

## **Pagination Controller – Detailed Explanation**

## Overview

This is the JavaScript controller part of an Aura component. It contains client-side functions that respond to user actions (like clicking buttons). These functions use a helper method called `loadContacts` to fetch and display Contacts in a paginated way.

## Functions Overview

Functions Overview: - `loadData`: Initialize table columns and load first page - `prevPage`: Go to the previous page of results - `nextPage`: Go to the next page of results - `gotoPage`: Jump to a specific page number

## Detailed Explanation

`loadData`: Runs when the component loads. It sets up the columns to be displayed in the datatable and then calls the `helper.loadContacts` function to fetch the first page of contacts.  
`prevPage`: Checks the current page number. If it is greater than 1, it reduces the page number by one and calls `helper.loadContacts` to load the previous page of data. `nextPage`: Gets the current page number and `totalPages`. If the current page is less than the total number of pages, it increases the page number and calls `helper.loadContacts` to load the next page. `gotoPage`: Used when the user clicks on a specific page number link. It gets the page number from the button label, updates `v.pageNumber` and then calls `helper.loadContacts` to load that particular page.

```

// Set up base column config
component.set("v.columns", [
  { label: "First Name", fieldName: "FirstName", type: "text" },
  { label: "Last Name",   fieldName: "LastName",  type: "text" },
  { label: "Email",       fieldName: "Email",     type: "email" },
  { label: "Phone",       fieldName: "Phone",     type: "phone" }
]);

helper.loadContacts(component);
},

prevPage : function(component, event, helper){
  var currentPage = component.get("v.pageNumber");
  if(currentPage > 1) {
    component.set("v.pageNumber", currentPage - 1);
    helper.loadContacts(component);
  }
},

nextPage : function(component, event, helper){
  var currentPage = component.get("v.pageNumber");
  var totalPages  = component.get("v.totalPages");
  if(currentPage < totalPages) {
    component.set("v.pageNumber", currentPage + 1);
    helper.loadContacts(component);
  }
},

gotoPage : function(component, event, helper){
  var target = parseInt(event.getSource().get("v.label"), 10);
  component.set("v.pageNumber", target);
  helper.loadContacts(component);
}
})

```