Paginação e Ordenação

Demonstração

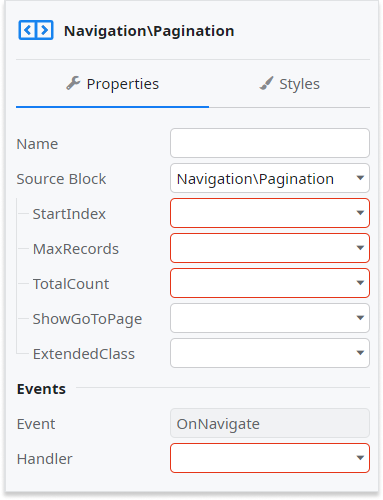
# Resumo

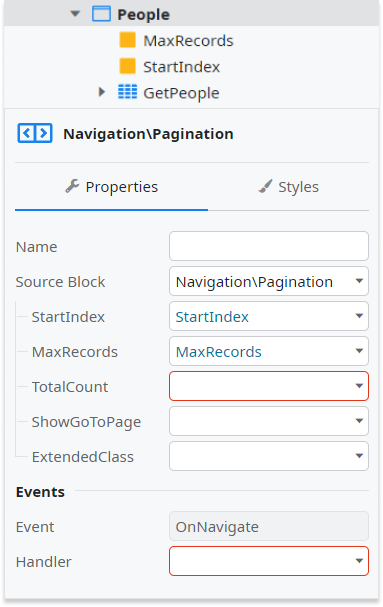
Este módulo não possui um conjunto de slides. O proposito desta demonstração é de mostrar paginação e ordenação dinâmica para Tabelas (tables) e Listas (lists). Neste exercício, os alunos irão aplicar paginação e ordenação dinâmica na tela Movies.

# Paginação

## Table

Para mostrar a paginação, essa demonstração deve cobrir os seguintes tópicos:

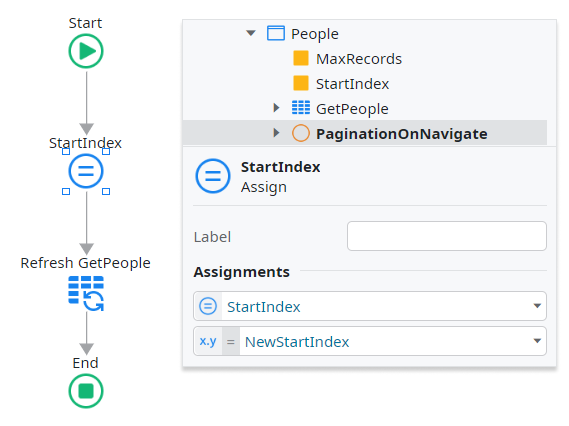
* Arraste e solte a Pagination (Paginação) abaixo do widget Table e explique suas propriedades
* Crie uma variável para a Start Index e Max Records



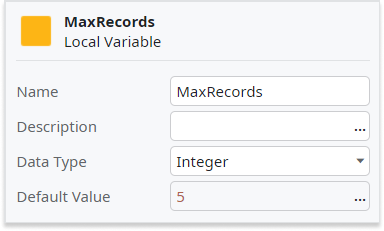
* Defina a TotalCount para ser o número total de registros buscados na consulta



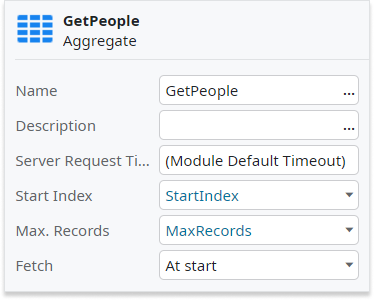
* Crie a Client Action que terá a lógica que suporta a mudança de página



* Mostrar o parâmetro de entrada na nova Action e explicar como ele se relaciona com a página que o usuário final seleciona.
* Defina alguns valores padrão para as variáveis para ajudar a explicar.



* Defina o Start Index e MaxRecords do Aggregate para as variáveis locais



* Mostre o funcionamento da paginação

## List

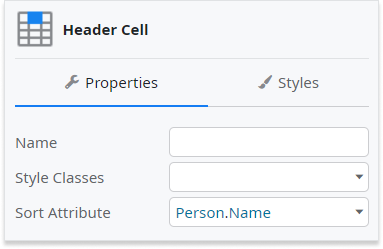
Em Lists, a paginação funciona de maneira semelhante. Sinta-se livre para mostrar isso ou apenas mencione para seus alunos.

# Ordenação

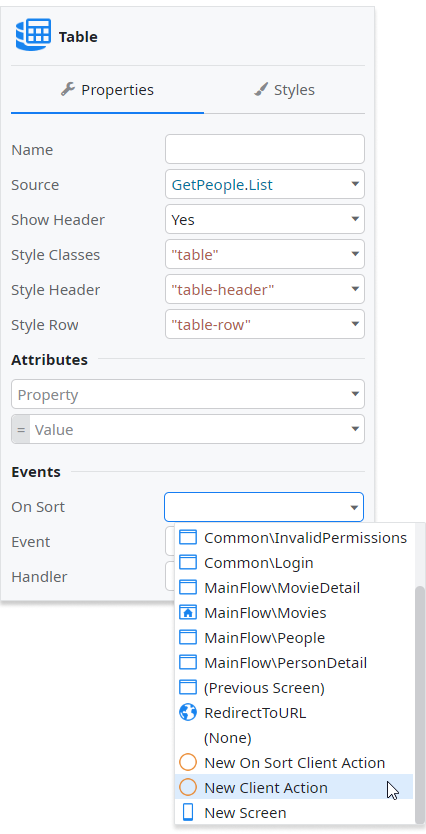
## Table

Para mostrara a ordenação, a demonstração deve cobrir os seguintes tópicos:

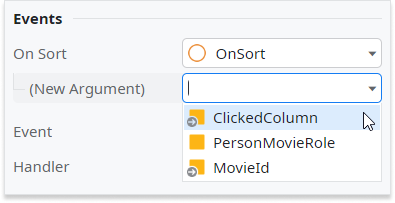
* Selecione a Name Header na Table e mostre o valor do atributo Sort. Explique como isso está relacionado ao atributo no banco de dados.



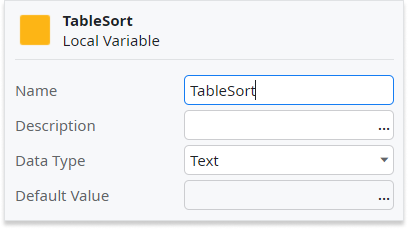
* Crie a Client Action OnSort para manter a lógica



* Adicione um novo parâmetro à chamada da Action para a Clicked Column



* Crie uma variável local para manter a Table Sort. Esta variável será muito útil para implementar a lógica da ordenação e usa-la no Aggregate Dynamic Sort.



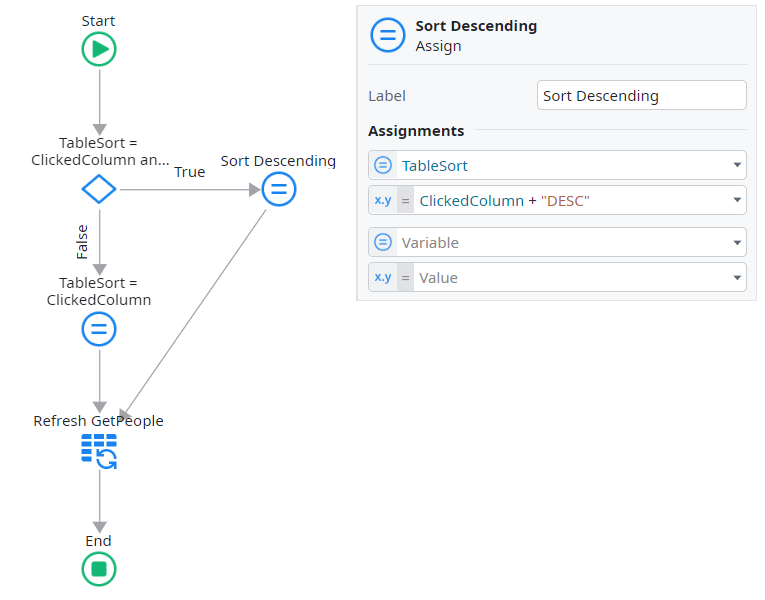
* Crie a lógica na Action OnSort.
  + Primeiro nós checamos se o usuário clicou na mesma coluna que antes (por isso salvo na variável TableSort)

*TableSort = ClickedColumn and ClickedColumn <> “”*

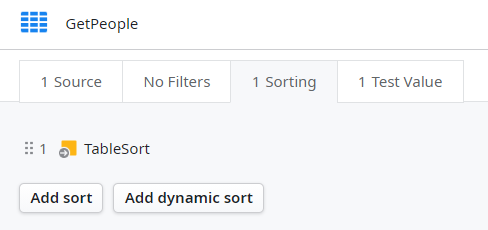
* + Se a condição for verdadeira, nós mudamos a ordenação para ser descendente.

*TableSort = ClickedColumn + “DESC”*

* + Se a condição é falsa, a TableSort recebe a nova ClickedColumn.
  + Ambos caminho devem ser seguidos pelo refresh do Aggregate.

**

* Adicione o Dynamic Sort no Aggregate

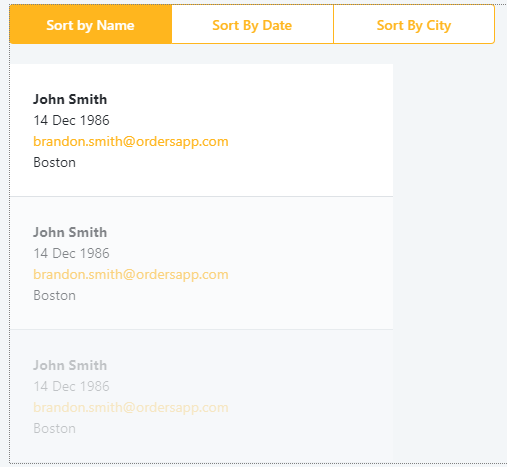


* Mostre a ordenação no aplicativo.

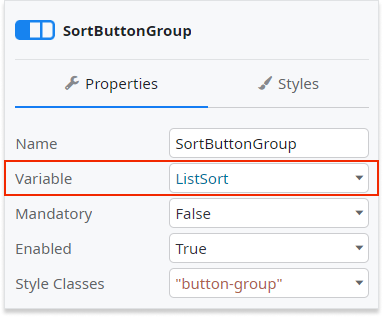
## List

The sorting in Lists works a little bit differently.

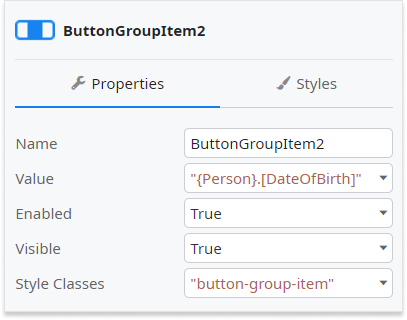
* Crie um cenário com a List e três itens Button Group.



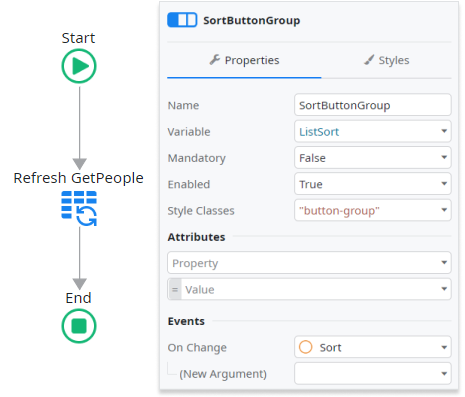
* Explique que o ButtonGroup tem uma variável local associada com isso, que conterá os critérios de classificação escolhidos pelo usuário.



* Explique que cada ButtonGroupItem tem seu critério de ordenação definido, para que quando o usuário clique em um item, a variável ListSort Variable vai manter esse valor.



* Explique a lógica necessária na propriedade OnChange do ButtonGroup, onde o Aggregate, que é a conte da List, é atualizado (Refresh).



* Adicione uma nova dynamic sorting noAggregate usando a variável local ListSort.

