Loader | WeWeb Documentation

|  |  |
| --- | --- |
| URL source | <https://docs.weweb.io/elements/loader.html> |
| Date de scraping | 2025-04-08T15:54:10.806292+00:00 |

## Loader ​

Loaders are a great way to improve a user's experience on your web-app.  
There are two main types of loaders:  
Loaders on page change, to let the user know they are indeed being redirected after clicking on a link Loaders during a process, to let the user know their action worked and the data is being fetched or a file is being uploaded for example

## Loaders on page change ​

When a user navigates from one page of your app to another, you can now show them a subtle loader to let them know their click triggered a page change:  
  
This option is available:  
when using the "Link" setting on an element as shown above, and when using the "Change page" action in a workflow.  
It will look something like this:  
  
If you want more control over when and how the loader appears, you can use the dedicated "Page loader" action in a workflow.  
In the example below, we decided to display the loader for 3 seconds before changing page:  
  
This can be useful when you have a workflow that you know will take time in the background.

### Images

<https://docs.weweb.io/assets/loader-8.mpo-Grc6.png>

<https://docs.weweb.io/assets/loader-9.IfsnWv40.gif>

<https://docs.weweb.io/assets/loader-10.FXehBeQh.png>

<https://docs.weweb.io/assets/loader-11.kDQ_ty48.png>

## Loaders on collection load ​

At times, your app will be waiting for data from your backend to display in the user interface.  
When that happens, it is a best practice to display a loader on the page.  
This lets the user know that nothing is broken, they just need to wait a second for the data.  
There are two steps to do this in WeWeb:  
1- add a Loader element where you want it on the page,  
2- define when you will show or hide the Loader, using the CSS display property or conditional rendering .

## Add a Loader element ​

You can find the Loader Element in the Add > Elements menu:  
  
Once you have dragged and dropped the element on the page, you can choose the type of Loader you would like and style it like you would any other element:

### Images

<https://docs.weweb.io/assets/loader-1.DMM5nE5L.png>

<https://docs.weweb.io/assets/loader-2.7CMvxXVo.gif>

## Bind the Loader to a condition ​

Once you have a Loader element on the page, you'll most likely want to display it when certain conditions are met, i.e. when the conditions are true or false .  
These conditions depend on your use case. Let's look at a couple of examples.

## Example 1 – Waiting for data from a Collection ​

This is a common use case for the Loader element: you are waiting for data from an external source.  
If the display depends on a Collection being fetched, you can bind the Loader element to the isFetching variable of that Collection:  
  
You will do something similar on the container with the elements you want to hide while the data is being fetched.  
Except in this case, you will bind it to the opposite of the isFetching variable by using the not no-code formula:  
  
What we are saying is that:  
if the Collection has been fetched already – so isFetching is false then we want to display the container with the data.

### Images

<https://docs.weweb.io/assets/loader-3.CUH44rSW.png>

<https://docs.weweb.io/assets/loader-4.D_b1yJW1.png>

## Example 2 – Loader for a Workflow in Progress ​

This can be very helpful when you have a workflow with a For Loop iterator that takes a bit of time for example.  
If you need to display a Loader while a workflow is in progress, you'll need to create a variable that is a Boolean, i.e. a variable that returns true or false :  
  
In the example above, we called our variable workflowInProgress and set it to false by default because, by default, workflows are inactive. They are triggered by an event.  
Then, you can add a display and hide action where you change the workflowInProgress variable at the beginning and end of your workflow.  
Since it's a boolean variable with a default value of false , you could use the not no-code formula as shown below:  
  
Once that's set up, you can bind the display properties of the Elements you want to show or hide to that Variable 🙂

### Images

<https://docs.weweb.io/assets/loader-5.nM8cQiS5.png>

<https://docs.weweb.io/assets/loader-6.CswR65-8.png>