LakeTree

LakeTree Dynamic Grid plugin for Nintex Cloud Automation

About

The LakeTree Dynamic Grid plugin is a powerful component that allows users to create dynamic and customisable grids using JSON, HTML, and CSS.

By utilising this plugin for a configurable grid, users can effortlessly create powerful and feature-rich grids suited to their needs. Whether building data-driven applications, dashboards, or reporting tools, this plugin provides the flexibility and customization options necessary for delivering visually appealing and user-friendly grid interfaces.

EMP #	Full Name ⇕	Date Of Birth ⇕	Salary ⇕	Email ⇕	Employee ⇕ Satisfaction	Leave Left ⇕ (hours)	Status ⇕
1	Zephr Workman	1 Feb 1976	$110,000.50	Email	80%	12	Active
2	Nero Bowen	3 Jul 1956	$231,783.20	Email	90%	54	Active
3	Octavia Velez	2 May 1992	$150,000.00	Email	100%	65	Away
4	Holly Merrill	5 Mar 2001	$80,000.00	Email	60%	78	Away
5	Jamalia Hampton	1 Mar 1998	$135,000.50	Email	77%	23	Active

Showing 1 to 5 of 5 results

Previous **1** Next

Contents

Version	Date	Modified By	Comment
1.0	10th July 2023	Adam Castle	Initial release
1.1	8th September 2023	Adam Castle	The old element name was documented. This has been updated

Configuration

Control Properties

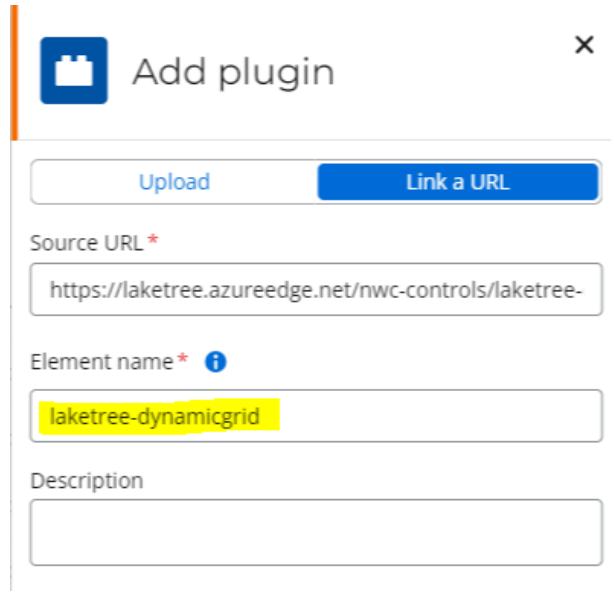| Name	Details

Column Configuration	Column configuration is based on a JSON object.
	
	``
	[
	  {
	    "id":"EmployeesID",
	    "name": "EMP #",
	    "sort":false,
	    "width": "100px"
	  }
	]
	``
	
	Please see the column configuration properties for more information.

Data	JSON data to be displayed.
Sortable	Sorting available on all columns by default
Pagination	Paging enabled
Custom CSS	Apply custom CSS to your grid. This lets you customise the look and feel
Enable onClick	This will allow you to extract the clicked item from the grid. It does cause the grid to refresh, so it is disabled by default.
Column Name	On row click, get the value of this column. If empty, the whole row will be returned.
Value	The extracted value from the grid when clicked on it. Either the data row or a specific column value.

Column Configuration Properties

Name	Details
Id	The name of the data column
name	This is the column heading. It can also be HTML.
format	This is format of the column. This can be HTML and uses [index] to get the column value.
hidden	true/false – You can hide a column if you want to use its value in another column. i.e. hide firstname and lastname, and create fullname. Please see the tutorial on how to do this.
sort	true/false – This column can be sorted
width	Width of the column. Needs to be as a string. i.e "200px"
type	This formats the value to a specified type. It uses the browsers locale for formatting. The accepted values are: • "date" • "number" • "decimal" • "percent" • "currency" Please note, the format property must refer to a single column i.e. [1] If it contains other characters the format will not work. Limitations : Currency will display as in USD.

align	"left", "center" or "right"

Adding the control

To add the control, drag the .js file into the file upload control, or store it in a CDN and link to it directly.

The element name for this plugin is: `laketree-dynamicgrid`

Tutorial

Tutorial for creating the below grid.

In this tutorial we will create a form that allows you to modify the runtime values of a grid.

You will be able to reuse this to test your configurations in the future.

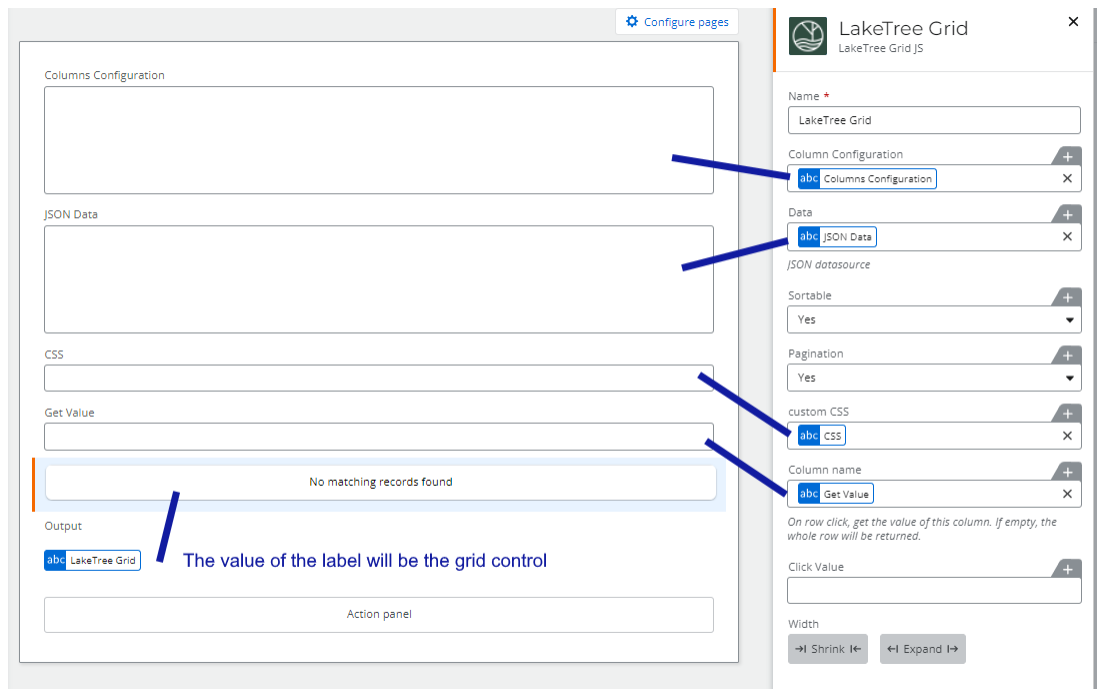| EMP #	Full Name	Date Of Birth	Salary	Email	Employee Satisfaction	Leave Left (hours)	Status
1	Zephr Workman	1 Feb 1976	$110,000.50	Email	80%	12	Active
2	Nero Bowen	3 Jul 1956	$231,783.20	Email	90%	54	Active
3	Octavia Velez	2 May 1992	$150,000.00	Email	100%	65	Away
4	Holly Merrill	5 Mar 2001	$80,000.00	Email	60%	78	Away
5	Jamalia Hampton	1 Mar 1998	$135,000.50	Email	77%	23	Active

Showing 1 to 5 of 5 results   Previous | 1 | Next

1. Create a new workflow, change the Start form to **Nintex Form** and click **Design Form**
2. Let create a form that where we can reconfigure our form at runtime.
   a. Drop the following control onto the form
      i. Text-Long
      ii. Text-Long
      iii. Text-Short
      iv. Text-Short
      v. LakeTree Grid JS
      vi. Label
   b. The structure of your form should resemble the below. Make sure to name your control labels the same as the highlighted text below.

   c. Configure the input values as per the grid below.

d. For this tutorial we will use the below JSON data. Paste this into the default value of the JSON data control.

[{"EmployeesID":1,"FirstName":"Zephr","LastName":"Workman","Email":"Lore m.ipsum.dolor@aliquetnec.ca","Status":"Active","Salary":110000.50,"Satisfact ion":0.80,"DOB":"1976-02-01","AvailableLeave":12.50},{"EmployeesID":2,"FirstName":"Nero","LastName ":"Bowen","Email":"In.at.pede@nequepellentesquemassa.com","Status":"Acti ve","Salary":231783.20,"Satisfaction":0.90,"DOB":"1956-07-03","AvailableLeave":54.23},{"EmployeesID":3,"FirstName":"Octavia","LastNa me":"Velez","Email":"amet.diam.eu@quam.net","Status":"Away","Salary":1500 00.00,"Satisfaction":1.00,"DOB":"1992-05-02","AvailableLeave":65.23},{"EmployeesID":4,"FirstName":"Holly","LastName ":"Merrill","Email":"purus@liberoest.org","Status":"Away","Salary":80000.00,"S atisfaction":0.60,"DOB":"2001-03-05","AvailableLeave":78.54},{"EmployeesID":5,"FirstName":"Jamalia","LastNa me":"Hampton","Email":"justo@Suspendissealiquet.com","Status":"Active","S alary":135000.50,"Satisfaction":0.77,"DOB":"1998-03-01","AvailableLeave":23.00}]

e. Click on **Preview**, and your form/grid should resemble the below.

In order to widen the form wider, to display additional columns;

f.   Click on **Styles** and open your CSS editor, Add the following:

```
.nx-form {
        max-width: 1500px;
        width: 1500px;
}
```

It should resemble the below.

Apply your changes and publish your workflow.

The column data may not be showing, this is fine as we'll customise this next.

IMPORTANT
Please note the column configuration take an array of objects. If the column configuration has errors, it will reset the grid back to it's default state.

g. Add the following JSON into the Grid Configuration.

```json
[
        {
            "id":"EmployeesID",
            "name": "EMP #",
            "sort":false,
            "width": "100px"
        }
]
```

This will;
- Display data from the Employee Id column
- Change the heading to EMP #
- Prevent sorting
- Set the width of the column

Your grid should look like the below.

EMP #
1
2
3
4
5

Showing **1** to **5** of **5** results   Previous **1** Next

Please notice only the <u>sort</u> property doesn't have double quotes around the value

h. Now let's extend the configuration to include but hide a few extra columns. Doing this will allow us to use this data in the configuration.

```json
[
    {
        "id":"EmployeesID",
        "name": "EMP #",
        "sort":false,
        "width": "100px"
    },
    {
        "name": "FirstName",
        "hidden":true
    },
    {
        "name": "LastName",
        "hidden":true
    }
]
```

Please note, the grid should look the same.

i.   We will now concatenate the first and last name together. Note, we refer to
     the column index within format property retrieve the data.
     At this point, this column indexes are;
     0 – EmployeeId
     1 – FirstName
     2 – LastName

     Copy the following into the column configuration text area.

```
[
    {
        "id":"EmployeesID",
        "name": "EMP #",
        "sort":false,
        "width": "100px"
    },
    {
        "name": "FirstName",
        "hidden":true
    },
    {
        "name": "LastName",
        "hidden":true
    },
    {
        "name": "Full Name",
        "format":"[1] [2]",
        "width": "200px"
    }
]
```

The grid should now look like the below. Note how we can concatenate the
columns together. **Please note:** This doesn't work when formatting columns.

EMP #	Full Name	
1	Zephr Workman	
2	Nero Bowen	
3	Octavia Velez	
4	Holly Merrill	
5	Jamalia Hampton	

Showing 1 to 5 of 5 results   Previous 1 Next

j.   We will now add a date field and format the data. Copy the following into the
     column configuration text area.

```
[
    {
        "id":"EmployeesID",
        "name": "EMP #",
        "sort":false,
        "width": "100px"
    },
    {
        "name": "FirstName",
        "hidden":true
    },
    {
        "name": "LastName",
        "hidden":true
    },
    {
        "name": "Full Name",
        "format":"[1] [2]",
        "width": "200px"
    },
    {
        "id": "DOB",
        "name":"Date Of Birth",
        "type":"date",
        "format":"[4]",
        "width": "130px"
    }
]
```

This will format the DOB value to a date.

As an experiment, remove **"type":"date",**

It will display the date as a string.

Your grid should now resemble the following:

EMP #	Full Name ⇕	Date Of Birth ⇕
1	Zephr Workman	1 Feb 1976
2	Nero Bowen	3 Jul 1956
3	Octavia Velez	2 May 1992
4	Holly Merrill	5 Mar 2001
5	Jamalia Hampton	1 Mar 1998

Showing **1** to **5** of **5** results   Previous **1** Next

k. We will now add Salary to the grid and format as currency.
    Copy the following into the column configuration text area.

```
[
    {
        "id":"EmployeesID",
        "name": "EMP #",
        "sort":false,
        "width": "100px"
    },
    {
        "name": "FirstName",
        "hidden":true
    },
    {
        "name": "LastName",
        "hidden":true
    },
    {
        "name": "Full Name",
        "format":"[1] [2]",
        "width": "200px"
    },
    {
        "id": "DOB",
        "name":"Date Of Birth",
        "type":"date",
        "format":"[4]",
        "width": "130px"
    },
    {
        "id":"Salary",
        "name":"Salary",
        "align":"right",
        "type":"currency",
        "format":"[5]",
        "width": "120px"
    }
]
```

In this case we are forcing the value to be formatted as currency and aligning the header and values to the right.

You grid show now look like the below:

EMP #	Full Name	Date Of Birth	Salary
1	Zephr Workman	1 Feb 1976	$110,000.50
2	Nero Bowen	3 Jul 1956	$231,783.20
3	Octavia Velez	2 May 1992	$150,000.00
4	Holly Merrill	5 Mar 2001	$80,000.00
5	Jamalia Hampton	1 Mar 1998	$135,000.50

Showing 1 to 5 of 5 results   Previous   1   Next

I.   Next, lets add a link into the grid, so you can click on the email field and open your email client, Copy the following into the column configuration text area:

```
[
    {
        "id":"EmployeesID",
```

```
                    "name": "EMP #",
                    "sort":false,
                    "width": "100px"
                },
                {
                    "name": "FirstName",
                    "hidden":true
                },
                {
                    "name": "LastName",
                    "hidden":true
                },
                {
                    "name": "Full Name",
                    "format":"[1] [2]",
                    "width": "200px"
                },
                {
                    "id": "DOB",
                    "name":"Date Of Birth",
                    "type":"date",
                    "format":"[4]",
                    "width": "130px"
                },
                {
                    "id":"Salary",
                    "name":"Salary",
                    "align":"right",
                    "type":"currency",
                    "format":"[5]",
                    "width": "120px"
                },
                {
                    "name":"Email",
                    "align":"center",
                    "id":"Email",
                    "format":"<a href='mailto:[6]'>Email</a>"
                }
            ]
```

Notice within the format property, we can create HTML elements.

You grid should look like the below.

EMP #	Full Name	Date Of Birth	Salary	Email
1	Zephr Workman	1 Feb 1976	$110,000.50	Email
2	Nero Bowen	3 Jul 1956	$231,783.20	Email
3	Octavia Velez	2 May 1992	$150,000.00	Email
4	Holly Merrill	5 Mar 2001	$80,000.00	Email
5	Jamalia Hampton	1 Mar 1998	$135,000.50	Email

Showing 1 to 5 of 5 results   Previous 1 Next

m. We will now amend the format of the Satisfaction data column from a decimal to a percentage. Please note that we included HTML into the column header. Copy the following into the column configuration text area.

```
[
    {
        "id":"EmployeesID",
```

```
                "name": "EMP #",
                "sort":false,
                "width": "100px"
            },
            {
                "name": "FirstName",
                "hidden":true
            },
            {
                "name": "LastName",
                "hidden":true
            },
            {
                "name": "Full Name",
                "format":"[1] [2]",
                "width": "200px"
            },
            {
                "id": "DOB",
                "name":"Date Of Birth",
                "type":"date",
                "format":"[4]",
                "width": "130px"
            },
            {
                "id":"Salary",
                "name":"Salary",
                "align":"right",
                "type":"currency",
                "format":"[5]",
                "width": "120px"
            },
            {
                "name":"Email",
                "align":"center",
                "id":"Email",
                "format":"<a href='mailto:[6]'>Email</a>"
            },
            {
                "id":"Satisfaction",
                "name":"Employee<BR>Satisfaction",
                "align":"right",
                "type":"percent",
                "format":"[7]",
                "width": "100px"
            }
        ]
```

Notice the <BR> in the name. This forces both words to be on different lines, your grid should look like the below.

EMP #	Full Name	Date Of Birth	Salary	Email	Employee Satisfaction
1	Zephr Workman	1 Feb 1976	$110,000.50	Email	80%
2	Nero Bowen	3 Jul 1956	$231,783.20	Email	90%
3	Octavia Velez	2 May 1992	$150,000.00	Email	100%
4	Holly Merrill	5 Mar 2001	$80,000.00	Email	60%
5	Jamalia Hampton	1 Mar 1998	$135,000.50	Email	77%

Showing 1 to 5 of 5 results   Previous 1 Next

n. We will add one more column to this grid and apply some styling. Copy the following into the column configuration text area.

```
[
    {
        "id":"EmployeesID",
        "name": "EMP #",
        "sort":false,
        "width": "100px"
```

```
        },
        {
            "name": "FirstName",
            "hidden":true
        },
        {
            "name": "LastName",
            "hidden":true
        },
        {
            "name": "Full Name",
            "format":"[1] [2]",
            "width": "200px"
        },
        {
            "id": "DOB",
            "name":"Date Of Birth",
            "type":"date",
            "format":"[4]",
            "width": "130px"
        },
        {
            "id":"Salary",
            "name":"Salary",
            "align":"right",
            "type":"currency",
            "format":"[5]",
            "width": "120px"
        },
        {
            "name":"Email",
            "align":"center",
            "id":"Email",
            "format":"<a href='mailto:[6]'>Email</a>"
        },
        {
            "id":"Satisfaction",
            "name":"Employee<BR>Satisfaction",
            "align":"right",
            "type":"percent",
            "format":"[7]",
            "width": "100px"
        },
        {
            "id":"Status",
            "name":"Status",
            "align":"right",
            "format":"<span class='[8]'>[8]</span> ",
            "width": "100px"
        }
    ]
```

Once again, we are creating an HTML element and showing the status.

Your grid should resemble the below.

EMP #	Full Name	Date Of Birth	Salary	Email	Employee Satisfaction	Status
1	Zephr Workman	1 Feb 1976	$110,000.50	Email	80%	Active
2	Nero Bowen	3 Jul 1956	$231,783.20	Email	90%	Active
3	Octavia Velez	2 May 1992	$150,000.00	Email	100%	Away
4	Holly Merrill	5 Mar 2001	$80,000.00	Email	60%	Away
5	Jamalia Hampton	1 Mar 1998	$135,000.50	Email	77%	Active

Showing **1** to **5** of **5** results   Previous **1** Next

Now we want to add some CSS for the Status format property to find.
You will notice with the below CSS there are two classes, one for each status.

.Active { color:green;border:1px solid green;border-radius:5px;padding:5px } .Away{ color:red }

Copy this code into the CSS textbox.
The grid and textbox should now look like below;

CSS

.Active { color:green;border:1px solid green;border-radius:5px;padding:5px } .Away{ color:red }

Get Value

EMP #	Full Name	Date Of Birth	Salary	Email	Employee Satisfaction	Status
1	Zephr Workman	1 Feb 1976	$110,000.50	Email	80%	Active
2	Nero Bowen	3 Jul 1956	$231,783.20	Email	90%	Active
3	Octavia Velez	2 May 1992	$150,000.00	Email	100%	Away
4	Holly Merrill	5 Mar 2001	$80,000.00	Email	60%	Away
5	Jamalia Hampton	1 Mar 1998	$135,000.50	Email	77%	Active

Showing **1** to **5** of **5** results   Previous **1** Next

You can see above we can apply CSS to our column values.

Onclick event

We can also extract the data when we click on from the grid. We can get the data for a row, or a specific column.

a. First you will need to ensure the **Enable OnClick** option is set to true in the grid's properties.

b. To get the row, click on the grid.

   The data for the row will be set into the property **Value**. We also setup the Output label to display the grids value, hence why we are now seeing JSON

Get Value

EMP #	Full Name ⇕	Date Of Birth ⇕	Salary ⇕	Email ⇕	Employee ⇕ Satisfaction	Status ⇕
1	Zephr Workman	1 Feb 1976	$110,000.50	Email	80%	Active
2	Nero Bowen	3 Jul 1956	$231,783.20	Email	90%	Active
3	Octavia Velez	2 May 1992	$150,000.00	Email	100%	Away
4	Holly Merrill	5 Mar 2001	$80,000.00	Email	60%	Away
5	Jamalia Hampton	1 Mar 1998	$135,000.50	Email	77%	Active

Showing **1** to **5** of **5** results   Previous **1** Next

Output

{"EmployeesID":1,"FirstName":"Zephr","LastName":"Workman","DOB":"1976-02-01","Salary":110000.5,"Email":"Lorem.ipsum.dolor@aliquetnec.ca","Satisfaction":0.8,"Status":"Active"}

c. To get the value for a specific column, enter the column name into the **Get Value** textbox.
   Below I have entered EmployeeId, and when I clicked on the 4th record I got the Id for that record.

Get Value

EmployeesID

EMP #	Full Name ⇕	Date Of Birth ⇕	Salary ⇕	Email ⇕	Employee ⇕ Satisfaction	Status ⇕
1	Zephr Workman	1 Feb 1976	$110,000.50	Email	80%	Active
2	Nero Bowen	3 Jul 1956	$231,783.20	Email	90%	Active
3	Octavia Velez	2 May 1992	$150,000.00	Email	100%	Away
4	Holly Merrill	5 Mar 2001	$80,000.00	Email	60%	Away
5	Jamalia Hampton	1 Mar 1998	$135,000.50	Email	77%	Active

Showing **1** to **5** of **5** results   Previous **1** Next

Output

4