

Pagination and Sorting

Exercise

Outline	1
Hands-on	1
Pagination	1
Sorting	2
Extra Challenge	2
People Screen	2
Sorting: Ascending vs Descending	2

Outline

In this exercise, we will add pagination and sorting functionality to the Movies Screen of the OSMDb app.

Our app can have multiple movies, which can lead to the Table in the Movies Screen having a lot of records to display. To have a better user experience, we want to add the pagination functionality so that each page only shows five movies at a time.

Then, we want to enable the users to dynamically sort the movies by clicking on any column header in the Table. This means that if the user clicks on the **Title** header, the movies should appear sorted by movie title.

In the end, we have some extra challenges where you'll implement two scenarios:

- When the user clicks on a column header twice in a row, the order of the sorting should change from ascending to descending.
- When the sorting by a column is done, the pagination should reset and go back to the first page.

Let's do it!

Hands-on

Pagination

Let's start by implementing the pagination in the Movies Screen. The pagination should appear below the Table and have 5 movies per page.

To create the pagination we need to:

- Use the pagination widget and set its properties.
- Define the logic that will run whenever the user changes the page.
- Adjust the Aggregate to work with the pagination accordingly.

At the end of this section, the Movies Screen should look like this:

Movies

Select a Movie Genre

▼

Title ↕	Year ↕	Plot Summary ↕	Gross Takings ↕
Star Wars: The Force Awakens	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
Raiders of the Lost Ark	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
Schindler's List	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

Sorting

Now, it's time to enable the users to dynamically sort the movies by clicking on a column header in the Table. This should be done for all the columns in the Table.

To create the sorting we need to:

- Define the OnSort logic for the Table
- Adjust the Aggregate to sort the records dynamically, considering what the user selected at runtime.

Extra Challenge

In this section, we have two extra challenges that improve our pagination and sorting functionality. Are you ready?

People Screen

Apply the same sorting logic to the People Screen, but this time use the OutSystems accelerator selecting the **New OnSort Client Action** option in the Table.

Also, you can follow the same strategy used for the movies and create the pagination for the people records.

Sorting: Ascending vs Descending

At this point, the sorting logic is prepared to sort the records following an ascending order. If you completed the previous challenge, you will see that the logic is a bit more complex to support the descending order.

If not, the objective is this: when the user clicks on a column header for the first time, the records should be sorted in ascending order by the attribute that corresponds to that column. However, when the user clicks on the same header column again, the sorting order should change to descending. To do so, you should use the TableSort variable and add the **DESC** keyword to the sorting criterion.