Pagination and Sorting

Exercício – Como fazer

[Resumo 1](#_Toc96008681)

[Como fazer 1](#_Toc96008682)

[Paginação 1](#_Toc96008683)

[Ordenação 7](#_Toc96008684)

# 

# Resumo

Neste exercício nós vamos adicionar paginação e ordenação na Tela Movies da aplicação OSMDb.

Nossa aplicação pode ter múltiplos filmes, o que pode levar à Table da Tela Movies ter muitos registros para mostrar. Para termos uma melhor experiência de usuário, nós queremos adicionar a funcionalidade de paginação, para que cada página mostre apenas cinco filmes por vez.

Então, queremos habilitar que os usuários possam ordenar dinamicamente os filmes, clicando em qualquer um dos cabeçalhos de coluna na Table. Isso significa que se o usuário clicar no cabeçalho Title, os filmes devem aparecer ordenados pelo título do filme.

No final, nós teremos um desafio extra para implementar dois cenários:

* Quando o usuário clicar em um cabeçalho da coluna duas vezes seguidas, a ordem da classificação deverá mudar de ascendente para descendente.
* Quando a classificação por uma coluna é concluída, a paginação deve ser redefinida para a primeira página.

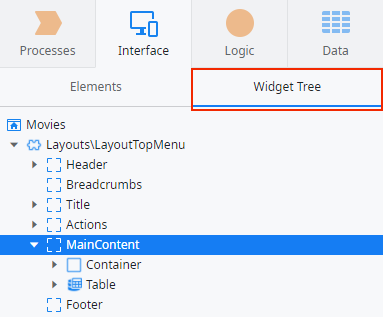
# Como fazer

Nesta seção nós vamos descrever, passo à passo, o exercício *10 – Paginação e Ordenação*.

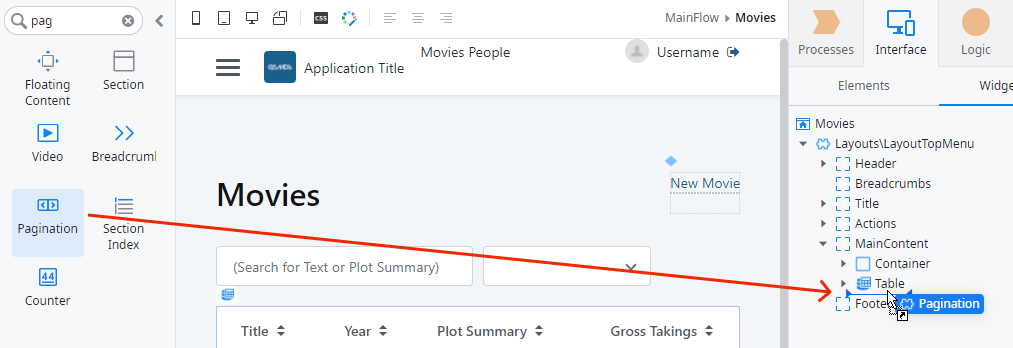
## Paginação

Primeiro, nós vamos implementar a funcionalidade de paginação na Movies Table, usando o widget Pagination. Com isso, nós queremos mostrar apenas cinco filmes por página.

1. Adicione o widget Pagination na Tela Movies, logo após a tabela Movies. Crie 2 Local Variables para representar o Star Index de cada página e o número máximo de filmes que queremos mostrar por página (MaxRecords).
   1. Abra a **Widget Tree** na Tela Movies e expanda o MainContent.

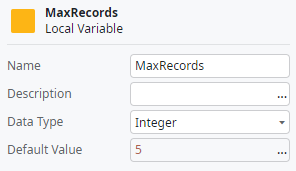


* 1. Na barra de ferramentas de widgets, na esquerda, escreva *pag*, encontre o widget **Pagination**, arraste e solte-o depois da Table.

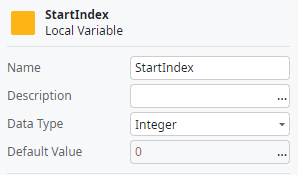


**OBS:** A Pagination possui quatro propriedades que apersentam um erro neste ponto: *StartIndex*, que representa o primeiro registro em cada página; *MaxRecords*, que representa o número de registros em cada página; *TotalCount*, que representa o número total de registros mostrados na Table; Event Handler, que espera que uma Action será acionada sempre que o usuário mudar a página.

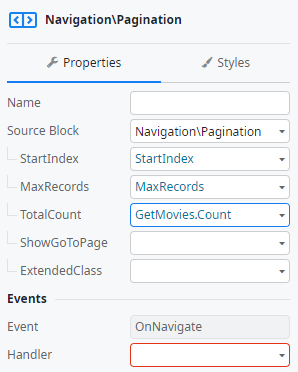
* 1. Crie uma variável local *MaxRecords*. Seu **Data Type** deve ser definido para *Integer* e o **Default Value** deve ser definido para *5*



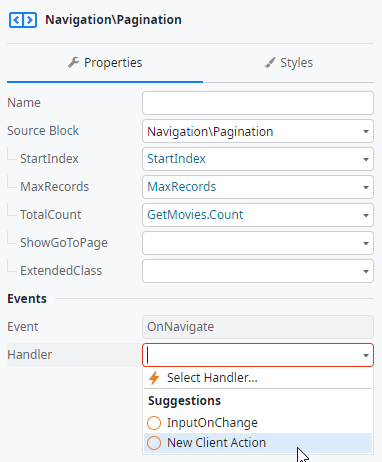
* 1. Crie uma variável local *StartIndex*. Seu **Data Type** deve ser definido para *Integer* e o **Default Value** deve ser definido para *0*



* 1. Selecione o widget Pagination e defina as propriedades **StartIndex** e **MaxRecords** para as respectivas variáveis locais. Além disso, defina o **TotalCount** para *GetMovies.Count*, o qual nos dá o número de registros retornados pelo Aggregate GetMovies.

**

1. Crie a lógica para dar suporte ao comportamento quando um usuário altera a página. A Pagination tem um Handler que deve desencadear uma Client Action. Com esta Action, nós devemos implementar a lógica que muda o Start Index e traga os registros para essa página.
   1. Na mesma caixa de diálogo Properties, expanda a propriedade **Handler** e escolha *(New Client Action)*. Essa Action terá a lógica que suporta a mudança de página.

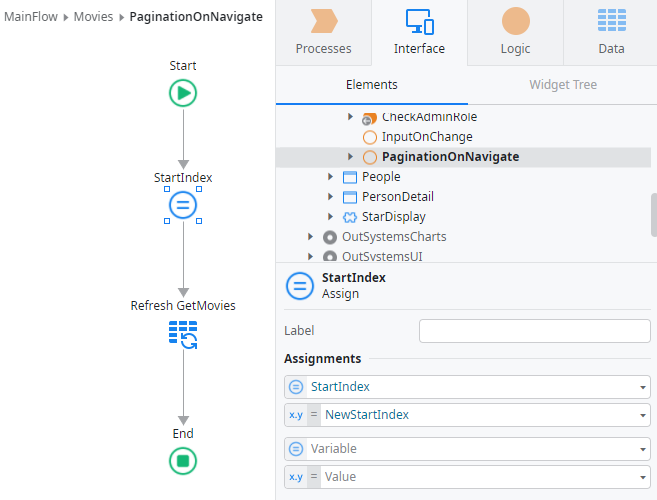


* 1. Uma nova Action, *PaginationOnNavigate*, é criada com um Input Parameter: *NewStartIndex*. Esse Input vai passar para a Action a informação sobre o Start Index da página clicada.
  2. Arraste um Assign e solte-o no fluxo da Action. Defina o assignment para:

*StartIndex = NewStartIndex*

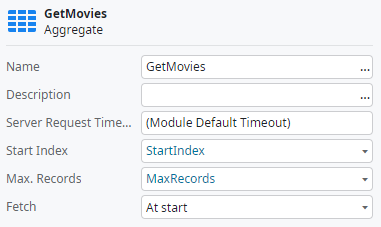
**OBS:** Este assignment armazena o valor do novo índice inicial (start index), dado pela página selecionada pelo usuário, na variável local StartIndex.

* 1. Arraste um **Refresh Data** e solte-o no fluxo da Action. Na nova janela, selecione o Aggregate **GetMovies**. O fluxo da Action deve parecer assim:



**OBS:** O nó Refresh Data desencadeará a execução do Aggregate the GetMovies novamente. A ideia é obter os registros que correspondem à página que foi selecionada. Entretanto, atualizar o Aggregate não é o bastante. Para isto acontecer, precisamos definir o Aggregate para retornar apenas o número de registros por página (MaxRecords), iniciando de um registro em particular ao invés de sempre pelo início (StartIndex).

1. Para finalizar, precisamos mudar o Aggregate **GetMovies** para considerar os valores de StartIndex e de MaxRecords. Apenas defina as propriedades para as respectivas variáveis locais.

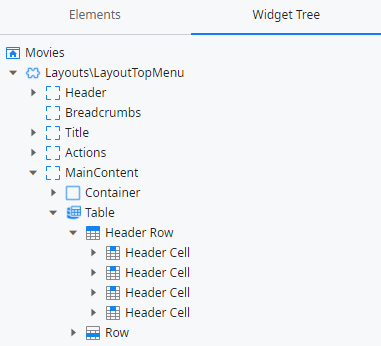


1. Publique o módulo e teste a paginação na Tela Movies. Certifique-se de que os registros sejam alterados de acordo.

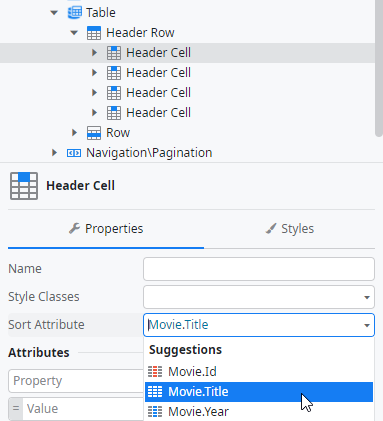
## Ordenação

Na segunda parte desde exercício, vamos implementar a ordenação dinâmica na Movies Table. Basicamente, sempre que um usuário clica no cabeçalho de uma coluna da Table, os resultados aparecerão automaticamente classificados pelos valores dessa coluna.

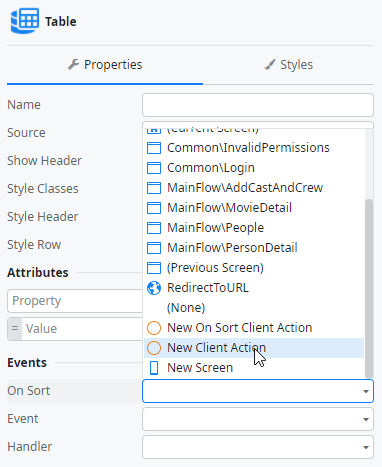
1. Na Tela Movies, defina o Sort Attribute de cada Header Cell, para o respectivo atributo Entity.
   1. Abra a Widget Tree na Tela Movies.
   2. Expanda o MainContent, depois a Table e finalmente a Header Row, até ver os Header Cells.



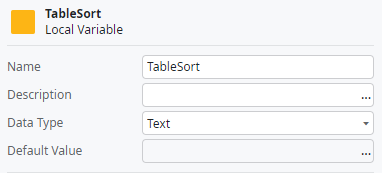
* 1. Para cada Header Cell, defina a propriedade **Sort Attribute** para o respectivo atributo Entity. Por exemplo, o primeiro Header Cell corresponde ao Title, então devemos definir o **Sort Attribute** para *Movie.Title*.



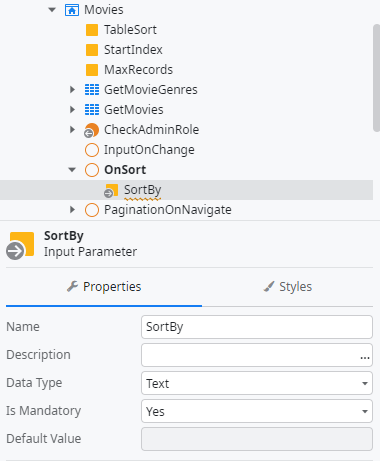
1. Defina o evento On Sort da Table para uma nova Client Action, onde a lógica da ordenação será implementada. Essa lógica exigirá que o critério de ordenação seja armazenado em uma variável local. Então, o start index deve ser redefinido e o Aggregate atualizado.
   1. Selecione a Table e no **On Sort Event** escolha *(New Client Action).* Isso criará uma Action *OnSort*.



* 1. Na Tela, crie uma variável local chamada *TableSort*, com o **Data Type** definida como *Text*.



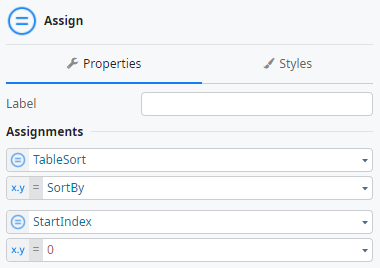
* 1. Na Action **OnSort,** adicione um novo Input Parameter chamado *SortBy* com o **Data Type** definido como *Text*.



* 1. Arraste um **Assign** e solte-o no fluxo da Action OnSort. Defina os seguintes assignments:

*TableSort = SortBy*

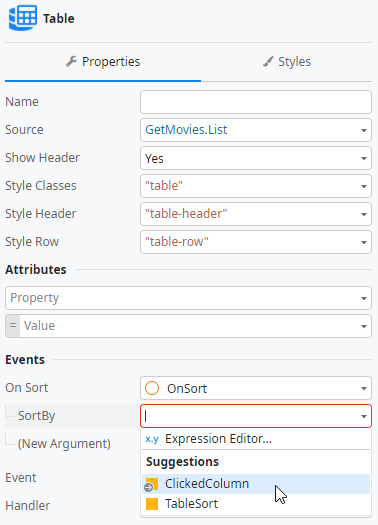
*StartIndex = 0*

**

**OBS:** O atributo SortBy manterá o valor da coluna a ser ordenada, selecionada pelo usuário. Então, definimos a variável TableSort para esse valor passado como parâmetro de entrada da Action OnSort. Voltaremos a isso mais tarde.

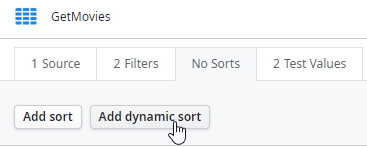
O StartIndex é definido como 0, para que depois de ordenar, voltemos à primeira página da paginação de Movies Table.

* 1. Arraste um **Refresh Data** e solte-o após Assign. Selecione o Aggregate **GetMovies** na nova janela.
  2. Para finalizar a lógica, volte para a Tela Movies, selecione a Table e note que há um erro no parâmetro de entrada da Action OnSort. Defina o valor do parâmetro de entrada para *ClickedColumn*.

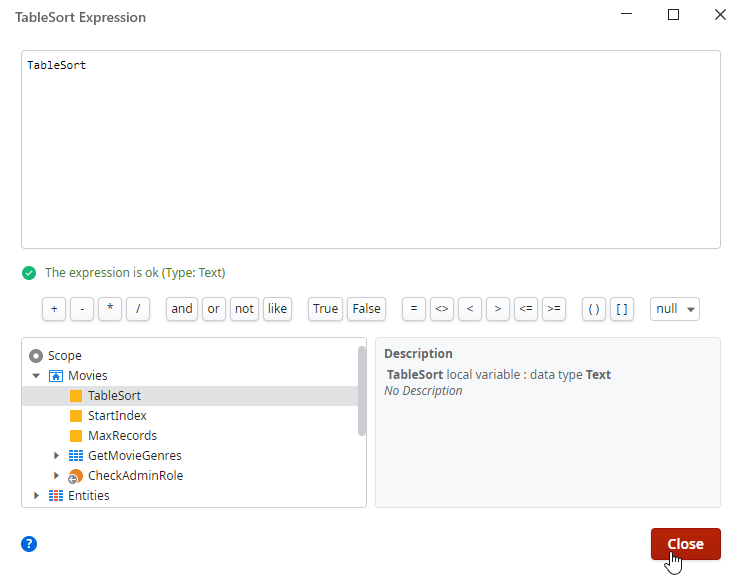


**OBS:** A entrada ClickedColumn aparece automaticamente quando estamos definindo o handler On Sort Event Action. Este valor da ClickedColumn irá manter automaticamente a seleção que o usuário final fizer durante o tempo de execução, dando significado ao Atributo Sort da coluna selecionada. Então, na Action salvamos essa informação na variável local TableSort, já que a usaremos mais tarde.

1. Neste ponto, toda a lógica foi definida, mas há um passo importante faltando. O Aggregate precisa ser ajustado para que obtenha os dados de acordo com o critério de ordenação selecionado pelo usuário. Para fazer isso, precisamos adicionar uma ordenação dinâmica no Aggregate.
   1. Abra o Aggregate **GetMovies** e selecione a guia **Sorting**.
   2. Clique na opção **Add Dynamic Sort**



* 1. Defina a Expression para *TableSort.* Clique em **Done** para confirmar.

****

**OBS:** A variável local TableSort possui os critérios de ordenação selecionados pelo usuário, então, definindo a classificação dinâmica para o valor da variável local, o Aggregate será ordenado pelo critério selecionado. A opção Add Sort não pode ser escolhida nesse caso, uma vez que apenas implementa a ordenação estática. A Add Dynamic Sort é a opção a escolher ao implementar uma ordenação dinâmica em uma Table / List.

1. Publique o módulo e teste a aplicação no navegador.