

multi-table-comp

multi-table-comp is a Vue.js (≥ 2.5) web component that displays multiple sub tables of data from a single parent table. The contents of a sub table is expanded/contracted by displaying/hiding the associated rows in the parent table.

multi-table-comp can be installed via with the included `package.json` file for a local installation via the [npm install](#) command. **multi-table-comp** depends on the [vue.js](#) framework. A demo folder is provided that used [Parcel](#) together with its associated `package.json` file to bundle together **multi-table-comp** along with its [vue.js](#) dependency for a simple application. Further details are provided below for running the demo.

Props

A prop in Vue.js is a custom attribute for passing information from a parent component hosting **multi-table-comp** instance(s) to an **multi-table-comp** as a child component.

multi-table-comp has the following props for a parent to bind and send information to:

`title` -- a title to be placed above the parent table (string, default: null)

`headings` -- the headings that all of the sub tables have in common (array, default: null)

`column_widths` -- the column widths (in pixels) for each of the sub tables' columns (array of numbers, default: [120,120,...])

`tree_line` -- defines the type of display lines that connect a sub table's name label with its rows. Accepted values are 'black', 'white', and 'dotted'. (string, default: 'black')

`css_variables` -- defines the css variables for **multi-table-comp** (object, default: null)

`tables` -- defines the sub table data to be displayed (object, default: null). The following is an example of `tables` taken from the included demo:

```
tables: {
  House: {
    open: true,
    data:[
      ['Carrion Fly','worldwide','gray'],
      ['Office Fly','California,Bay Area','white'],
      ['Common House Fly','-','brown']
    ]
  },
  Horse: {
    open: true,
    data: [
      ['Horn Fly','Kansas','red'],
      ['Face Fly','-','green'],
      ['Stable Fly','-','black']
    ]
  }
}
```

The keys to the `tables` object act as the labels for each of the sub tables. Each table key is assigned an object with the `open` key (a boolean that expands the table when initially rendered if `true`) and the `data` key which is an array of arrays of the sub table's data.

Styling

The `css_variables` prop is a javascript object that contains any combination of css variable names as keys and associated values. The following list are the css variable names along with their default values for a quick styling of **multi-table-comp**:

```
{
  multitable_comp_font_family: 'Verdana, serif',

  multitable_comp_title_font_size: '1rem',
  multitable_comp_title_color: 'black',

  multitable_comp_thead_color: 'black',
  multitable_comp_thead_border_bottom: '1px solid black',
  multitable_comp_thead_background: 'transparent',

  multitable_comp_tbody_height: '20rem',

  multitable_comp_down_icon: '\21D3';

  multitable_comp_cell_font_size: '1rem'
  multitable_comp_cell_color: 'black'
}
```

Events

multi-table-comp has an event that notifies the parent component when a sub table cell is clicked. **multi-table-comp** emits the following single named event:

```
'multitable_comp_cell' -- returns the label of selected sub table along with the
row and column cell index
```

The parent component can listen to the above event and provide a callback for further processing. Events emitted from a child component back to the parent is explained at [Vue Custom Events](#).

Demonstration

One demonstration of **multi-table-comp** is provided and can be viewed by hosting the `index.html` file in the `demo/dist` folder. The demo (templated in the `App.vue` file) displays two sub tables labelled `House` and `Horse` for two type of American flies. The sub tables have columns for `Fly Name`, `Location`, and `Color`.

As a suggestion, install [http-server](#) locally/globally via [npm](#) then enter the command `http-server` in the **multi-table-comp** `dist` directory. From a browser enter the url: `localhost:8080/` to view the demo.

Here is some example code for using **multi-table-comp** taken from the `App.vue` file:

```

<multi-table-comp
  :title="title"
  :headings="headings"
  :tables="tables"
  :column_widths="column_widths"
  :tree_line="tree_line"
  :css_variables="css_variables"
  v-on:multitable_comp_cell="show_cell">
</multi-table-comp>

```

And here are the data references:

```

{
  title: 'American Flies',
  headings: ['Fly Name', 'Location', 'Color'],
  tables: {
    House: {
      open: true,
      data: [
        ['Carrion Fly', 'Worldwide', 'gray'],
        ['Office Fly', 'California, Bay Area', 'white'],
        ['Common House Fly', '-', 'brown']
      ]
    },
    Horse: {
      open: true,
      data: [
        ['Horn Fly', 'Kansas', 'red'],
        ['Face Fly', '-', 'green'],
        ['Stable Fly', '-', 'black']
      ]
    }
  },
  column_widths: [240, 240, 80],
  tree_line: 'dotted',
  css_variables: {multitable_comp_cell_font_size: '20px'}
}

```

Note that the demo listens to the `multitable_comp_cell` event (i.e. `v-on:multitable_comp_cell="show_cell"`) and displays the payload at `console.log`.