**CRUDDataTable Component Documentation**

**Overview**

The **CRUDDataTable** component is a versatile tool designed to simplify the process of creating, reading, updating, and deleting data within a tabular interface in a React application. It streamlines the management of data operations by providing a modular structure and customizable options.

**Usage**

To utilize the **CRUDDataTable** component effectively, follow these steps:

1. **Installation**: Ensure that you have the required dependencies installed, including React, React Query, and PrimeReact.
2. **Import Dependencies**: Import the necessary components and hooks:

A screen shot of a computer

Description automatically generated

1. **Define Columns and Labels**: Define the columns and labels for your data:
2. A black screen with green text

   Description automatically generated**Create and Render the Component**:

A screen shot of a computer program

Description automatically generated

**Example:**

A practical example of using this component can be found at: src/pages/productCRUD.

A screen shot of a computer program

Description automatically generated

**Configuration Options**

The **CRUDDataTable** component offers various configuration options to customize its behavior according to your application's needs:

* **http**: The HTTP client instance to communicate with your API.
* **keyField**: The unique identifier for your data.
* **view**: Configuration for viewing data.
* **edit**: Configuration for editing data.
* **add**: Configuration for adding data.
* **delete**: Configuration for deleting data.
* **multiDelete**: Configuration for bulk deleting data.
* **permissions**: Object specifying user permissions.
* **export**: Boolean indicating whether export functionality is enabled.
* **labels**: Object containing labels for UI elements.

**Column Definitions**

The **columns**, **editColumns**, and **addColumns** arrays define the structure of data columns for display, editing, and adding respectively. Each column object should contain the following properties:

* **field**: Field name from the data object.
* **header**: Column header label.
* **body**: Function to customize the content rendering.
* **other properties**: Additional configuration based on the column type (text, checkbox, etc.).

**Labels**

The **labels** object defines various labels used in the component's UI, such as labels for buttons and dialogs.