**Tutorial 5: Other Mouse Events**

**Vue** offers a variety of **mouse events** that can be useful when creating web applications. We will be looking at 4 different mouse events and viewing the event object.

The first being a mouse over event, when the mouse enters the element.

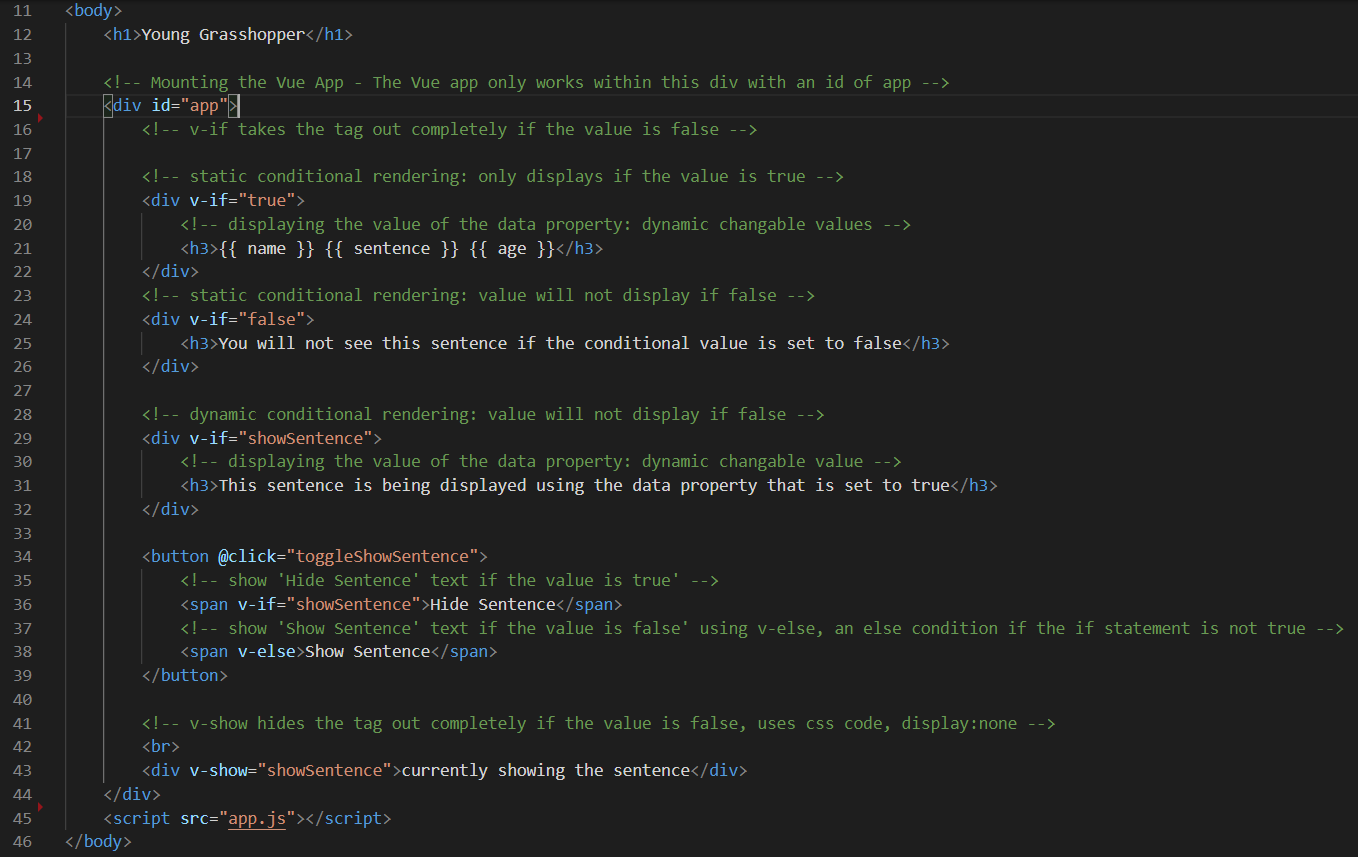
The second being a mouse leave event, when the mouse exits the element.

The third being a double click, when the element is clicked twice.

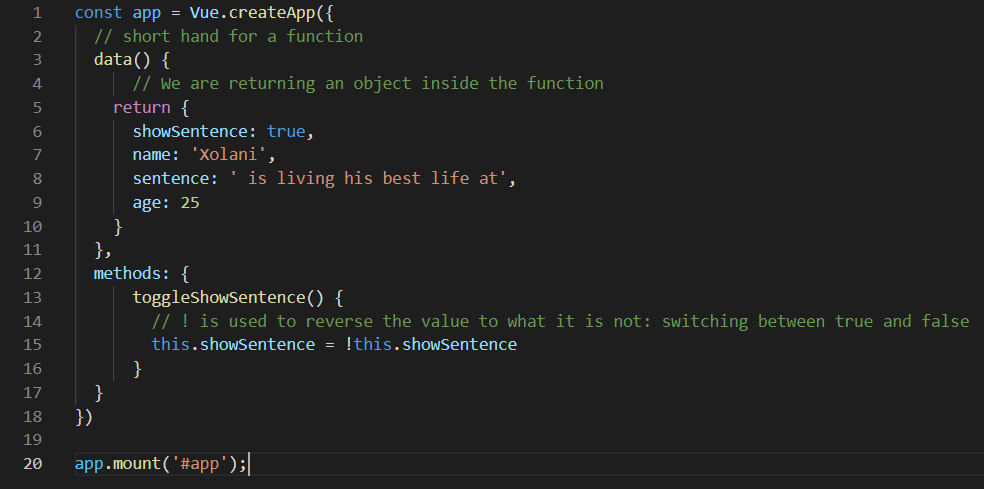
The fourth will determine the co-ordinate within the element, when mouse moves over the element.

Make sure **Tutorial 4** is complete to be able to move on to this tutorial or clone the [vue3-firebase-course](https://github.com/younggrasshopperza/vue3-firebase-course) repository and go to branch called **Tutorial-4**.

In the **index.html** file, make sure the **body** tag has the below code:



In the **app.js** file, make sure the code looks like the below:



**Step 1:** Continuing from Tutorial 4, in the **index.html** file, create a stylesheet in the **head tag** with the following CSS code:

.box {

        padding: 100px 0;

        width: 400px;

        text-align: center;

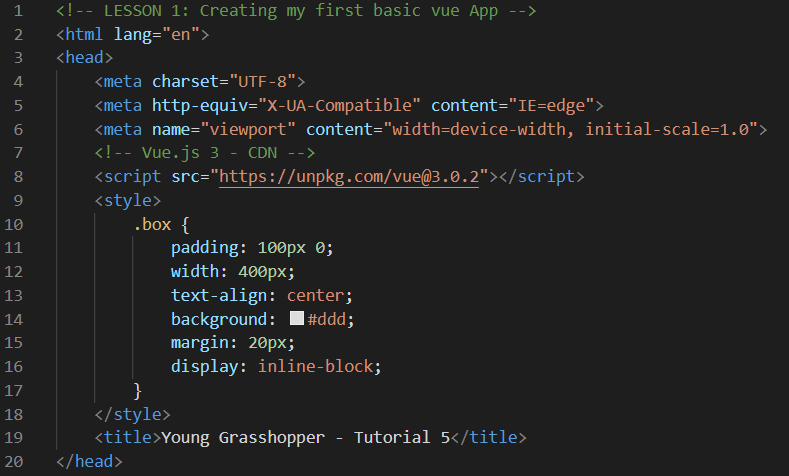
        background: #ddd;

        margin: 20px;

        display: inline-block;

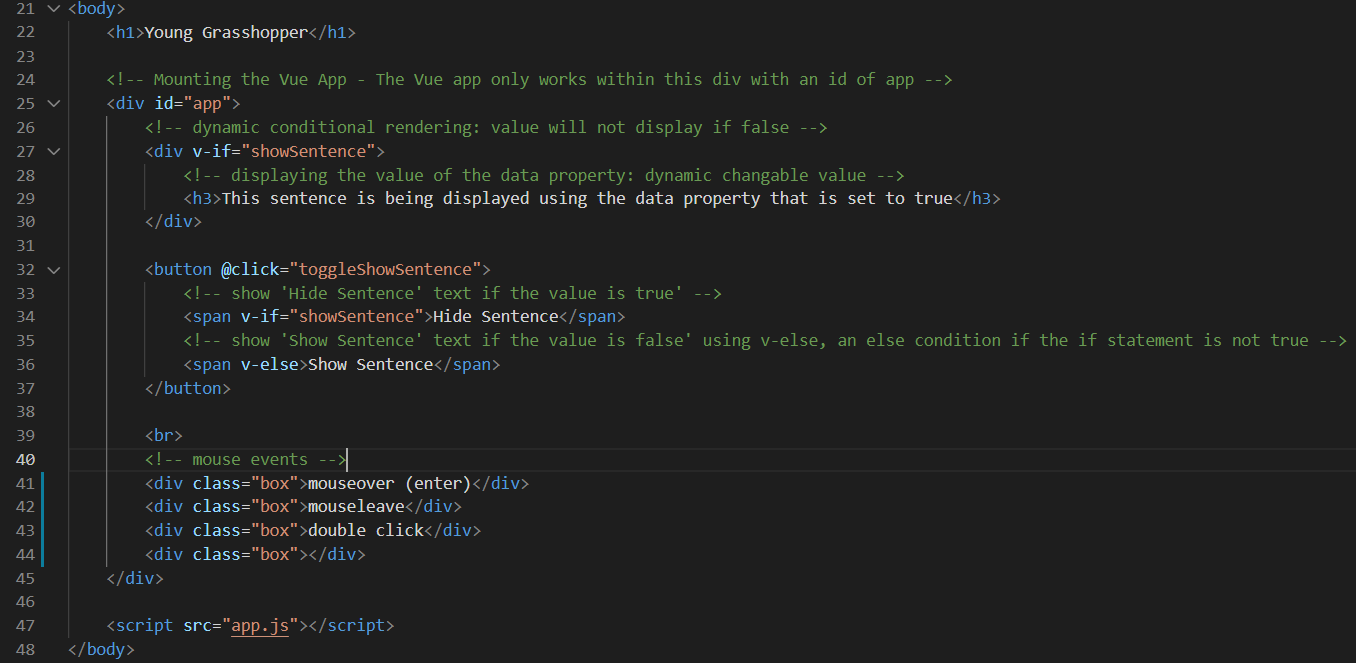
     }

Look at the example below (**Line 10 to 17**):



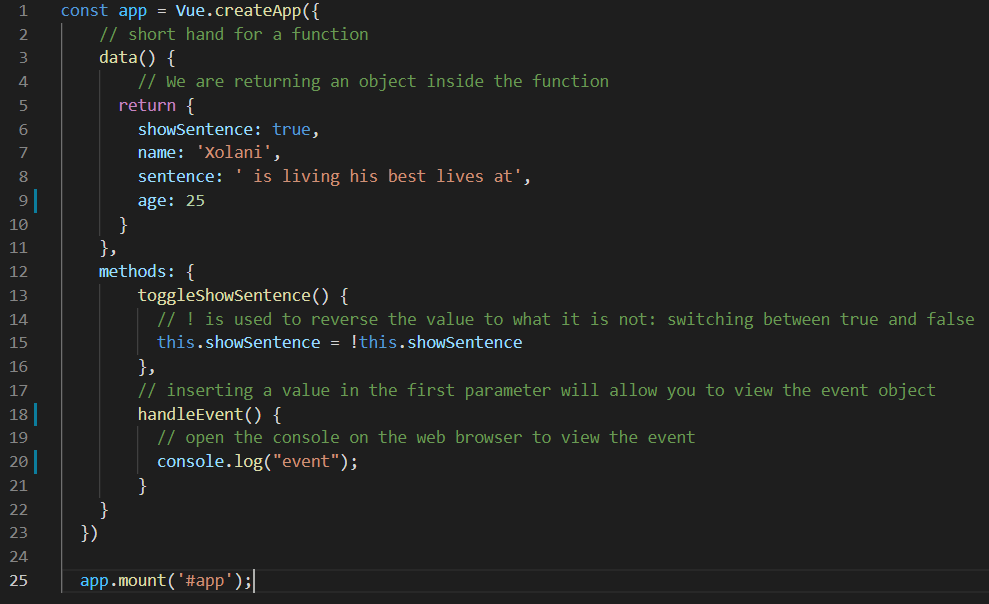
In the **index.html** file, create 4 different **div** tags and with the **class property** defined in the **style tag**.

Look at the example below (**Line 41 to 44**):



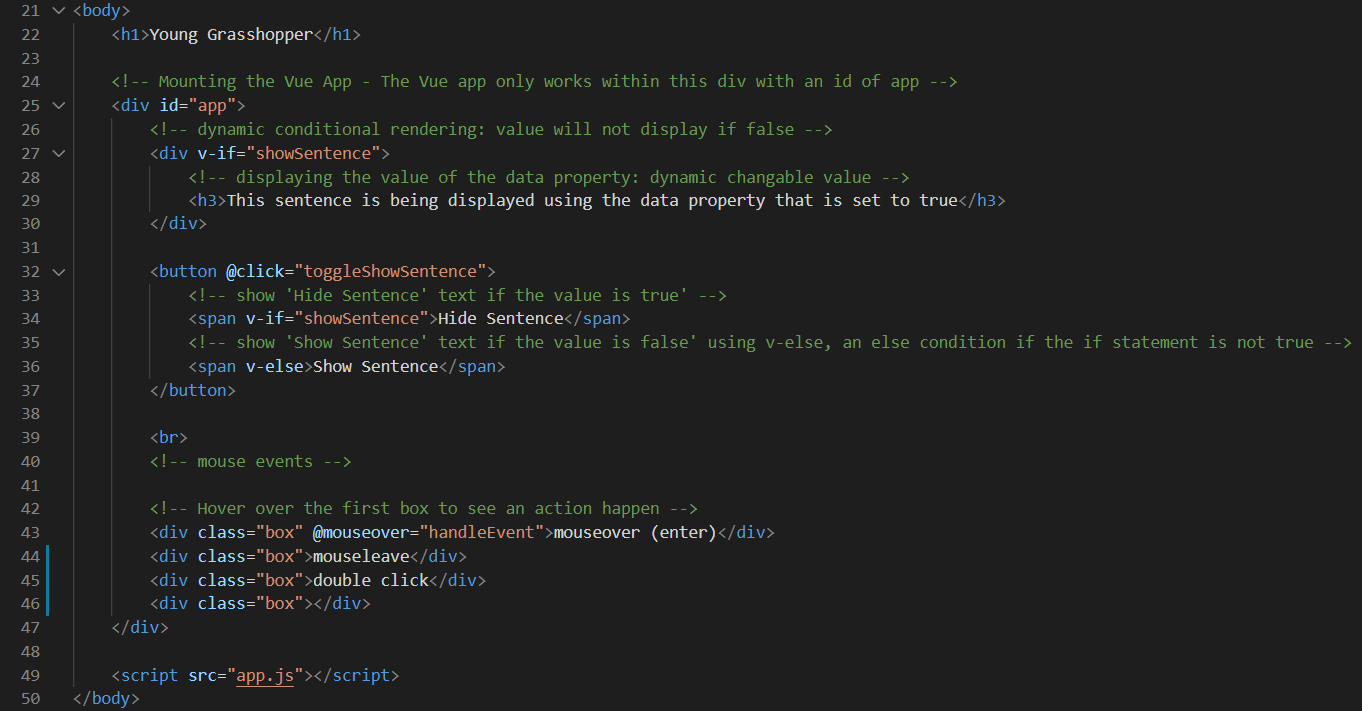
**Step 2:** In the **app.js** file, create a **function** called **handleEvent** that **console logs** the word event.

Look at the example below (**Line 18 to 21**):



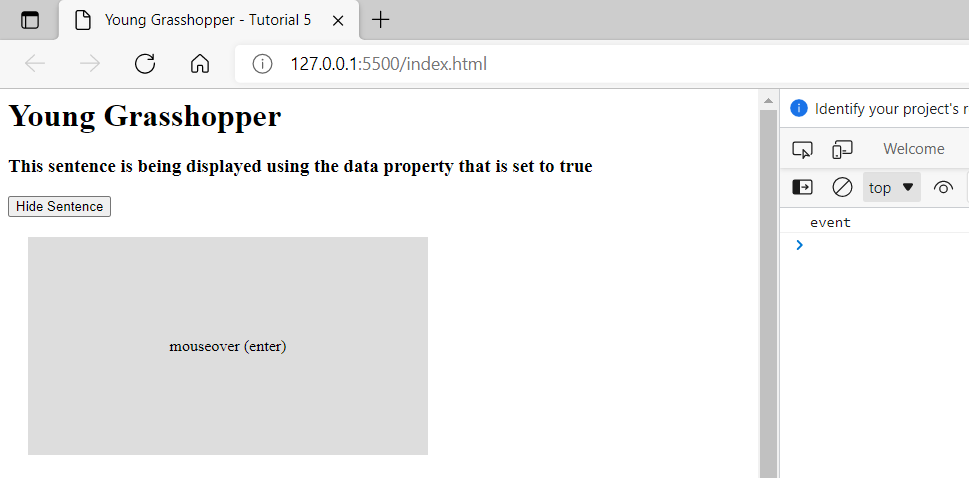
In the **index.html** file, create a **mouseover** property that references the function **handleEvent** in the first div.

Look at the example below (**Line 43**)



**Hover over** the first element to display the output in the **console log**.

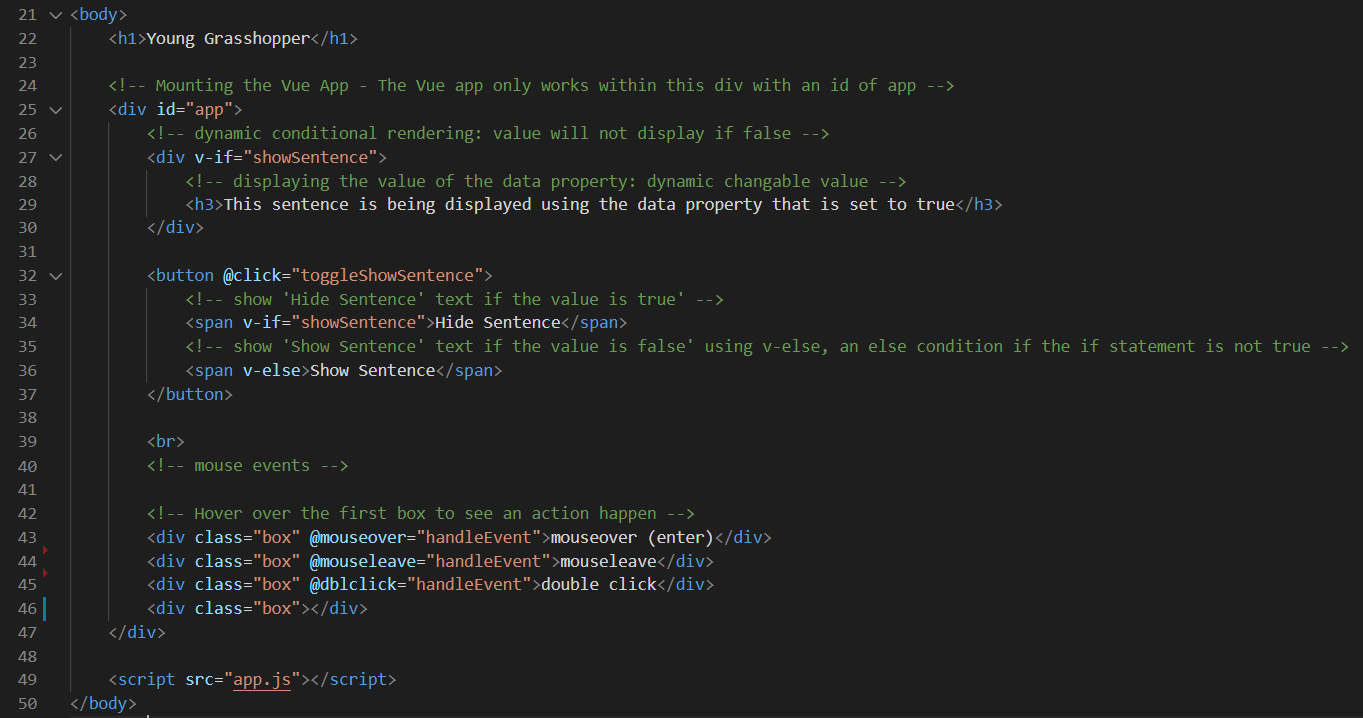
**Output:**



**Step 3:** In the **index.html** file, create a **mouseleave** property that references the function **handleEvent** in the second div.

Create a **ddlclick** property that references the function **handleEvent** in the third div.

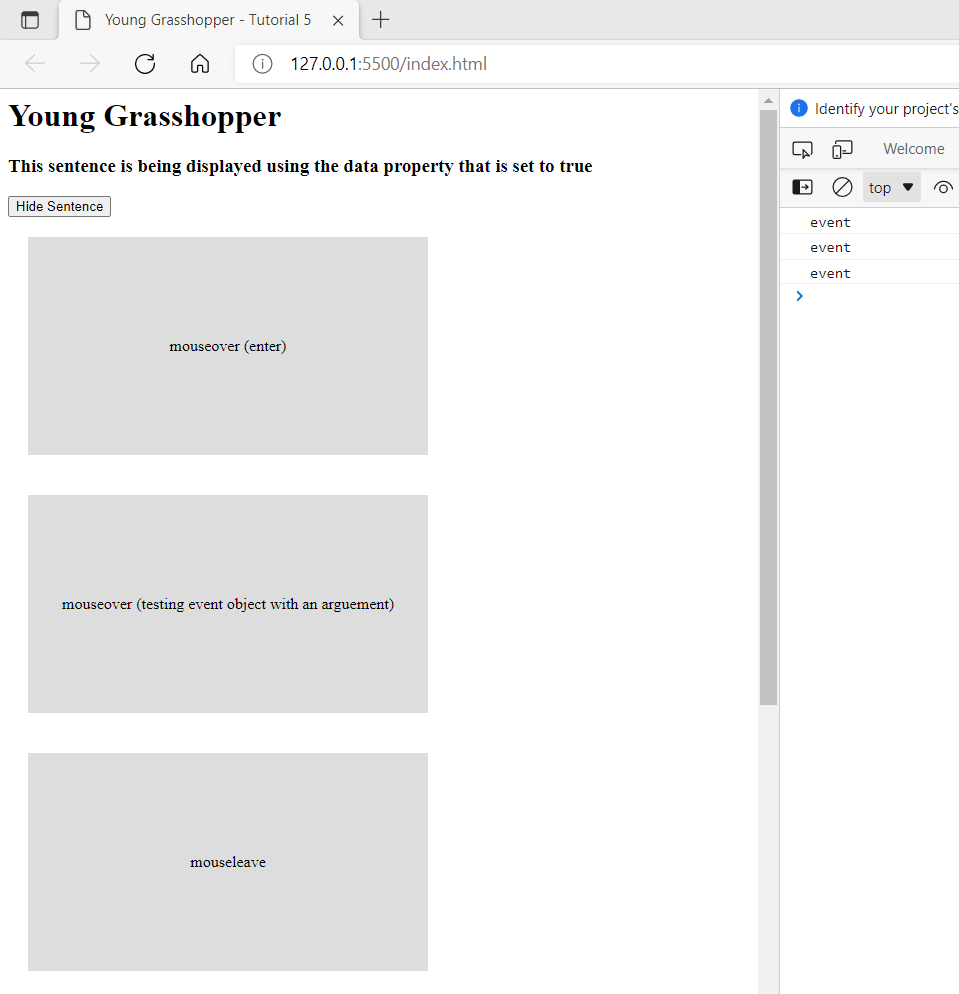
Look at the example below (**Line 44 to 45**)



**Hover over** the second element and move your mouse away from the element to display the output in the **console log**.

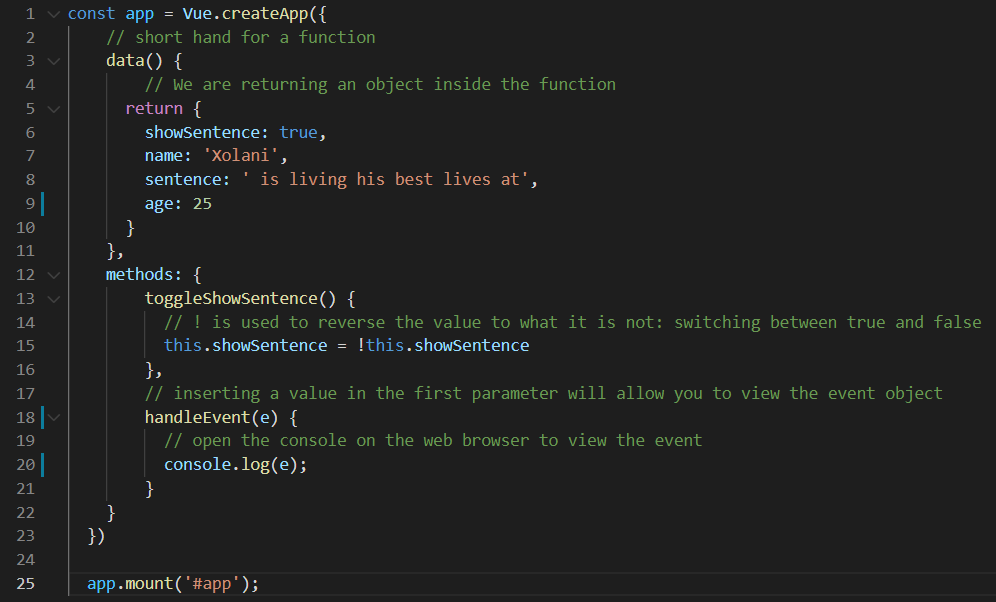
**Double click** the third element to display the output in the **console log**.

**Output:**



**Step 4:** In the **app.js** file, add a value in the parameter in the **handleEvent** function and replace the console log text with the parameter. Make sure it is the first parameter.

Look at the example below (**Line 18 to 21**)

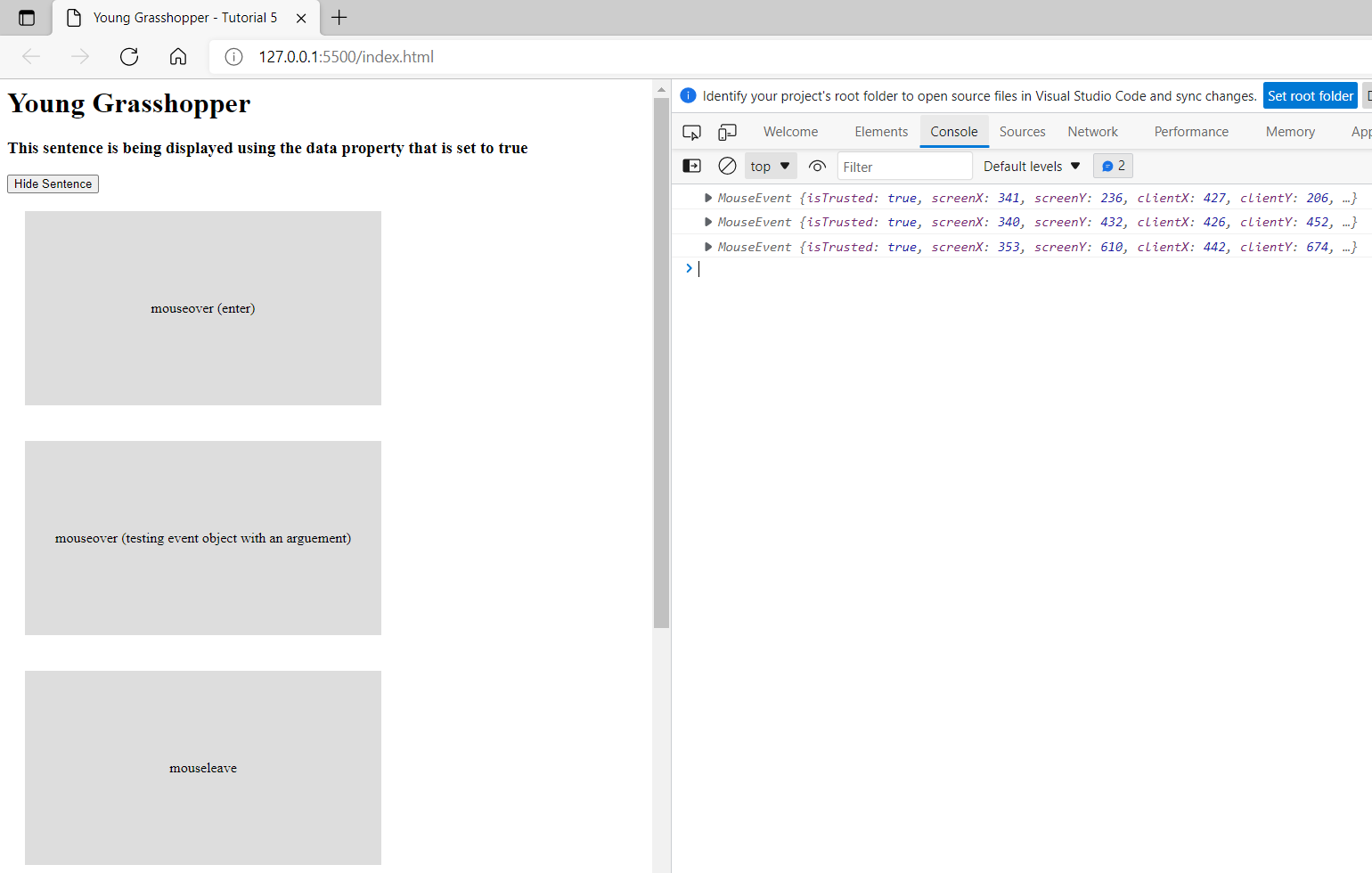


**Hover over** the first element to display the output in the **console log**.

**Hover over** the second element and move your mouse away from the element to display the output in the **console log**.

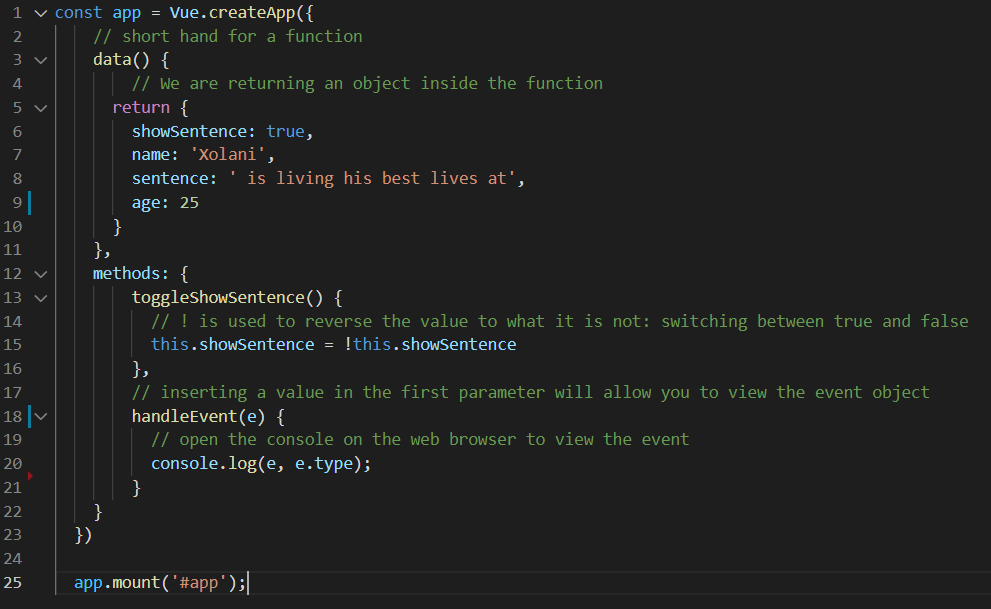
**Double click** the third element to display the output in the **console log**.

**Output:**



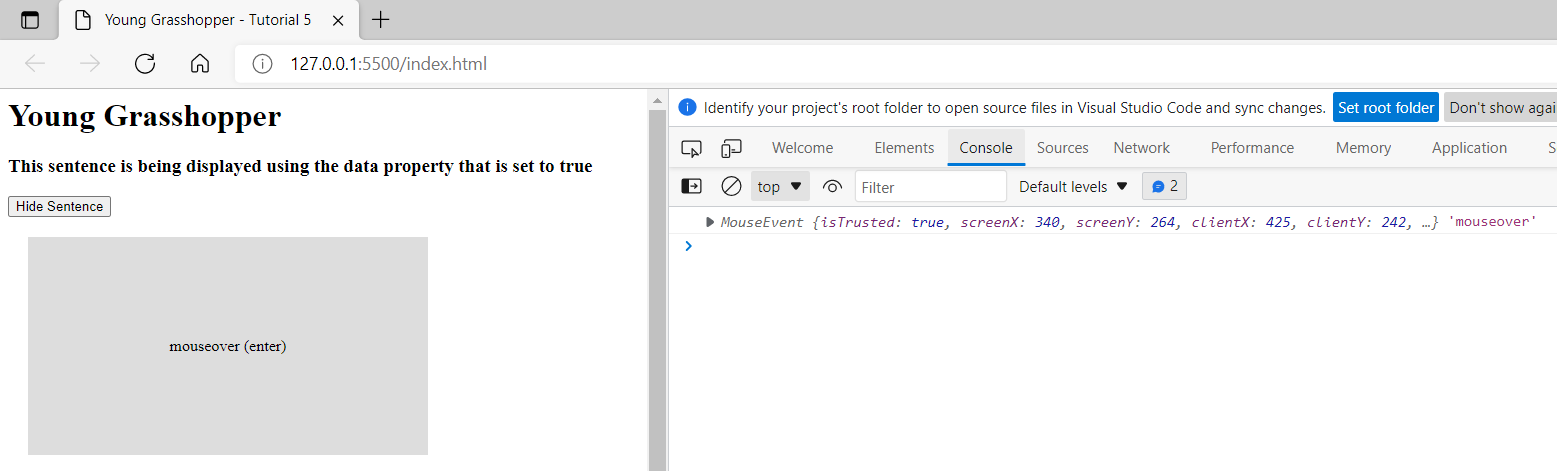
**Step 5:** In the **app.js** file, console log the event as well as the event type.

Look at the example below (**Line 18 to 21**)



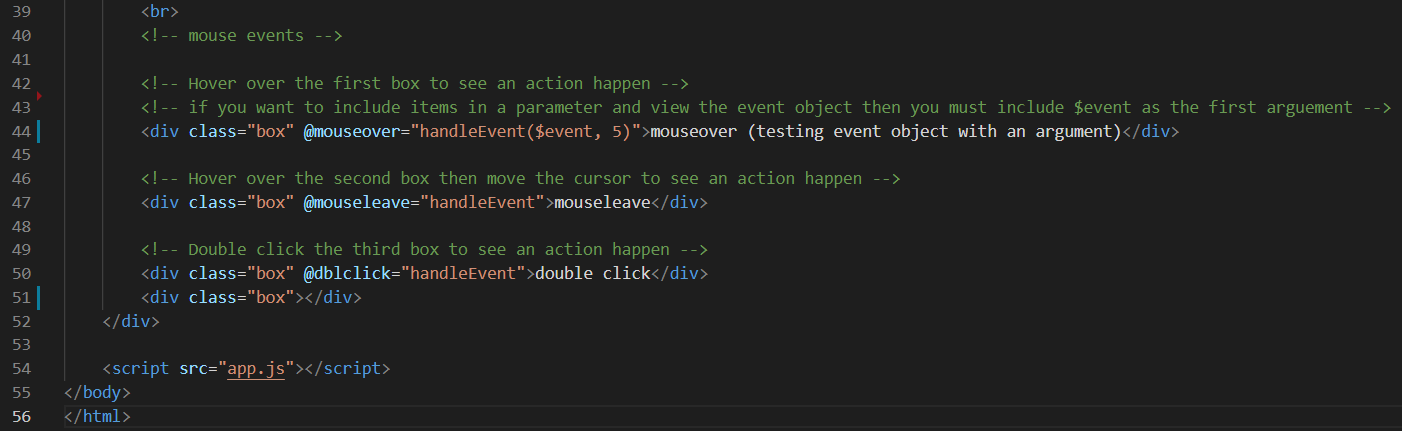
**Hover over** the first element to display the output in the **console log** that includes the **mouse event**. You can view the output of the other elements to see the type of mouse event.

**Output:**



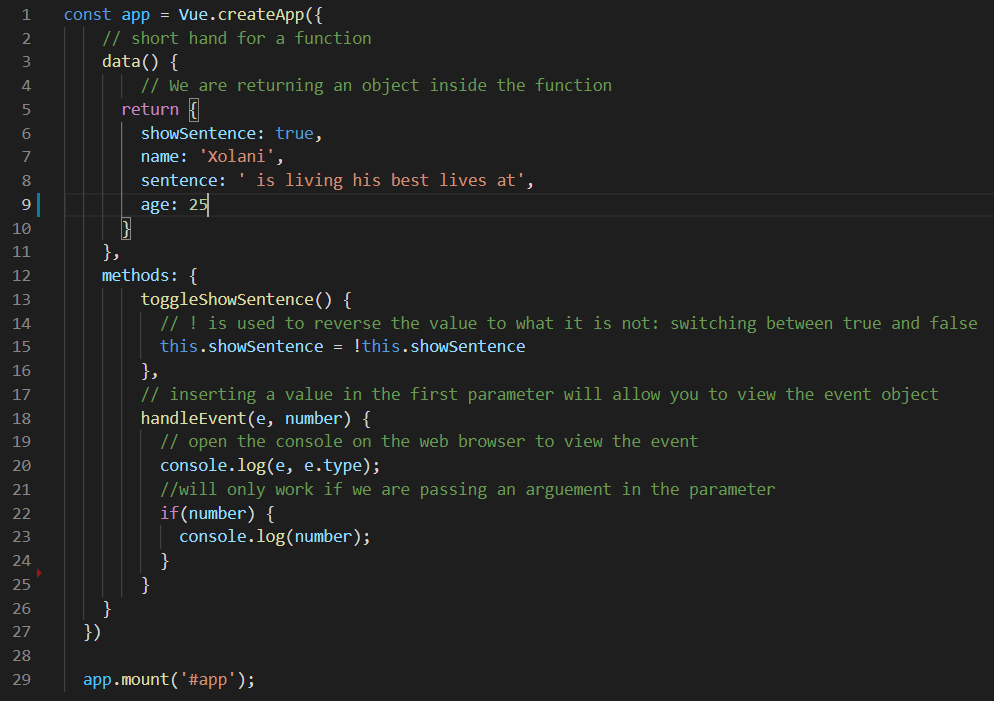
**Step 6:** In the **index.html** file, in the first **div** element, in the **mouseover** property calling the **handleEvent** function, pass through parameters by explicitly defining the **event object**, **$event**, as a **first argument** and **numeric** value as a **second argument**.

Look at the example below (**Line 46**)



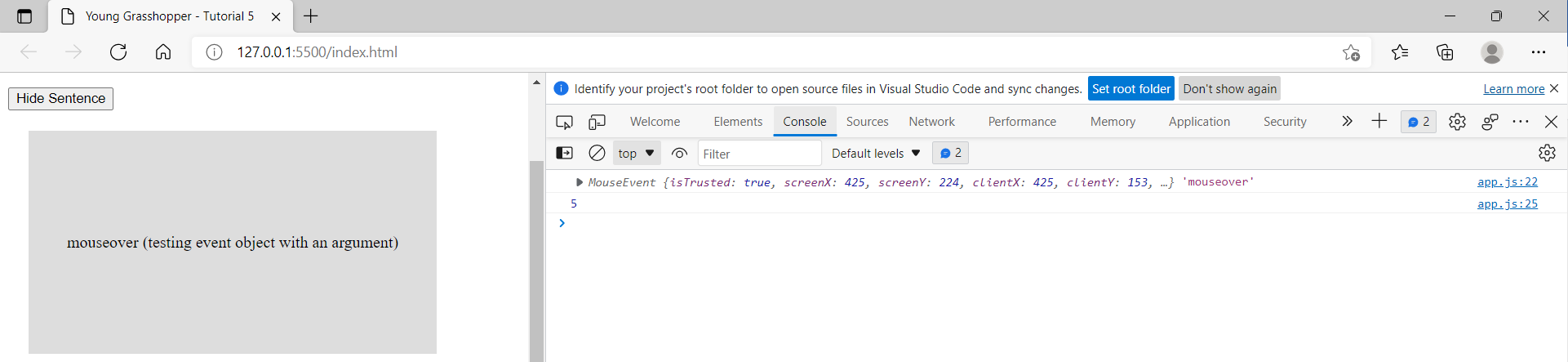
In the app.js file, in the **handleEvent** function, create a **condition** that **checks** whether a value exists in the **second parameter** and output the value in a **console log** if the value exists.

Look at the example below (**Line 18 to 25**)



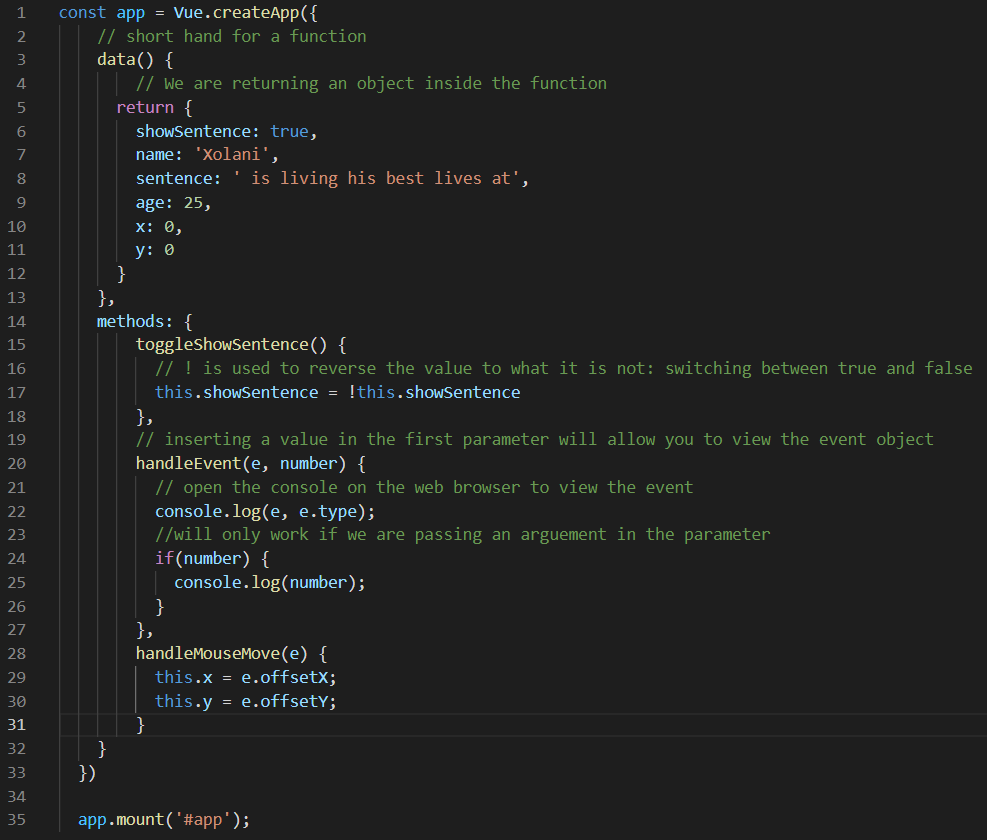
**Hover over** the first element to display the output in the **console log** that includes the **mouse event**. You can view the output of the other elements to see the type of mouse event.

**Output:**

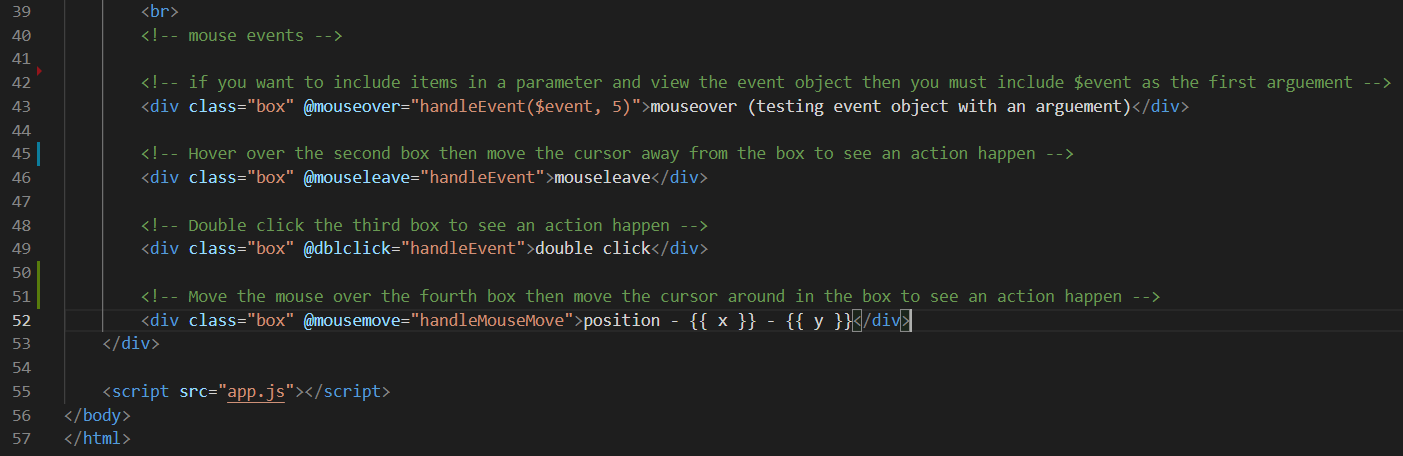


**Step 7:** In the **app.js** file, create two data properties, x and y, both with the value 0. Create a function called **handleMouseMove** that takes a single argument, e. Assign the x data property with the event of offsetX and the y data property with the event of offsetY.

Look at the example below (**Line 10 to 11, and line 28 to 31**)

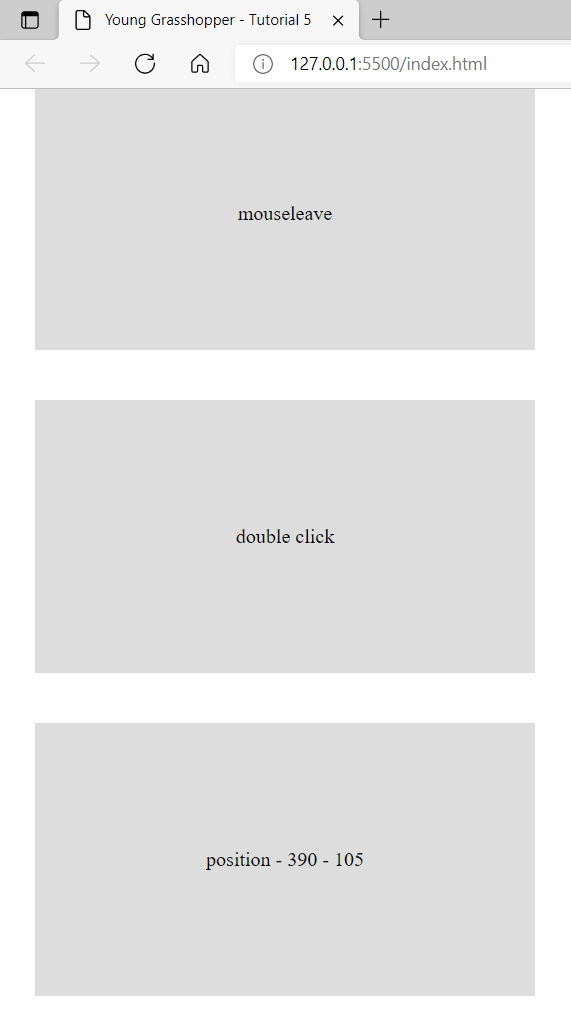


In the **index.html** file, create a **mousemove** property that references the function **handleMouseMove** in the fourth div. Display the x and y data properties in the div element from the **app.js** file.



**Move the mouse over** the fourth div to display the co-ordinate output in the **box**.

**Output:**



**You can view the code snippets from the** [**vue3-firebase-course**](https://github.com/younggrasshopperza/vue3-firebase-course) **repository under the Tutorial-5 branch.**