

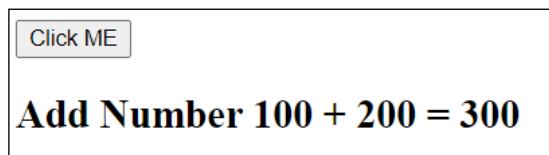
VUE C3-S2 – PRACTICE

EVENTS + FOR + IF + BINDING

EXERCISE 1 : *On click event*

In this exercise, we will make the sum of two values when clicking a button.

0. Open **exercise1.html**
1. Add a method to sum the variables **num1** and **num2**, and put the result in a new variable : **total**.
2. Add an event to execute the method when clicking on the button.

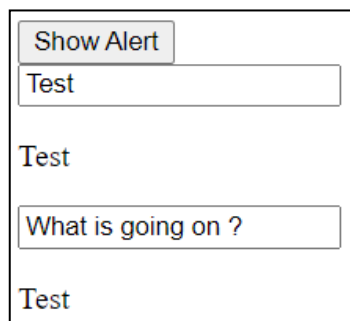


EXERCISE 2 : *keydown*

In this exercise, we will play with the key down event in inputs.

0. Open **exercise2.html**
1. Listen to the "keydown" event and store the value of the first input in a data property (hint: `event.target.value` gives you the value).
2. Display the data in the paragraph below.
3. Adjust the code from 1. in the second input, to only fire if the "key down" is the ENTER key.

Hint : use the [event modifiers](#)



EXERCISE 3 : *Mouse Over*

In this exercise, we want to change the color of a square when hovering it with the mouse.

0. Open **exercise3.html**

First, we need to make the style of the square easy to change.

1. Using the **styleobj** data, bind the style of the colored square.
2. Using the console, try to change the color of the square to green by changing the **styleobj** data.
3. Create a method to change the color of the square to green, and add an event to execute the method when the mouse is over the square.
4. Add an event & method to make the square back to red when the mouse is not over the square.



EXERCISE 4: Image Browser

In this part, you will make a website to browse images.

In each step, you should use the **VueJs directives** you know to achieve the objective.

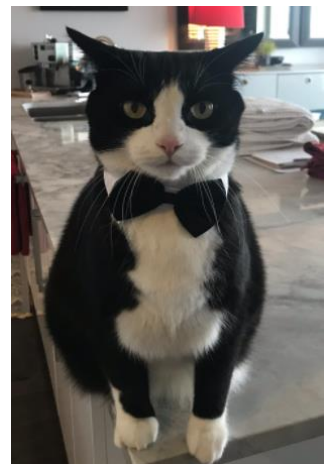
0. Open **exercise4.html**

The “**images**” data contains the links of images. The “**currentImage**” data will be the index of the current image to display in the “**images**” array.

1. Change the **** element to display the current image.
2. Define two methods, to go to the next image, and go to the previous image, using **currentImage**.
3. Bind the two methods to the according buttons **Previous** and **Next**.

Make sure currentImage does not go out of range.

Test your app !



EXERCISE 5: *Remake the Goal app !!*

In this exercise, you need to transform the application coded in vanilla JavaScript and use a Vue app instance

Goal

learn vue

sing a song