

# Search Filters

## Exercise

<b>Outline</b>	<b>1</b>
<b>Hands-on</b>	<b>1</b>
Adding the Search Widgets	1
Saving the Searched Keywords	2
Search Logic	2
GetMovies Aggregate	2
Triggering the Search	3

# Outline

In this exercise, we will expand the functionality of the app with a new filter functionality on the Movies Screen. We want our users to be able to search by movie title, plot summary, and movie genre and we want to make sure that when we change to a different Screen and come back, the search filter is not lost. The search filter should look like this:

### Movies

Select a Movie Genre

▼

Title ↕	Year ↕	Plot Summary ↕	Gross Takings ↕
<a href="#">Star Wars: The Force Awakens</a>	2015	Three decades after the defeat of the Galactic Empire, a new threat arises. The First Order attempts to rule the galaxy and only a ragtag group of heroes can stop them, along with the help of the Resistance.	\$815,843,529.00
<a href="#">Raiders of the Lost Ark</a>	1981	Archaeologist and adventurer Indiana Jones is hired by the US government to find the Ark of the Covenant before the Nazis.	\$242,374,454.00
<a href="#">Schindler's List</a>	1993	In Poland during World War II, Oskar Schindler gradually becomes concerned for his Jewish workforce after witnessing their persecution by the Nazis.	\$16,439,233.00

Note that the search should be done automatically when the users finished typing the title/plot summary keyword, or when they just chose the movie genre. So, we don't want a button/link to trigger the search. Also, the end-user does not need to type the whole movie title and the whole plot summary for the search to return the movies that match the keyword that was typed.

This needs to be done in two steps:

- 1) The UI part, where we need to add input fields to allow searching for the keywords
- 2) The logic part, where we need to adjust the Aggregate that fetches the movies, save the search filters in variables that persist even when we change Screens, and trigger the search logic automatically. For this, we will use Client Variables.

## Hands-on

### Adding the Search Widgets

We will start this exercise by defining the UI part of the search filters. We want to make sure any end-user can search for the movie title and plot summary, as well as for the movie genre. As indicated above, we want to ensure that the search filter looks like the following screenshot:

## Movies

Title ↕	Year ↕	Plot Summary ↕	Gross Takings ↕
<a href="#">Star Wars: The Force Awakens</a>	2015	Three decades after the defeat of the Galactic Empire, a new threat arises. The First Order attempts to rule the galaxy and only a ragtag group of heroes can stop them, along with the help of the Resistance.	\$815,843,529.00
<a href="#">Raiders of the Lost Ark</a>	1981	Archaeologist and adventurer Indiana Jones is hired by the US government to find the Ark of the Covenant before the Nazis.	\$242,374,454.00
<a href="#">Schindler's List</a>	1993	In Poland during World War II, Oskar Schindler gradually becomes concerned for his Jewish workforce after witnessing their persecution by the Nazis.	\$16,439,233.00

**Hint:** For the *(Search for Title or Plot Summary)* and *(Select a Movie Genre)* texts, the input widgets have a property that can help you define those texts.

## Saving the Searched Keywords

Any keyword that the user types for the title/plot summary, or any movie genre that the user selects, needs to be saved somewhere. And there is a special requirement we cannot forget, which is when the end-user searches for something and eventually navigates to a different Screen, the search filter used should not be lost. This means that when the user eventually returns to the Screen, the search criteria are still applied.

The search criteria should only change when the user explicitly changes it on the page (by typing a new title or selecting a new genre), or when the user logs out. So, it's time to save the data in Client Variables to help us support this behavior.

## Search Logic

So far, we have the input widgets to support the UI part of the search and we have the data saved in the corresponding variables. The last thing we need is to trigger the logic that supports the search.

### GetMovies Aggregate

On this Screen, we have the GetMovies Aggregate that fetches all the movies. Now, we need to change it to make sure that it considers the search criteria introduced by the user.

**Hint:** For the title and plot summary, we don't need to have the full title or summary in the search box to find a match. A simple keyword that is contained in the title/plot summary matters. For instance, if I search for 'sta', every movie with 'sta' in the title or plot summary should appear. To accomplish that, we can do the following:

*Movie.Title like "%" + TitleAndSummarySearch + "%" or Movie.PlotSummary like "%" + TitleAndSummarySearch + "%"*

The % are used as wildcards, so this expression can be read as: we want a movie title with some text (possibly empty), followed by the search keyword, followed by more text (possibly empty).

## Triggering the Search

We want this particular search to be triggered automatically, whenever the user types a keyword or selects a genre. For that, we need to use the **OnChange** property of the input fields. This OnChange is similar to the OnClick on Buttons and Links. With the OnClick property, when the user clicks on the widget, it triggers an Action. Here, with the OnChange property, every time a user changes the search criteria, an Action will be triggered.

Now, we just need to implement the logic needed to make sure the movies list is refreshed using the new search criteria.