ReactJS & Chakra UI Dynamic

Data Table

Table of Contents

[**Introduction** 1](#_Toc133321607)

[**Usage** 1](#_Toc133321608)

[**Examples** 2](#_Toc133321609)

[**Table with custom column definition** 2](#_Toc133321610)

[**Paged table with custom column definition** 4](#_Toc133321611)

# **Introduction**

By utilizing React and TypeScript, the component delivers exceptional user experience when dealing with a limitless amount of data. It boasts an instinctive API that allows for real-time updates, accessibility, theming, and customizable templates, all while ensuring high-speed performance.

Dependencies:

* ReactJS - <https://react.dev/>
* Chakra UI - <https://chakra-ui.com/>

# **Usage**

The component is very simple to use, it just need to be imported using ES6 format and it can be used as a tag <DynamicDataTable />.

Component has all the properties of a Chakra UI table which enables the infinite possibilities of stilling. More than that, this component has its own properties, mandatory and optional.

Mandatory properties:

* **colDefs:** this represents an array of column definition that follows the structure of ColDef Interface
* **request:** the request represents one method with 2 parameters, tableProps and a callback method. Table props represent the value for skipCount and the pageSize and the callback method ensures that the response is the one required for the table.

Optional properties:

* **pagination:** this property is Boolean and represents the option to use or not the pagination for displaying data in table.
* **pageSize:** in case of pagination the page size is also mandatory so it can be defined using this property. In case of missing the default value is 10.
* **t:** this property represents a method used for localizing the table headers or the table keys is case this is required.

# **Examples**

## **Table with custom column definition**

The first example is a simple table with no pagination. For this chose to go with following column definition that shows that any type of react element can be displayed in table cell.

Graphical user interface, application

Description automatically generated

Text

Description automatically generated

Text

Description automatically generated

## **Paged table with custom column definition**

For this table the option is to use a paged table with the following column definition that shows that any type of react element can be displayed in table cell.

Graphical user interface, application

Description automatically generated

**Text

Description automatically generated**

**Text

Description automatically generated**

Text

Description automatically generated