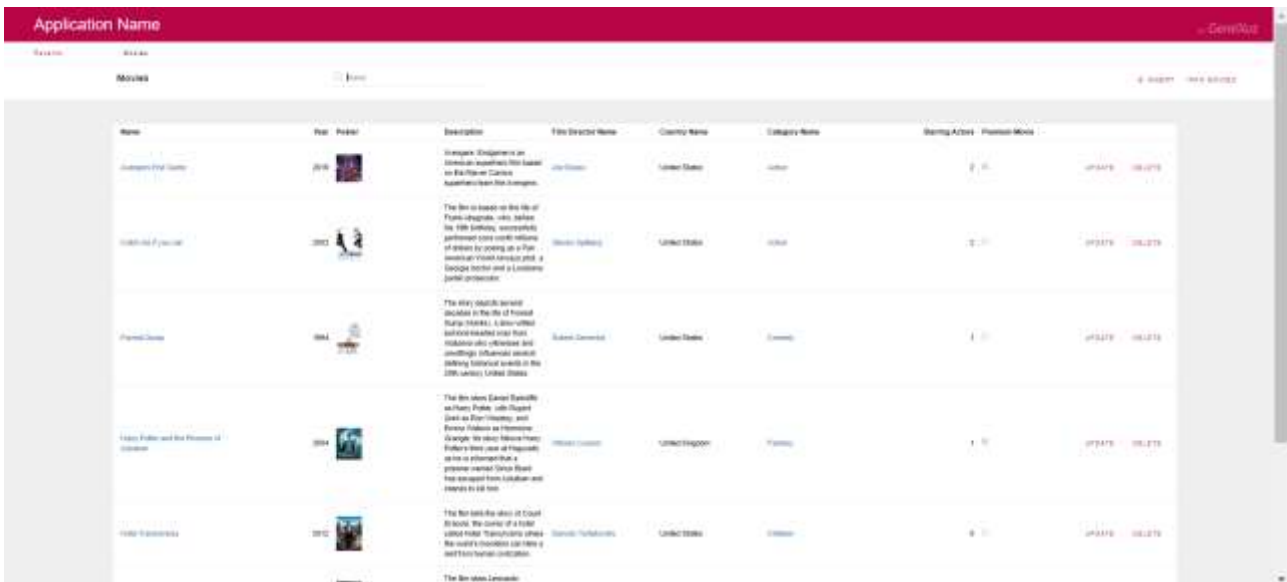
WEB PANELS

Next, we will start generating customized interactive screens that have been requested for the Cinemas system.

WORKING WITH THE PATTERN

In the screen that allows us to work with the movies we are asked to make the following changes:

- IDs must not be displayed.
- A calculated field that shows the number of lead actors in the movie.
- A button that shows the information about the movies previously created where the Category, the movie and the actors in it are displayed.




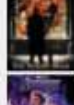



CREATING INTERACTIVE SCREENS

- A screen showing the Schedule information sorted by date of screening is requested. It should be possible to filter by:
 - Date, which must be greater than or equal to today's date. (The system must automatically suggest using today's date.)
 - Time.
 - Type of Room.
 - Name of the movie.

If the filters are empty, all the information of the schedule should be displayed.

For each record, a button is required to show the details of the selected movie when clicked on. In the detail screen, implement a button that allows returning to the previous web panel.

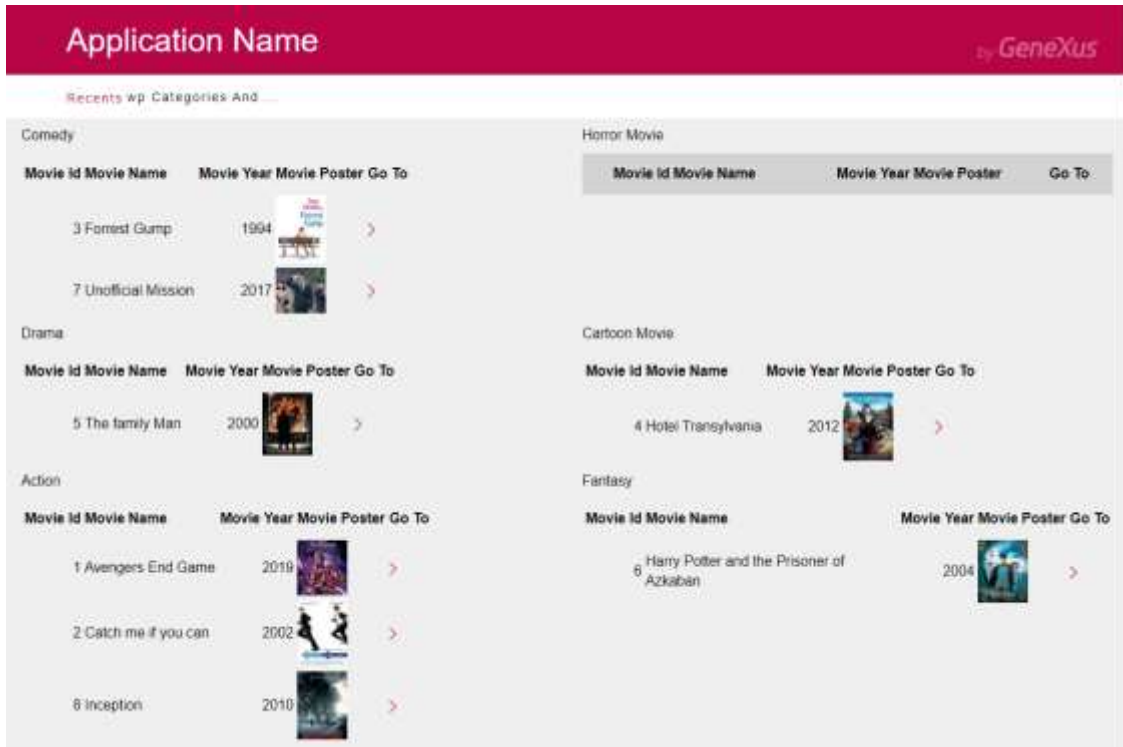
Room Type	Movie Date	Movie Time	Movie Name	Movie Poster
Normal	12/17/20	11 AM	Inception	 >
3D	12/18/20	11 AM	Hotel Transylvania	 >
Normal	12/18/20	11 AM	Harry Potter and the Prisoner of Azkaban	 >
VIP	12/23/20	11 AM	The family Man	 >
VIP	05/06/21	11 AM	Avengers End Game	 >

MULTIPLE GRIDS

A Web Panel is requested that allows viewing all Categories and for each one of them their associated movies.

For each movie, a button is requested that shows the details of the selected movie when clicked on. In the detail screen, implement a button that allows returning to the previous web panel.

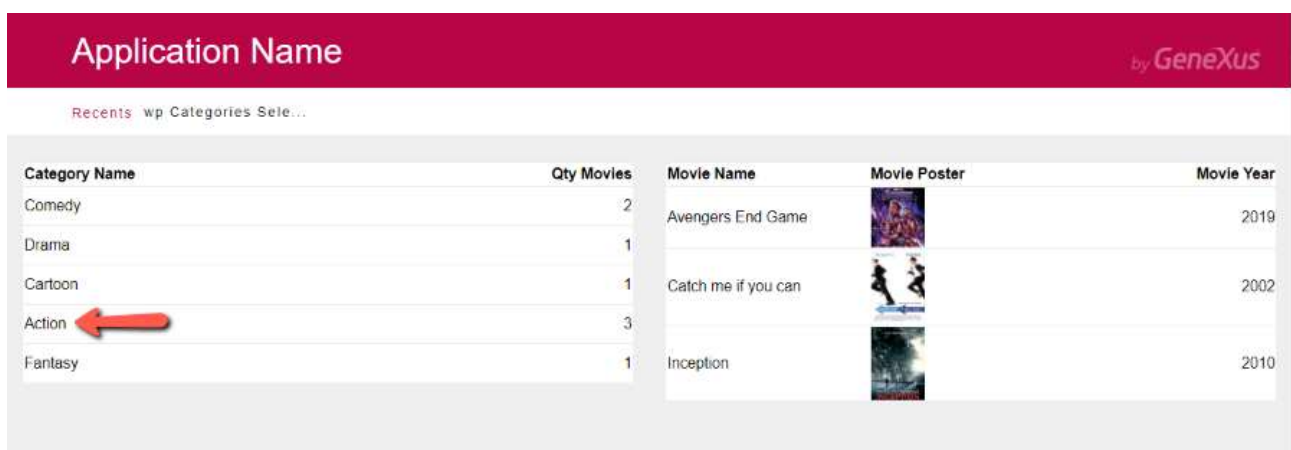
Note: The web panel of the film detail can be reused.



How could you make sure that only categories that have associated movies are displayed?

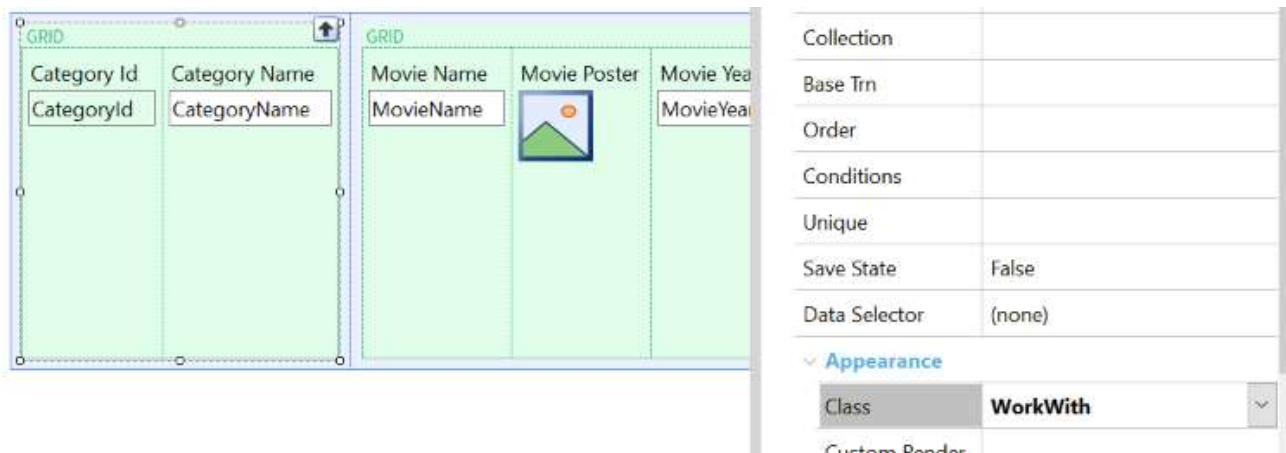
Implement this change.

A screen is required to show the list of categories that have associated movies (without repetitions) and their number of movies. In addition, it should allow selecting a category and then display the associated movies on the same screen.



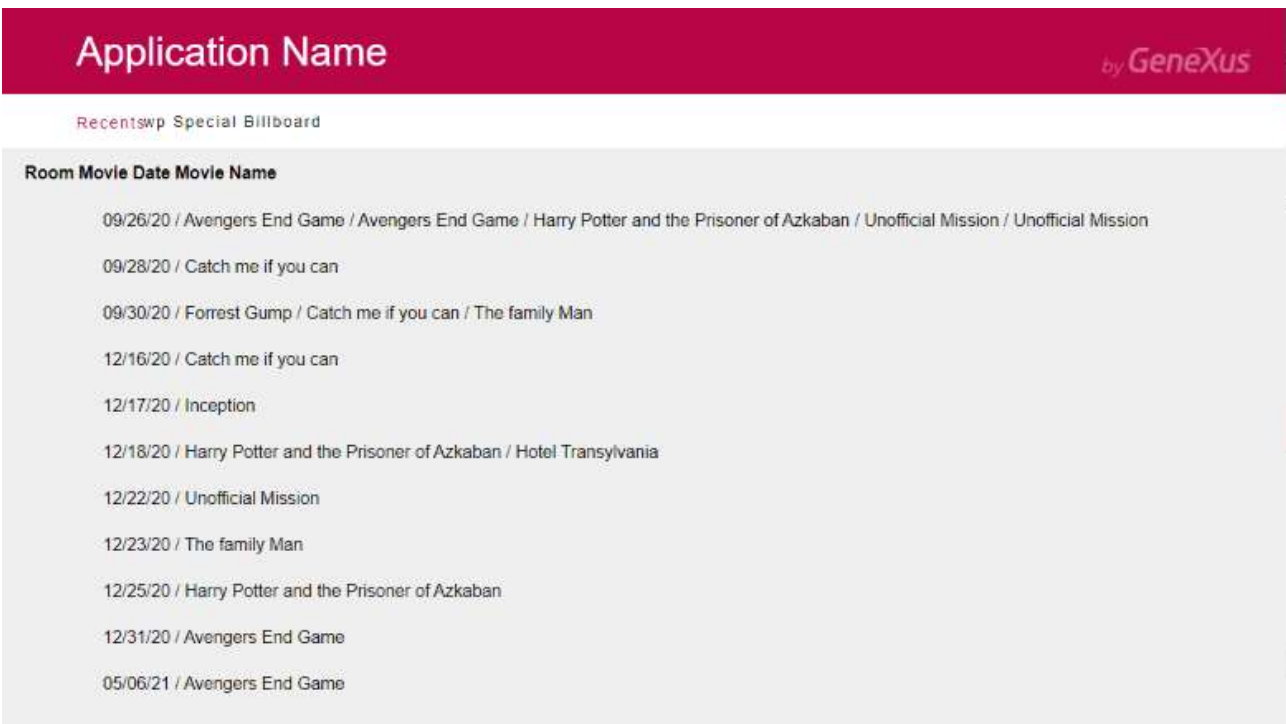
What should be the base table of the Grid that displays the Category information?

Note: It is recommended that the WorkWith class be applied to the Grid to improve its appearance.



GRID WITHOUT BASE TABLE

A special screen is required that shows the movies to be presented for each date. The format is as follows:



LOADING COLLECTIONS

A ranking of Customers is required according to the seats of each ticket they have purchased. To solve the above, use a structure.

Name	Type	Description	Is Collection
sdtRankingCustomer		sdt Ranking Customer	<input checked="" type="checkbox"/>
sdtRankingCustomerItem			
CustomerName	Attribute:CustomerName	Customer Name	<input type="checkbox"/>
CustomerTickets	Numeric(4.0)	Customer Tickets	<input type="checkbox"/>

In addition to the above, when selecting a customer, the screen must show the customer's information in PDF format with the corresponding number of tickets.

Tip

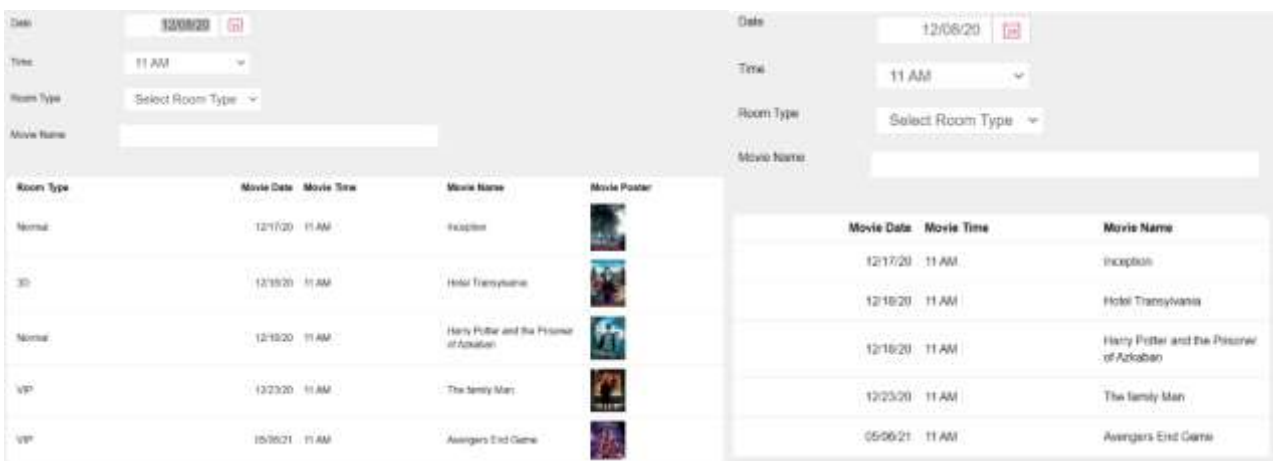
Use the AllowSelection property of the Grid and the OnLineActivate event.

SAVING CONTEXT INFORMATION

Open the web panel containing the entire information of the Schedule. It is required that when returning to this screen after viewing the movie's details, the filter values are saved.

SPECIAL SCREEN FOR TICKET SALES

Make a copy of the previous screen and modify the web panel classes so that it looks similar to the following:

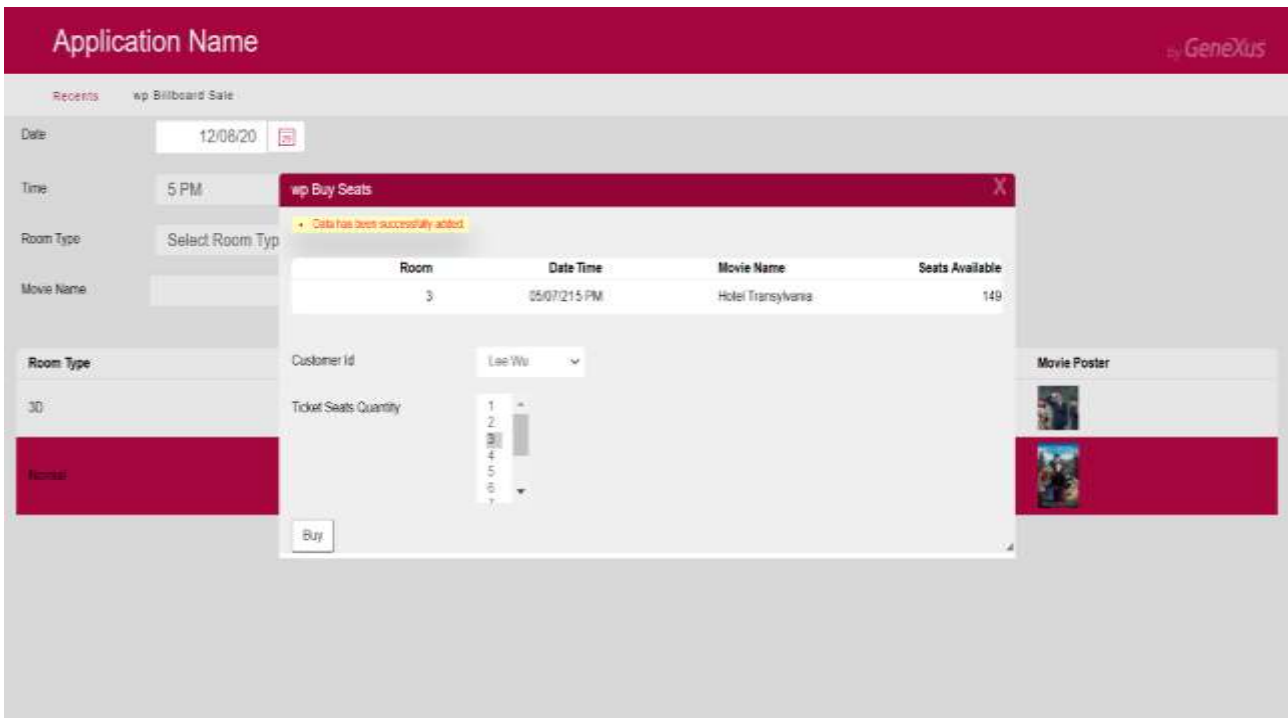
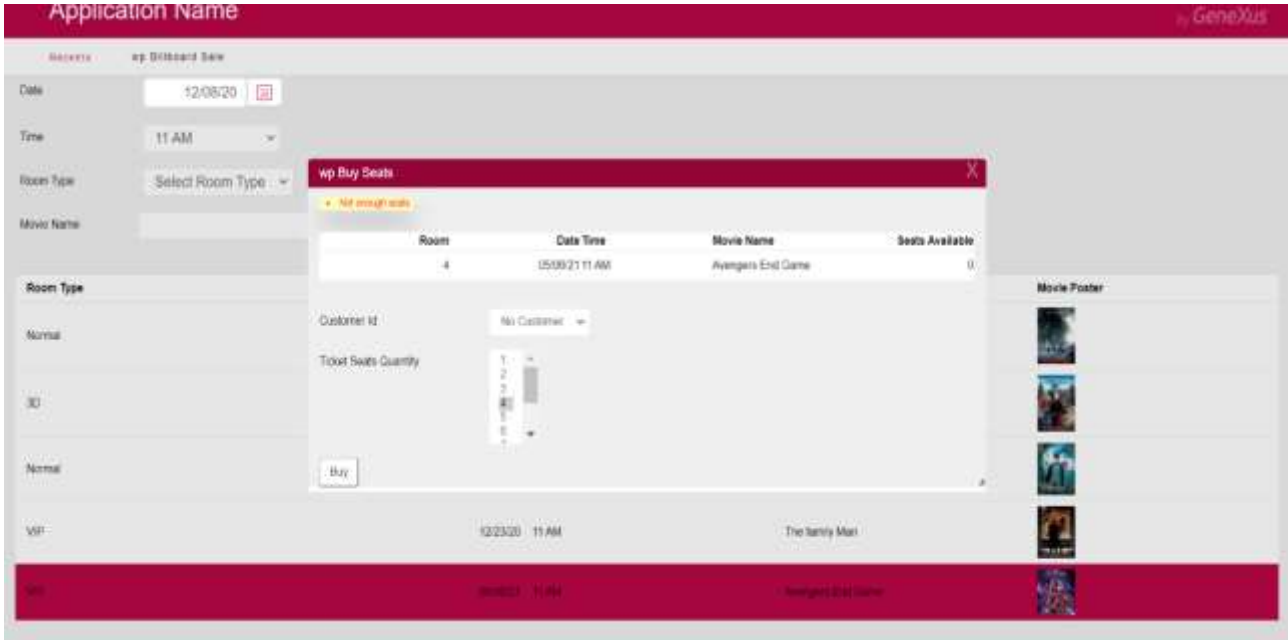


Tip

Use the WorkWith class of the Grid and the WWColumn and WWOOptionalColumn classes.

When selecting a line of the Schedule, a popup window should be displayed with the details of the selected movie. The Employee may or may not select a customer and the number of seats to be

purchased; by clicking on the Buy button, the purchase will be made. Note: sometimes it will not be possible to make the purchase due to some control that we have programmed, so you should see the system messages on the screen:



MULTI-CHANNEL SCREENS: PANELS

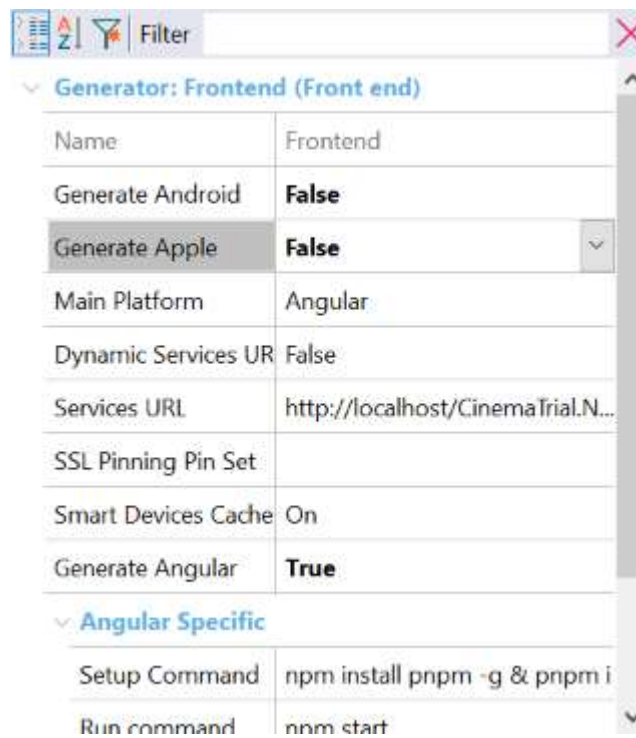
To complete the following exercise, you will need to check that you meet the [necessary requirements to generate in Angular](#).

A panel generated with Angular containing the information of the schedule is required.

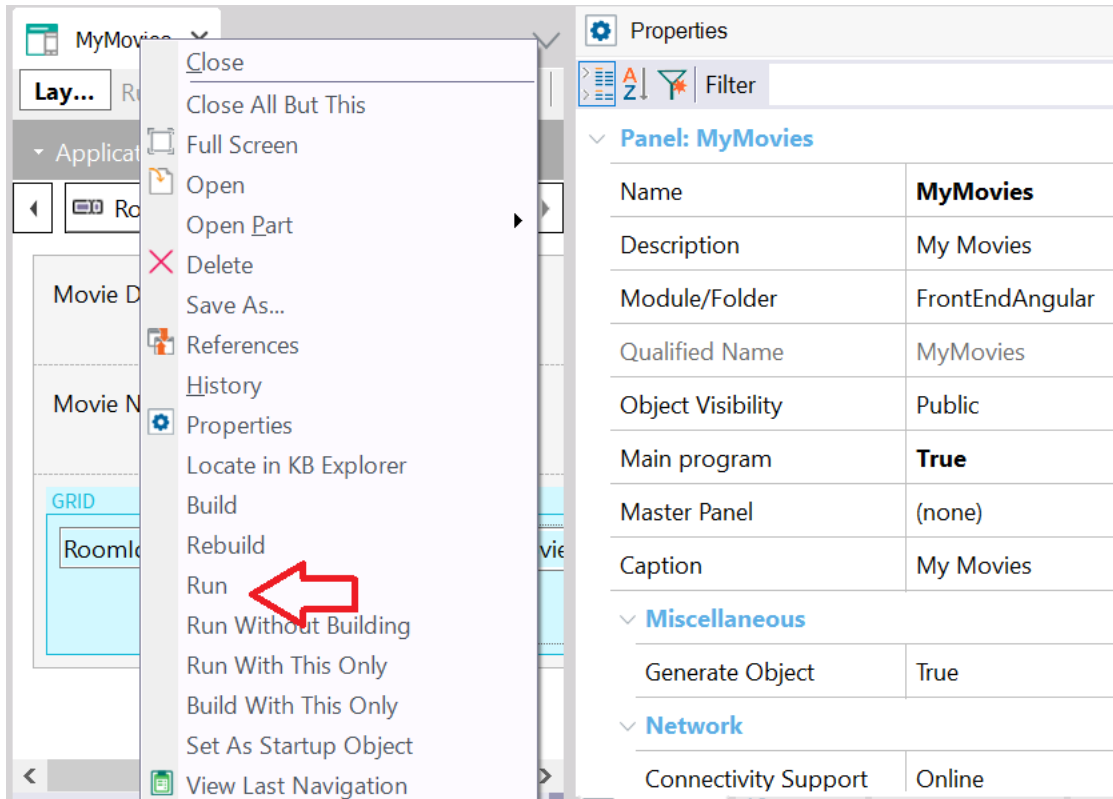
To solve the above, press Ctrl+N and select the Panel object. In the panel, drag a Grid and select the movie's date, time, room, name and poster.

Set the Autogrow property of the Grid to True.

After saving the panel object, to start running on Angular, go to *FrontEnd* in the *KBExplorer*. Change the *Generate Angular* property to *True*; since we are not going to develop for mobile applications at the moment, change the *Generate Android* and *Generate Apple* properties to *False*.



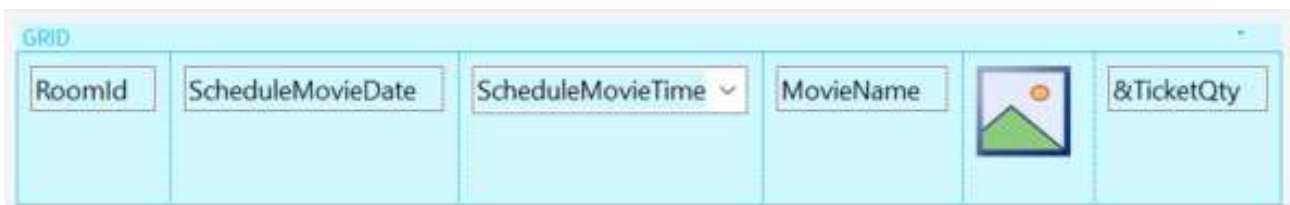
In the panel properties, change the Main Program property to True, right-click on the object name and click on Run to run the application.



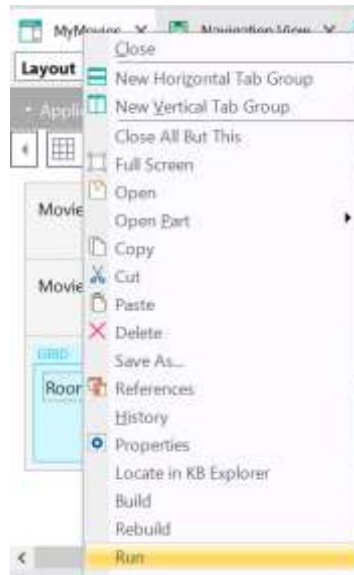
Check that the application looks like this.

3	7/5/2021	5 PM	Hotel Transylvania	
4	6/5/2021	11 AM	Avengers End Game	
6	30/4/2021	8 PM	Inception	

Now add a variable that calculates the number of sales issued by Schedule. *Note: we have an attribute that calculates the number of seats sold; we do not want to show that information, but the number of tickets or sales.*



Program what is needed in the events section and test how the application works.



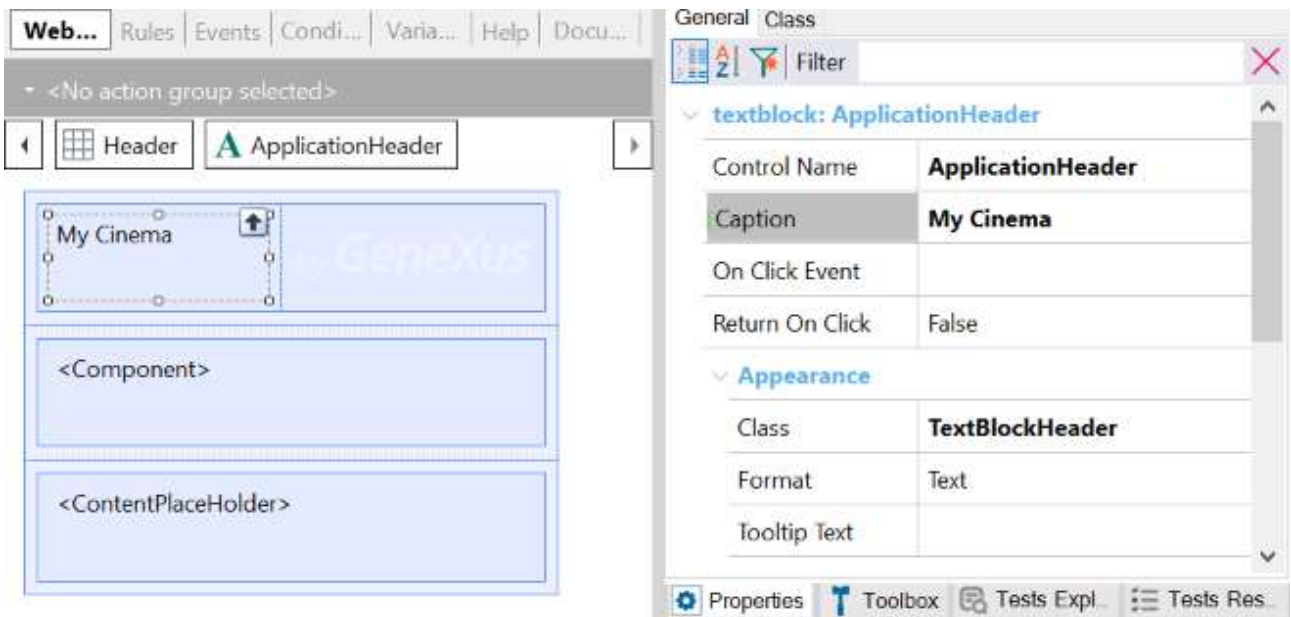
PROGRESSIVE WEB APPLICATIONS [SUGGESTED]

We will slightly change the application's appearance to make the PWA look better. Open the KB Explorer and search in the filter "RWD Master Page." This is the Master Page associated with all the GeneXus objects in our KB.

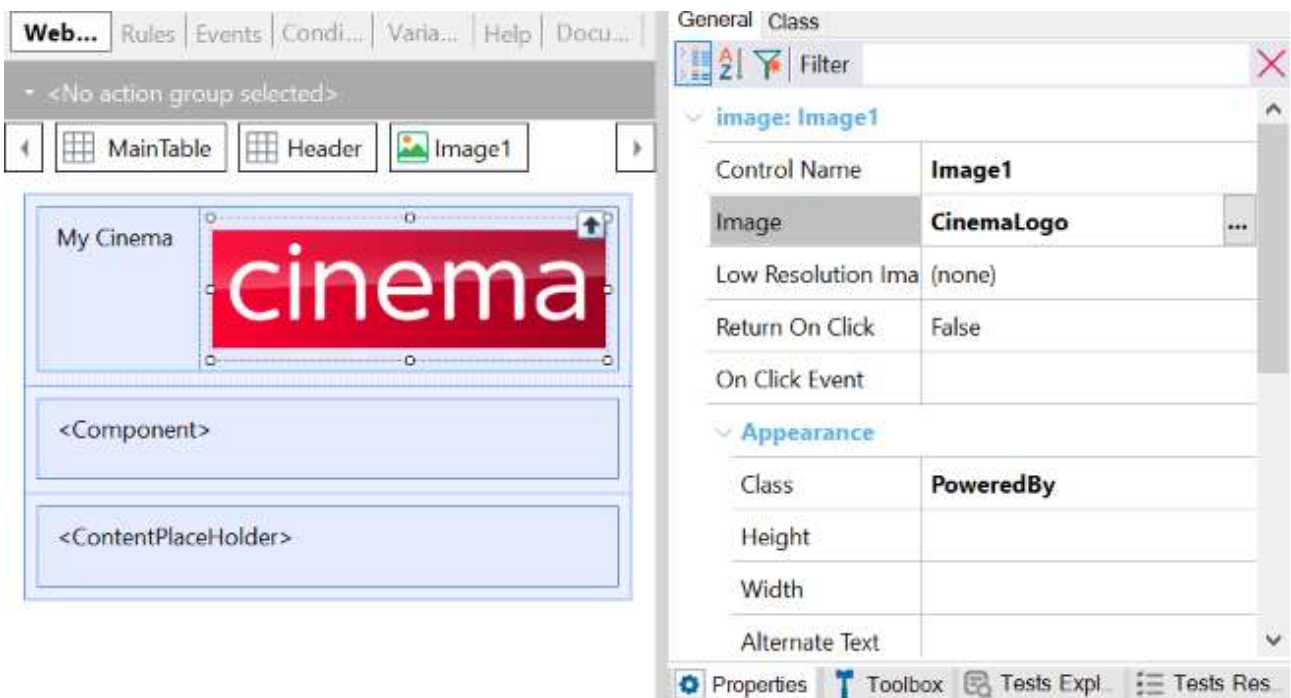
Note that it has three sections:

- A Header containing the name of the application and logo.
- A web component object that is responsible for providing the menu of recent links.
- A content placeholder where the objects of our application are displayed.




In the *Header*, select the *ApplicationHeader* control and change the caption property to the application's name: My Cinema.








Select the *Image1* control and in the Image property select an image as company logo. You can use the images provided for the practice exercises. Check that the *PoweredBy* class is applied.



Make a copy of the web panel where all the schedule information is displayed:

Movie Date: 11/10/20 
 Movie Time: 11 AM 
 Room Type: Select Room Type 
 Movie Name:

Room Type	Movie Date	Movie Time	Movie Name	Movie Poster
Normal	12/17/20	11 AM	Inception	 >
3D	12/18/20	11 AM	Hotel Transylvania	 >
Normal	12/18/20	11 AM	Harry Potter and the Prisoner of Azkaban	 >
VIP	12/23/20	11 AM	The family Man	 >
VIP	05/06/21	11 AM	Avengers End Game	 >

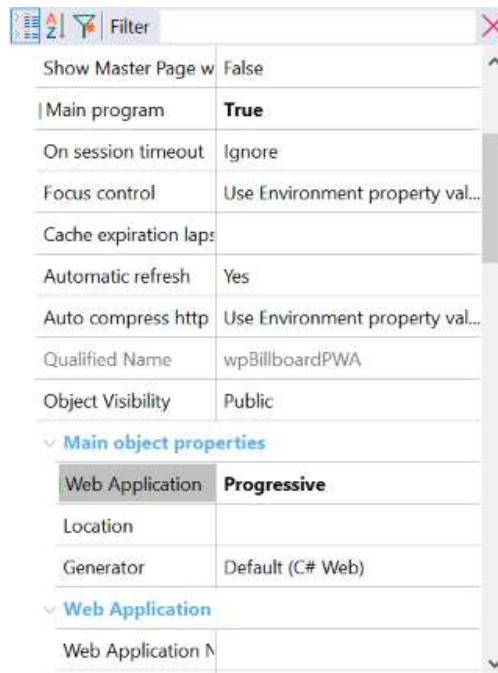
Inside the new web panel, select the Grid and in its class property change it to `WorkWith`.

Also, hide the Movie ID and delete the variable that allows us to see the movie's details and its associated method.

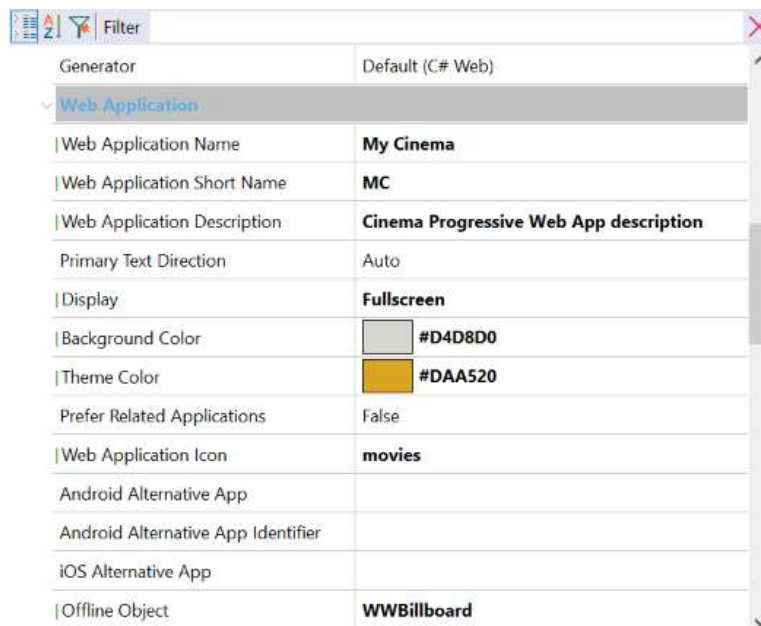
Select the attributes `RoomId`, `ScheduleMovieDate`, `ScheduleMovieTime` and `MovieName`; in the class property, enter `WWColumn`. For each column in the Title property, change their names to shorter ones.

Select the attributes `RoomType`, `MovieId`, and `Movie Poster`; in the class property, enter `WWColumn` `WWOptionalColumn`. Also, enter shorter names for each column.

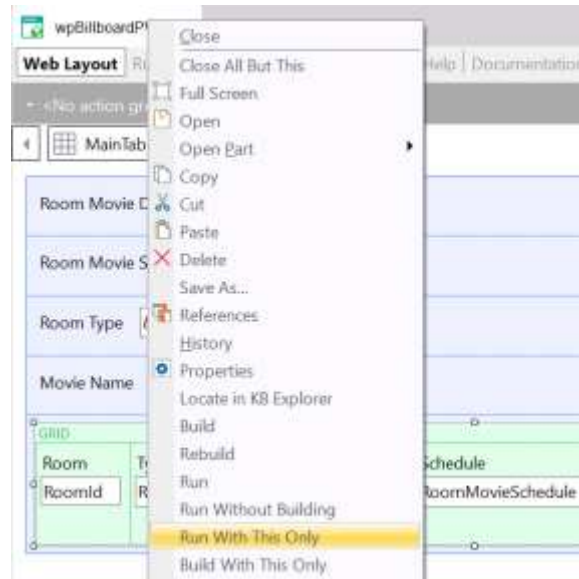
Within the properties of the web panel where the nested Grid is programmed, change `Main Program` to `True`. Note that a new group of properties called `Main object properties` has been activated. In this block, change the `Web Application` property to `Progressive`.



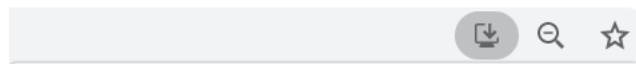
By changing the default property to Progressive, a new group of properties called Web Application has been activated and we will configure them in this way:



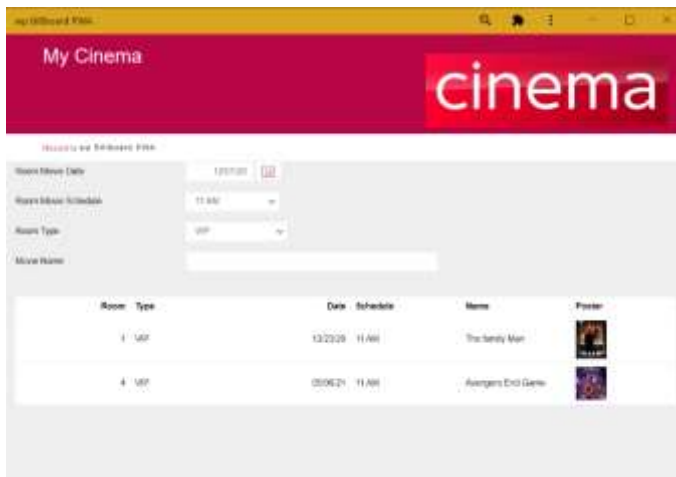
Right-click on the web panel and select the option Run With This Only.



In the browser, note the option to install the application, click on it and go to the shortcut that has been created on the desktop.



Open the application with a double click. Note that the colors, icon, and screen resize correspond to the settings we have made.

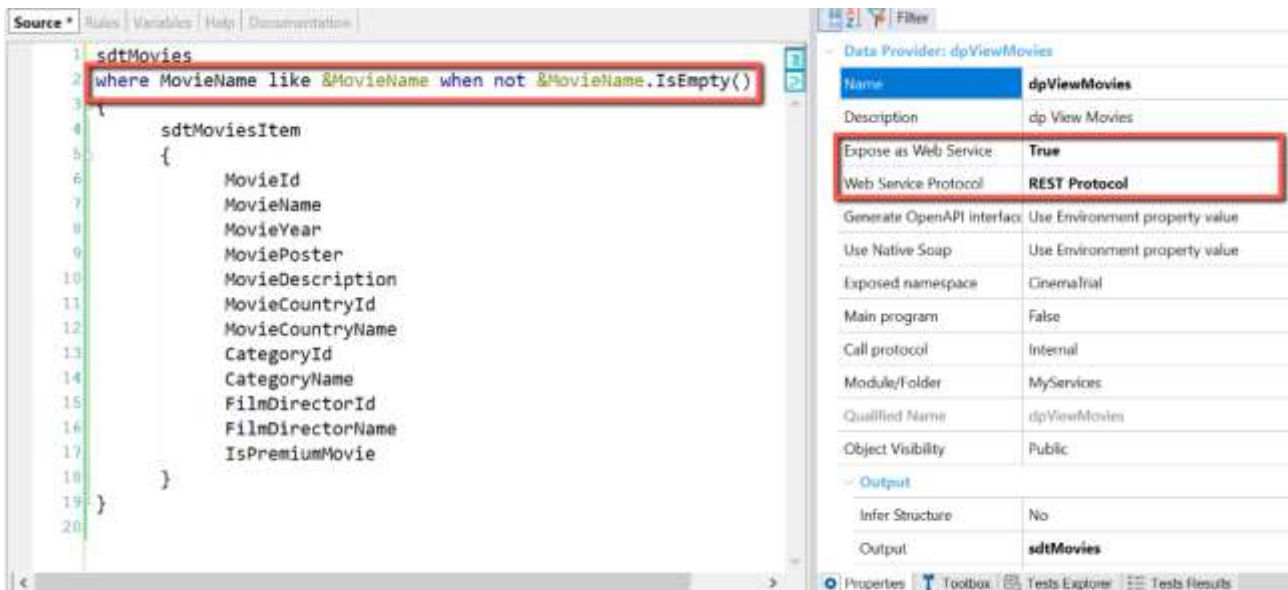


WEB SERVICES [SUGGESTED]

We are requested to publish a web service that allows querying the information of a certain movie by its name.

To implement the above, we are going to create an SDT collection based on the Movie transaction. You can drag the transaction to the structure or create the SDT manually.

Next, drag the structure to a data provider and configure the following:



The condition previously programmed will allow us to retrieve the list of all movies even if the parameter is empty. We will expose the service as Rest.

Place a rule that receives the attribute we will filter by: `parm(in:&MovieName);`

Now right-click on the DP name and select the option “Build With This Only” to expose the service. To confirm that everything is correct, place the following link:

<http://server/baseURL/rest/DataProviderName>

The following should be displayed:

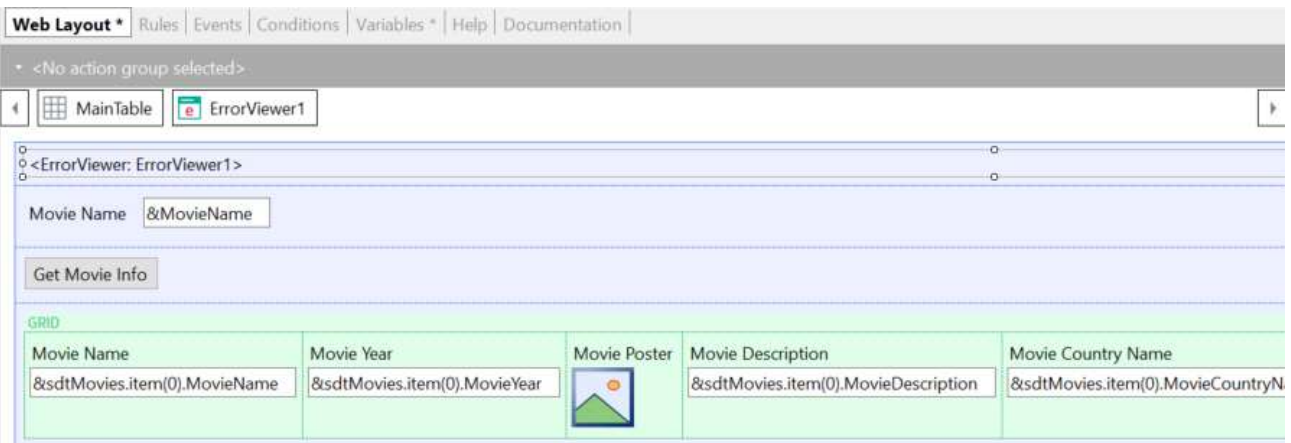
```
[{"MovieId":1,"MovieName":"Avengers: End Game","MovieYear":2019,"MoviePoster":"PublicTempStorage/multimedia/AvengersEnd_Game2160474e79e27112f112ab9770.jpg","MovieDescription":"Avengers: Endgame is an American superhero film based on the Marvel Comics superhero team the Avengers.", "MovieCountryId":2,"MovieCountryName":"United States","CategoryId":5,"CategoryName":"Action","FileDirectorId":1,"FileDirectorName":"Joe Russo","IsPremiumMovie":true}, {"MovieId":2,"MovieName":"Catch me if you can","MovieYear":2002,"MoviePoster":"PublicTempStorage/multimedia/CatchMe_If_You_Can2118e8c71423298186a8a27e0f24.jpg","MovieDescription":"The film is based on the life of Frank Abagnale, who, before his 19th birthday, successfully performed over worth millions of dollars by posing as a Pan American World Airways pilot, a Georgia doctor and a Louisiana parish prosecutor.", "MovieCountryId":2,"MovieCountryName":"United States","CategoryId":5,"CategoryName":"Action","FileDirectorId":3,"FileDirectorName":"Steven Spielberg","IsPremiumMovie":false}, {"MovieId":3,"MovieName":"Forrest Gump","MovieYear":1994,"MoviePoster":"PublicTempStorage/multimedia/ForrestGump_1298aa703bd488abd0a7f906011c.jpg","MovieDescription":"The story depicts several decades in the life of Forrest Gump (Hanks), a slow-witted but kind-hearted man from Alabama who witnesses and unwittingly influences several defining historical events in the 20th century United States.", "MovieCountryId":2,"MovieCountryName":"United States","CategoryId":1,"CategoryName":"Comedy","FileDirectorId":7,"FileDirectorName":"Robert Zemeckis","IsPremiumMovie":false}, {"MovieId":4,"MovieName":"Hotel Transylvania","MovieYear":2012,"MoviePoster":"PublicTempStorage/multimedia/HotelTransylvania_0a5b8a60c549bd20dc0a7e025a5.jpg","MovieDescription":"The film tells the story of Count Dracula, the owner of a hotel called Hotel Transylvania where the world's monsters can take a rest from human civilization.", "MovieCountryId":2,"MovieCountryName":"United States","CategoryId":4,"CategoryName":"Cartoon","FileDirectorId":8,"FileDirectorName":"Genndy Tartakovsky","IsPremiumMovie":false}, {"MovieId":5,"MovieName":"The Family Man","MovieYear":2008,"MoviePoster":"PublicTempStorage/multimedia/FamilyMan_c6b6c30d66a43b7b0738fb6ce18.jpg","MovieDescription":"The film centers on a man who experiences what his life might have been if he had made a different decision earlier in his life.", "MovieCountryId":2,"MovieCountryName":"United States","CategoryId":3,"CategoryName":"Drama","FileDirectorId":4,"FileDirectorName":"Brett Ratner","IsPremiumMovie":false}, {"MovieId":6,"MovieName":"Harry Potter and the Prisoner of Azkaban","MovieYear":2004,"MoviePoster":"PublicTempStorage/multimedia/Prisoner_of_Azkaban_1c12c88a191c4d868a81471ba388.jpg","MovieDescription":"The film stars Daniel Radcliffe as Harry Potter, with Rupert Grint as Ron Weasley, and Emma Watson as Hermione Granger. Its story follows Harry Potter's third year at Hogwarts as he is informed that a prisoner named Sirius Black has escaped from Azkaban and intends to kill him.", "MovieCountryId":4,"MovieCountryName":"United Kingdom","CategoryId":8,"CategoryName":"Fantasy","FileDirectorId":2,"FileDirectorName":"Alfonso Cuaron","IsPremiumMovie":true}, {"MovieId":7,"MovieName":"Unofficial Mission","MovieYear":1987,"MoviePoster":"PublicTempStorage/multimedia/UnofficialMission0f1c117248892b8a4d306223205e41181f.jpg","MovieDescription":"Unofficial Mission is a documentary comedy, directed by Alfonso Cuarero, Denny Brechner and Marcos Hecht.", "MovieCountryId":1,"MovieCountryName":"Uruguay","CategoryId":1,"CategoryName":"Comedy","FileDirectorId":5,"FileDirectorName":"Denny Brechner","IsPremiumMovie":false}, {"MovieId":8,"MovieName":"Inception","MovieYear":2010,"MoviePoster":"PublicTempStorage/multimedia/Inception_438f1d207e3419f98146f0b45c1e82.jpg","MovieDescription":"The film stars Leonardo DiCaprio as a professional thief who steals information by infiltrating the subconscious of his targets.", "MovieCountryId":2,"MovieCountryName":"United States","CategoryId":5,"CategoryName":"Action","FileDirectorId":6,"FileDirectorName":"Christopher Nolan","IsPremiumMovie":false}]
```

Now we will build a web panel that will allow us to consume the Rest service with the HttpClient data type. It can be within the same KB or in a different one. The web panel should have the following structure:

Variables:

Variable	Data type
HttpClient	HttpClient
MovieName	Character(50)
SdtMovies	Based on the SDT previously created
StringParameter	Character(200)

Web Layout



Events

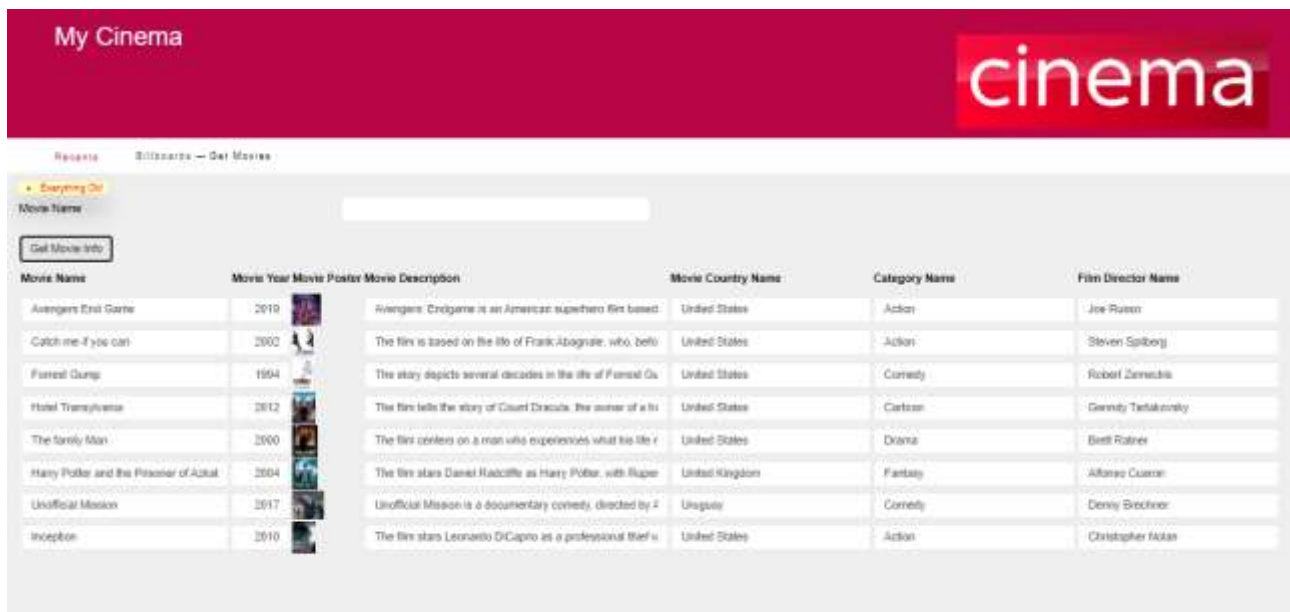
```

1 Event 'Get Movie Info'
2   &httpClient.Host= "localhost"
3   &httpClient.Port = 80
4   &httpClient.BaseUrl = "/KBNAME.NetEnvironment/rest/"
5   &httpClient.secure = 0
6   &StringParameter = 'DPNAME?' + 'MovieName=' + &MovieName
7   &httpClient.Execute('GET', &StringParameter)
8
9
10  if &httpClient.StatusCode = 200
11    msg("Everything Ok!")
12    &sdtMovies.FromJson(&httpClient.ToString())
13  else
14    msg("Error: " + &httpClient.StatusCode.ToString())
15  endif
16 Endevent

```

Run the application and check that the service can be consumed. Type the name of a movie or leave the field empty; in both cases, you should be able to consume the service.

NOTE: Enable the read-only property to True for each field of the SDT that is being printed on the screen.



MONTEVIDEO - URUGUAY
 CIUDAD DE MÉXICO - MÉXICO
 MIAMI - USA
 SÃO PAULO - BRASIL
 TOKYO - JAPAN

Av. Italia 6201- Edif. Los Pinos, P1 (598) 2601 2082
 Hegel N° 221, Piso 2, Polanco V Secc. (52) 55 5255 4733
 7300 N Kendall Drive, Suite 470 (1) 201 603 2022
 Rua Samuel Morse 120 Conj. 141 (55) 11 4858 0300
 2-27-3, Nishi-Gotanda (81) 3 6303 9381
 Shinagawa-ku, Tokyo, 141-0031 (81) 3 6303 9980