**<https://www.udemy.com/course/quasarframework/>**

**Vue Basics:**

**The index.vue is divided into 3 parts:**

**1) Template (html tags)**

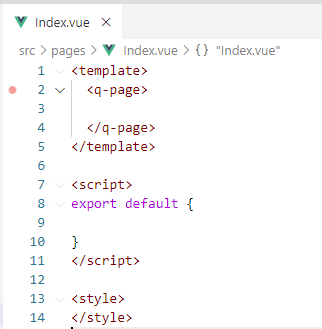
**2) Scripts**

**3) Styles**

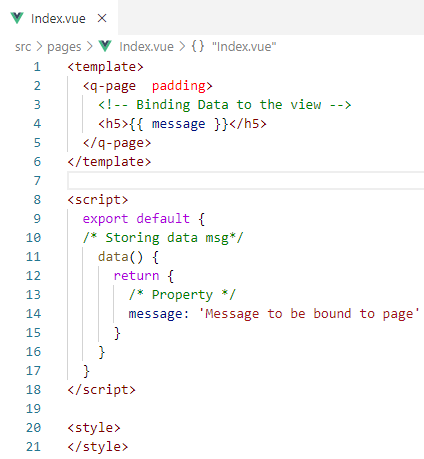


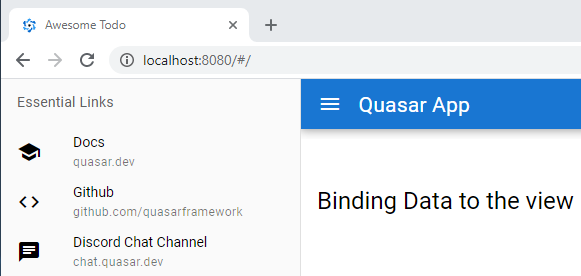
**Root element of all Quasar pages is the <q-page>**

**Cleanup:**



**Bind data to a View:**

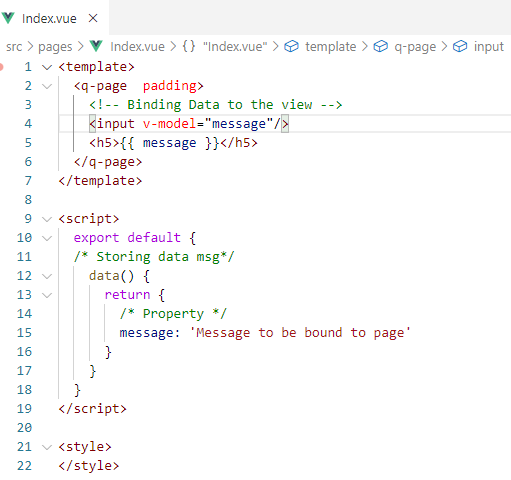


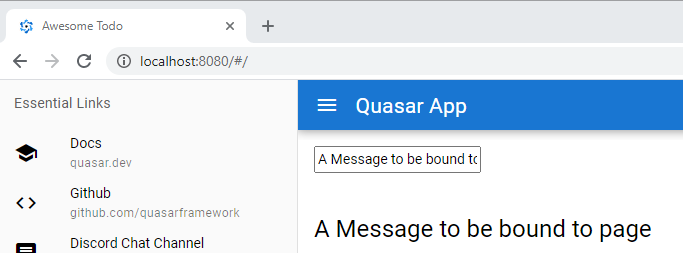


**Two-way binding with a view model:**

(Letting the user change the message)

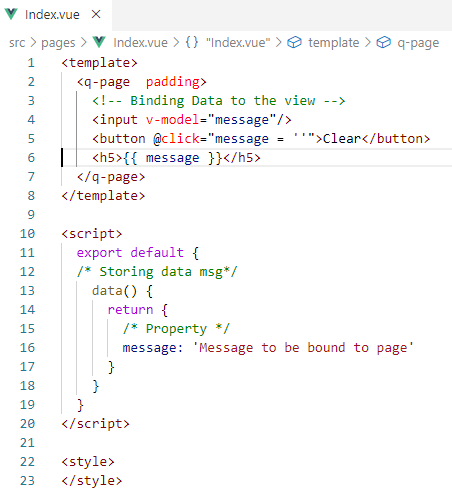
**Use v-model=”name of property”**



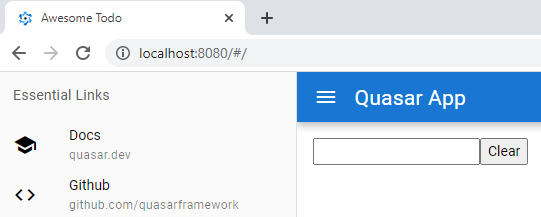


**Events and methods:**

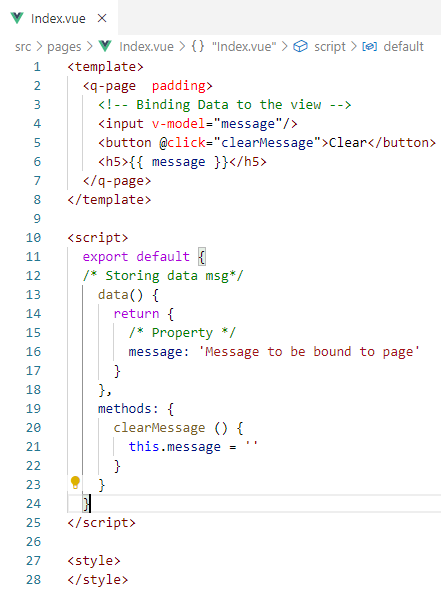
**( clicks, mouseovers, etc )**



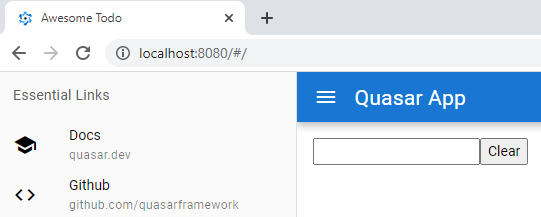
**Text is erased after clicking button:**



**If you want to fire more than one line of javascript, use a method name:**



**Now, the button uses a method name and that method clears the text:**

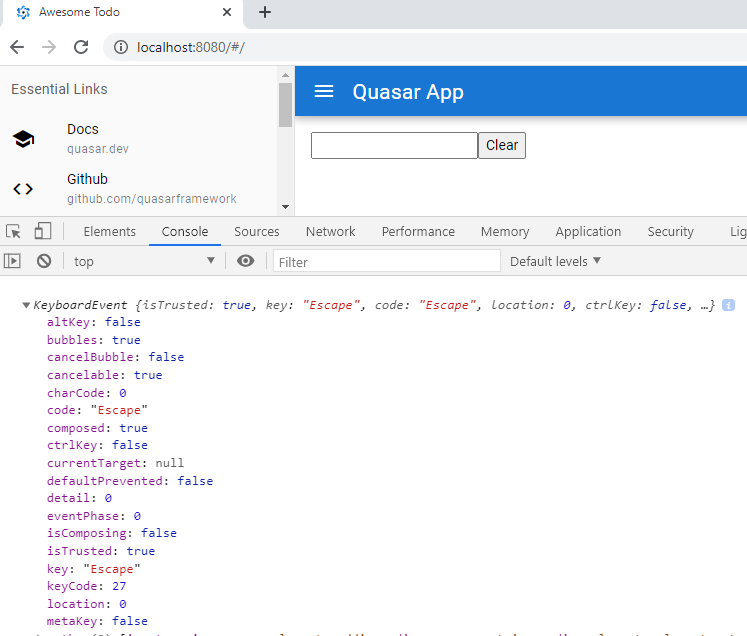


**What about key events:**

Use a keyup event and pass the event to the method handleKeyup(e)

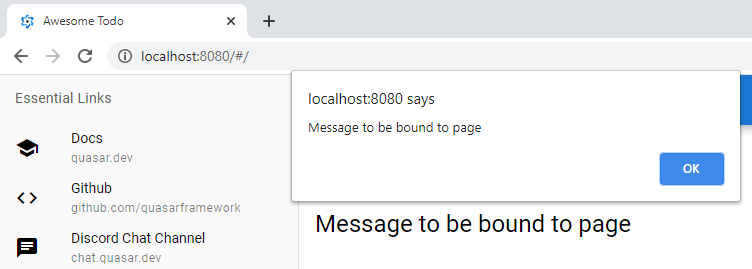


**Press Escape in the input and check Developer Tools for the keyCode:**



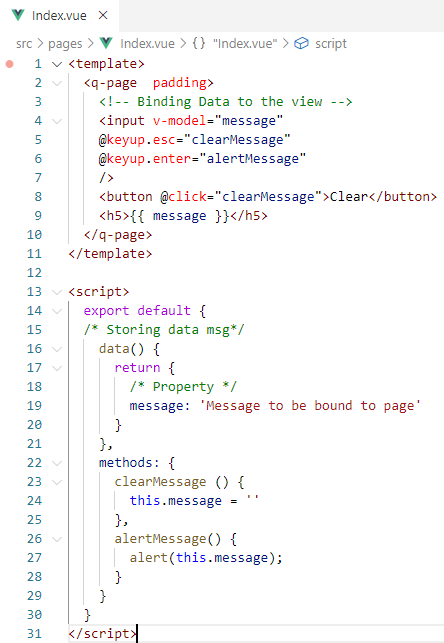


**Test the alertMessage with key Enter (keyCode == 13)**

**There is a shortcut in vue.js @keyup.esc removes the need for a keyCode.**

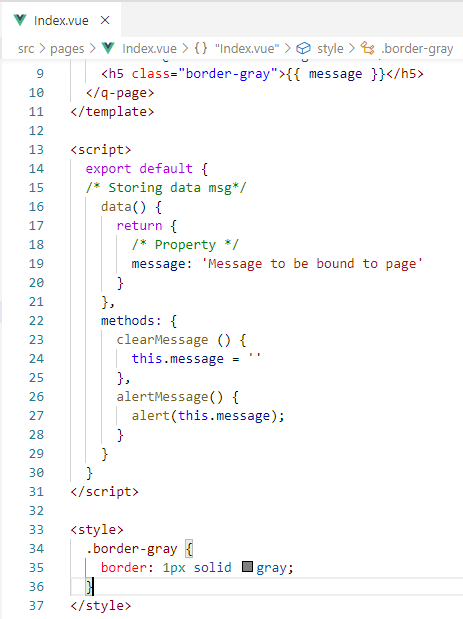


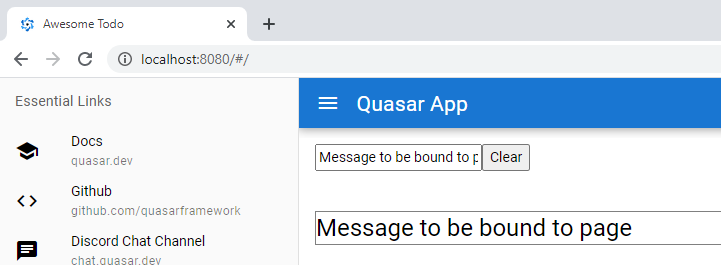
**We can remove handleKeyup method:**



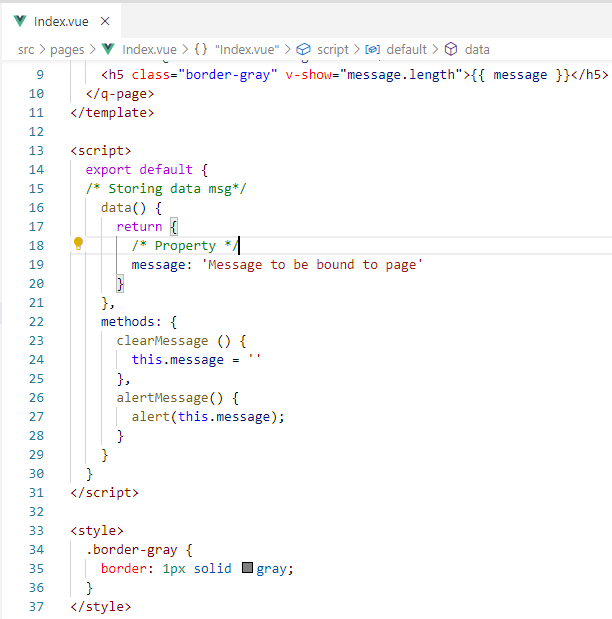
**Showing and Hiding Elements:**

**For demo purposes use a border around the message**

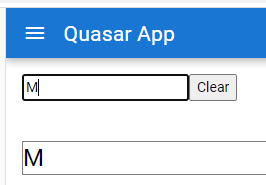


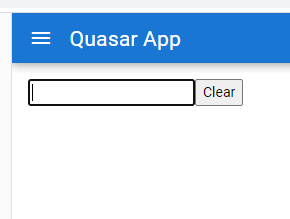


**Now, use v-show to hide and show the message based on length of the message being greater than zero:**

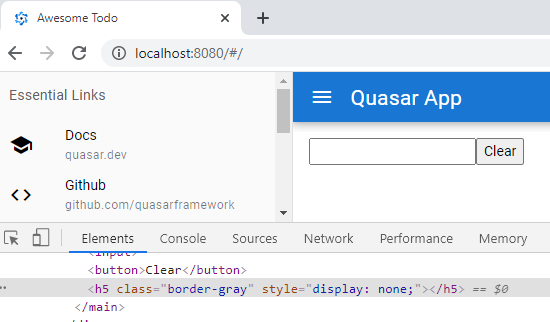


**At zero message length the border disappears as well!!**



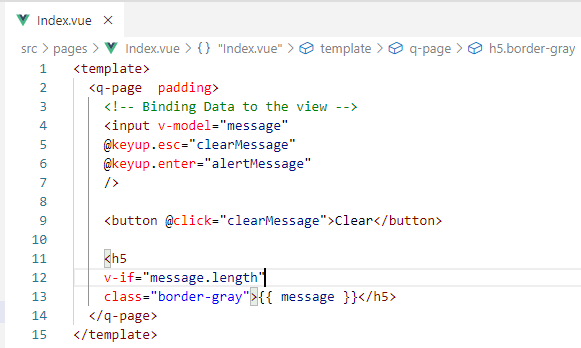


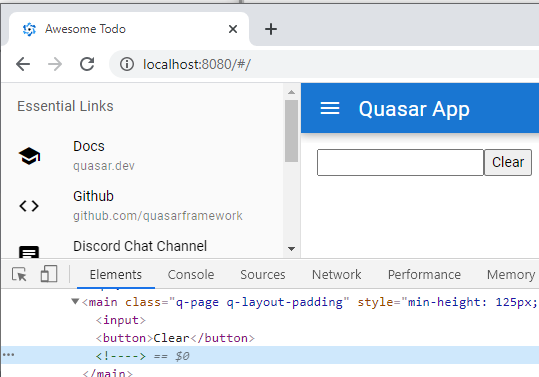
**display: none is used to hide the element**



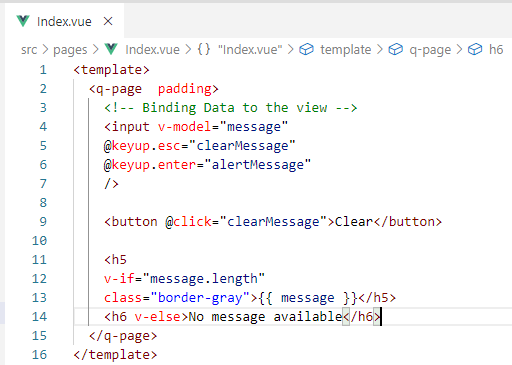
**If-else conditional rendering:**

This will remove the element from the DOM completely:

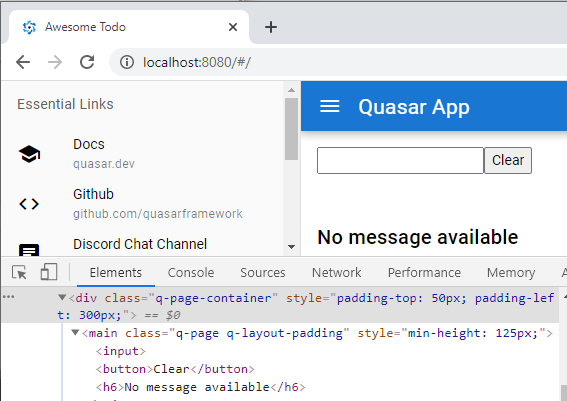




**Using v-else provides for the false condition alternative:**



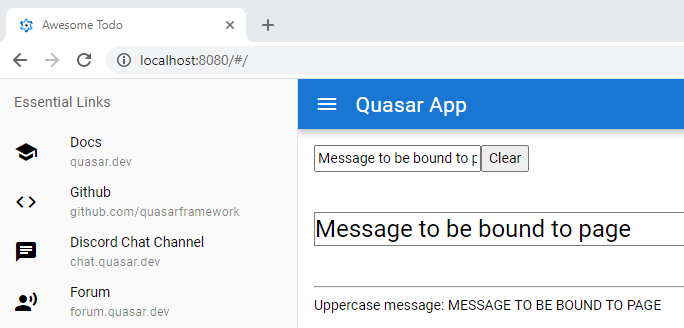
If v-if is false then, v-else is true!



**Computed Properties:**

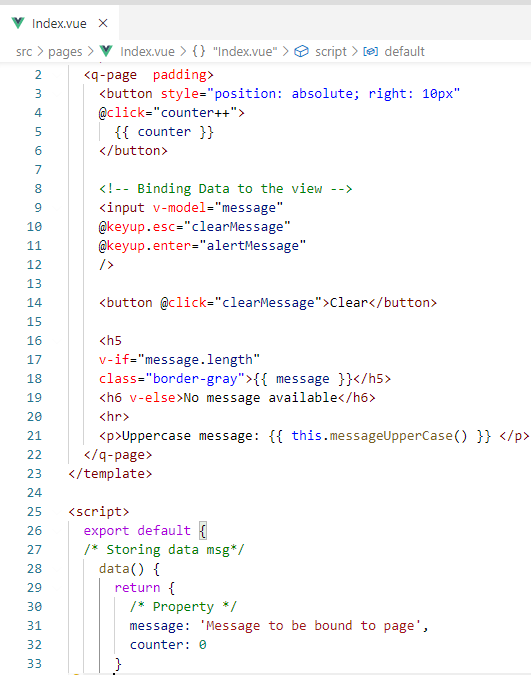
**How might we display a message somewhere else?**



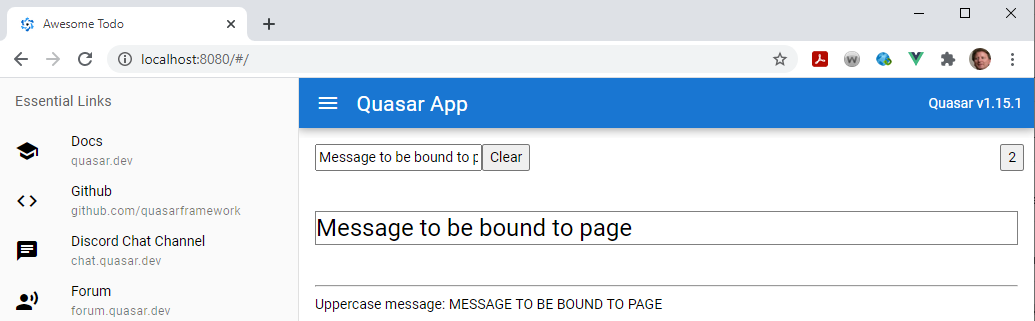


**Now, every time the page is refreshed the method will be called:**

Let us prove this with a counter method:



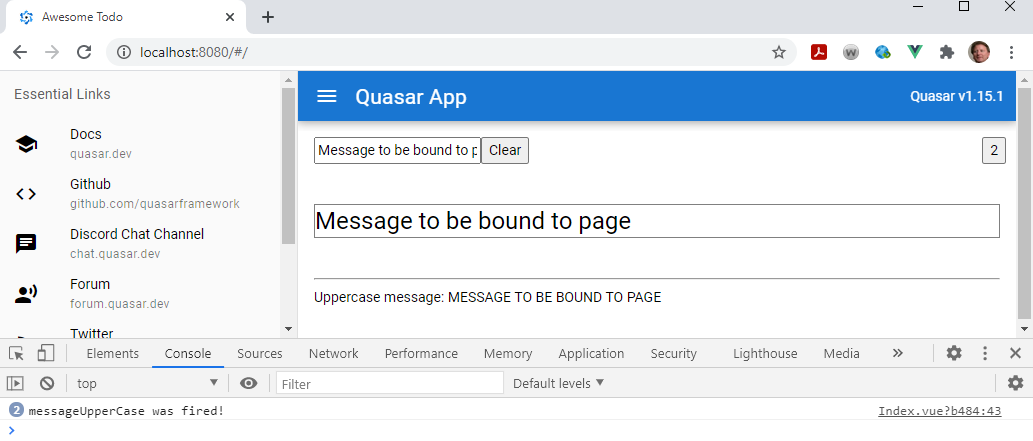
Clicked twice:



Incrementing the counter causes the methods to be fired!

Add something to show the firing:

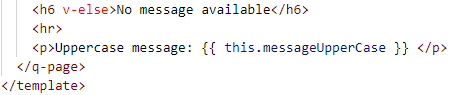




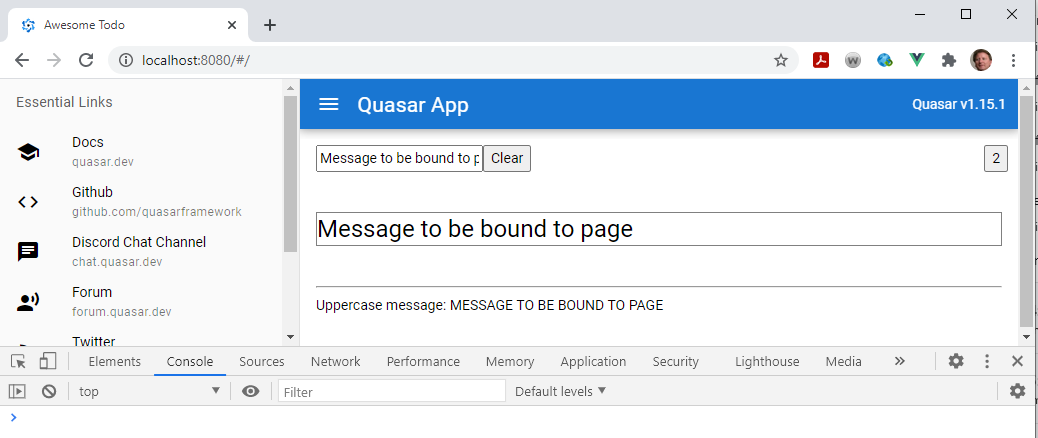
Move the method to the computed property section which will only fire on refresh:



We have to remember to remove the parentheses from the called method:

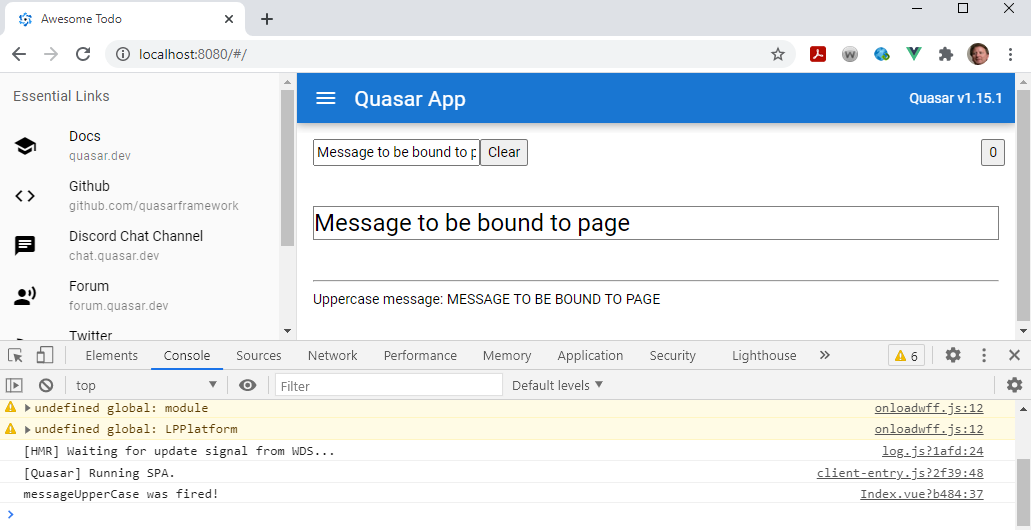


Test the counter button again.



The property never fires as designed!

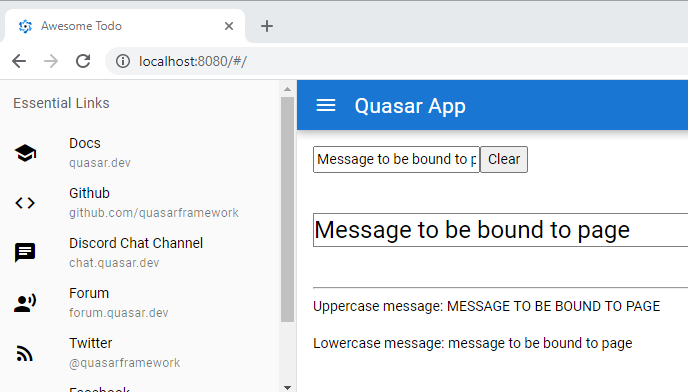
Only on refresh or on message change!!



**Manipulating data by using Filters:**

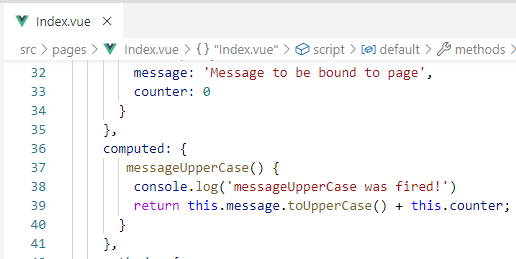
Filter is a method that accepts a value

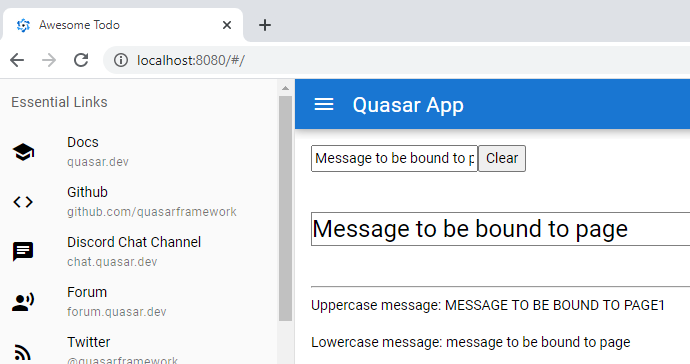




The Computed: property has access to all for the data(){} section

You can append the counter to the Computed: property



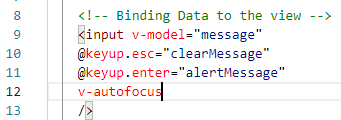


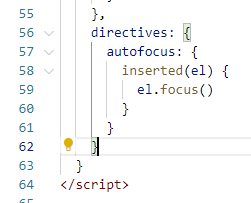
Filters are less powerful in that they have no access to data(){}, they can only accept values

Use filters for example dates

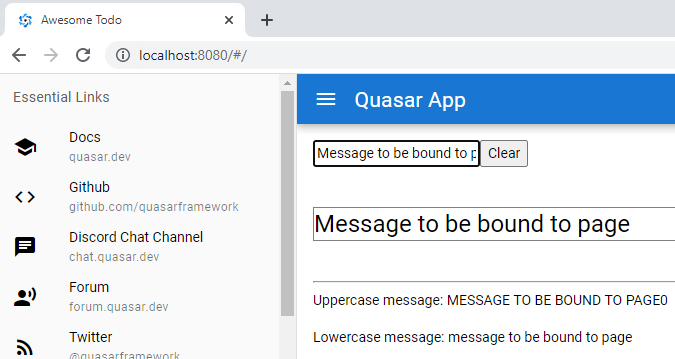
**Create a custom Directive in Vue.js**

Example will be an autofocus: <v-autofocus/>



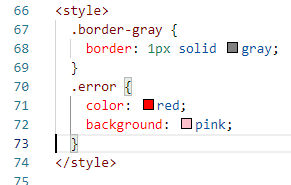


Notice the focus in on the input element:

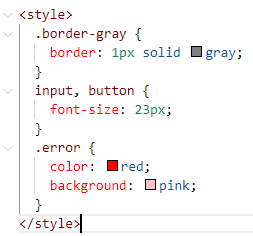


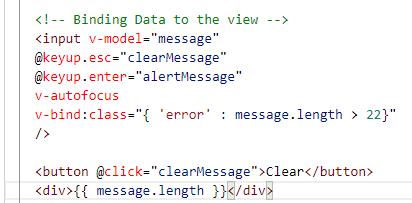
**Binding to attributes and CSS:**

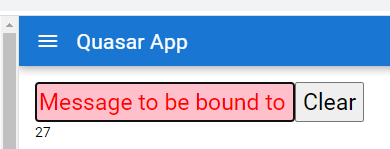
First create a style to bind with v-bind

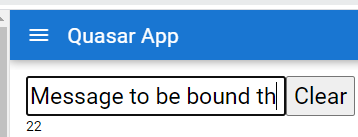


Increase the font size:

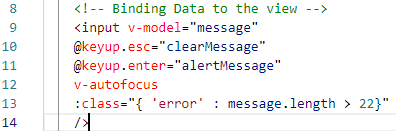




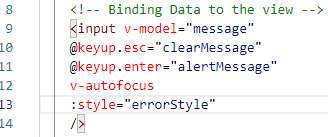




Now, there is a shortcut to v-bind:class => :class

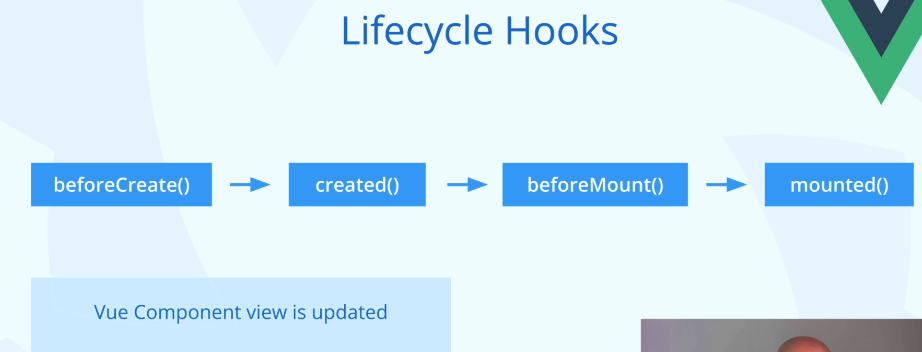


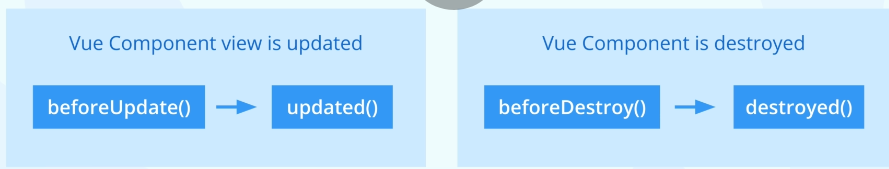
A different way is to use Computed: properties:

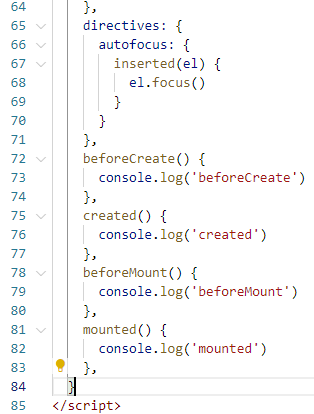


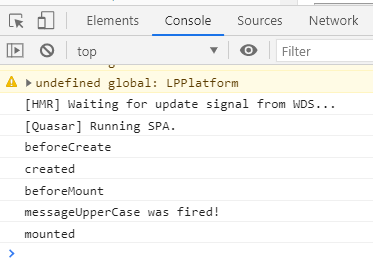


**Lifecycle hooks:**

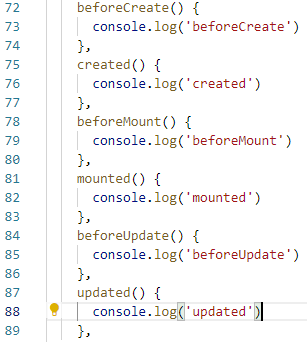




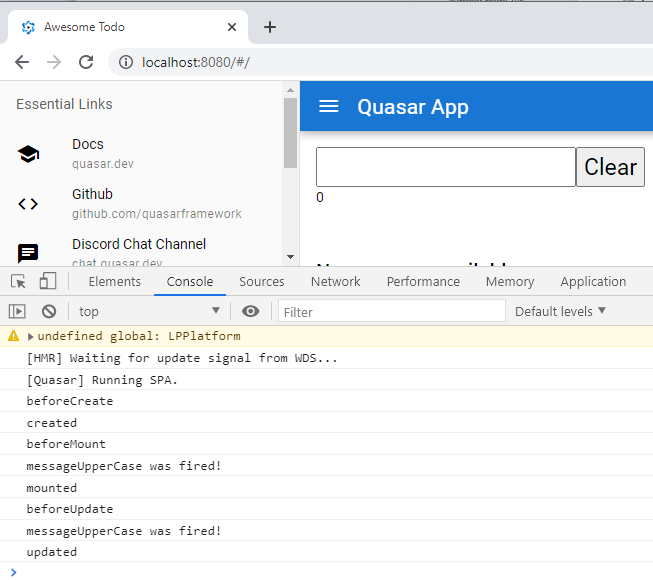




Now, add the hooks that fire on page changes

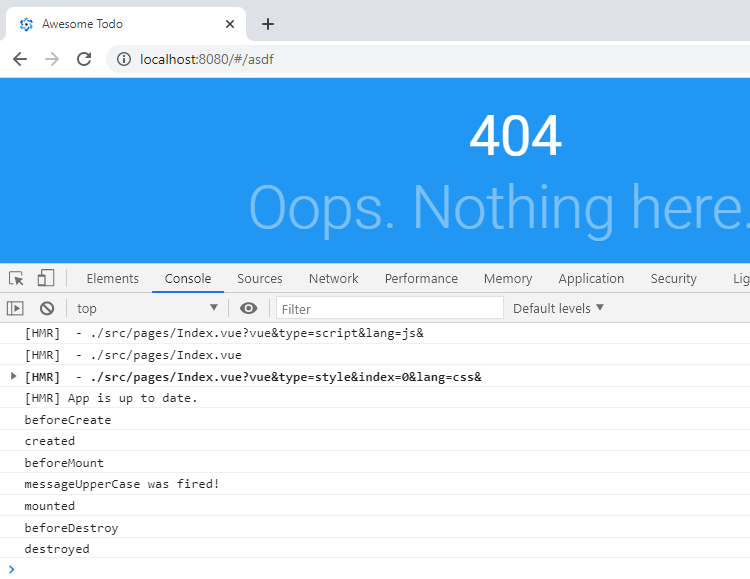


Run the page and then click a button:

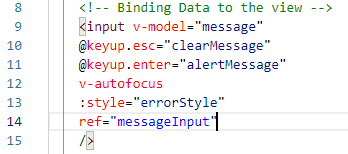




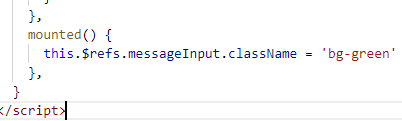
To demo navigate to the Error page with a bad url

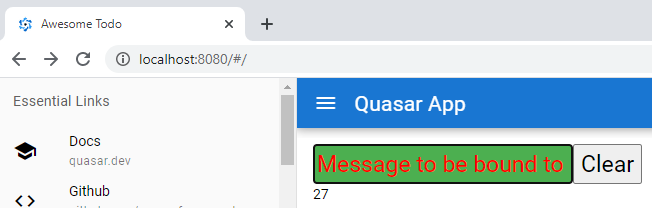


**References:**

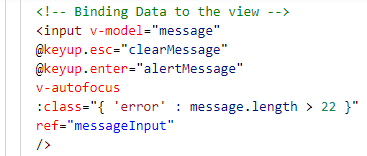


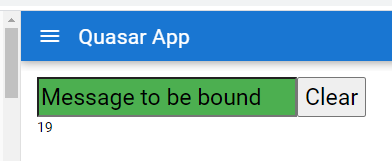
**The ref points to the input element:**



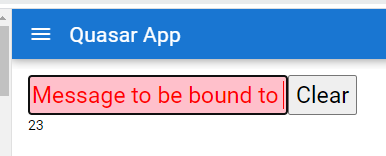


**Now, use :class instead of :style**





**Starts off green with a less than 22 message ‘mounted’, but is overwritten by red on > 22**



**Assignment:**

