DynamicTable Component Documentation

The DynamicTable component is a reusable and flexible table component built using the PrimeReact library. It allows users to render, filter, search, and manage dynamic datasets in a table format. It also supports row clicking and optional row creation functionality.

**Props**

Here’s a detailed breakdown of the props that DynamicTable accepts:

**1. columns: ColumnConfig<T>[]**

* **Type:** Array of ColumnConfig<T>
* **Required:** Yes
* **Description:** This array defines the structure of the table. Each object in this array specifies a column’s configuration. A ColumnConfig has the following properties:
  + field: string | keyof T: Specifies the field in the data array that maps to the column.
  + header: string: Defines the column header text.
  + filterType?: "dropdown" | "multiselect" | "tristatecheckbox": Specifies the filter control type (optional).
  + filterOptions?: any[]: Options for the dropdown or multiselect filters (optional).
  + filterFunction?: (data: T[], value: string | number | boolean | null) => T[]: Custom filter function (optional).
  + bodyTemplate?: (rowData: T) => JSX.Element: A custom template for rendering the content of each cell (optional).

**2. data: T[]**

* **Type:** Array<T>
* **Required:** Yes
* **Description:** The data array that will be displayed in the table. Each object in this array corresponds to a row.

**3. globalSearch: (data: T[], value: string) => T[]**

* **Type:** Function
* **Required:** Yes
* **Description:** This function is responsible for handling global search functionality. It takes the current dataset (data) and the search value, returning the filtered data array.

**4. loading: boolean**

* **Type:** Boolean
* **Required:** Yes
* **Description:** Controls whether the table is in a loading state. If true, a loader will be shown instead of the table.

**5. pageName: string**

* **Type:** String
* **Required:** Yes
* **Description:** The name of the page that will be displayed in the title of the card that wraps the table.

**6. onRowClick?: (e: DataTableRowClickEvent) => void**

* **Type:** Function (Optional)
* **Required:** No
* **Description:** A callback function that triggers when a row is clicked. Receives the row click event as a parameter.

**7. onCreate?: () => void | null**

* **Type:** Function (Optional)
* **Required:** No
* **Description:** A callback function that triggers when the "Create" button is clicked. If this prop is provided, the "Create" button will be rendered; otherwise, it will be omitted.

**Behavior and Features**

**1. Data Initialization**

* The component initializes the table with the provided data and sets both tableData and filteredData state variables with this data.

**2. Global Search**

* The component performs global searching based on the globalSearchValue. If the globalSearchValue is not empty, it filters filteredData using the provided globalSearch function.

**3. Filtering**

* Each column can have a custom filter element, controlled by the filterType prop.
* Supported filter types are:
  + **Dropdown**: Select a single value from a dropdown list.
  + **MultiSelect**: Select multiple values.
  + **TriStateCheckbox**: A three-state checkbox (true, false, null).

**4. Custom Filter Logic**

* The filterFunction for each column can be customized. If the filterFunction is provided, the column will use it to filter data; otherwise, it will perform a default filter based on string matching.

**5. Create Button**

* If onCreate is provided, a "Create" button will be shown above the table. Clicking the button triggers the onCreate function.

**6. Clear Filters Button**

* Clicking the "Clear Filters" button resets all filters and the global search value, restoring the original tableData.