JavaScript

Calculator: Back end

We now want a way to get all the buttons in our calculator.html to our javascript file. So, create a js file called main.js, in this form, we will make a variable called calculator and another constant called keys and use the document.querySelector() method to retrieve the calculator and key classes.

Short quiz
What is the difference between document.querySelector() and document.querySelectorAll() method?

Next, we want to use the constant we defined earlier on (the keys), to add an event listener to the keys. We want to see the output of when we click the buttons on the calculator, okay? Here, we will introduce the arrow function into our js code!

Now, we said we want to add an event listener to our keys so that when we click the keys, it gives us an output; the event listener will take an action in a string form called click and a variable event which we will define inside of our arrow fuction as its parameters.

In our arrow function, the first thing we do is we give our condition- remember conditions in js? We will use the if conditions, so, if a key that matches this is clicked, do that. It's that simple! This condition will have one argument event.target.matches("key/ anything").

We need to know which type of key was pressed, and here is where our data-action comes into play. Remember the variable we passed as an argument before? The event variable? We now want to define it. Define another variable called keys inside the if statement and assign it to event.target. Now, does the event.target.matches("blablabla") make sense now?

Add another constant called key and chain it to the above constant and dataset and finally action. At this point, we want to make sure that when we click a button, we get an output on the console. So, number keys didn't have a data- action attribute, so if a button does not have a data-action attribute we know it's a number.

If it has, then it will either be a clear or an add or subract. Now, I will guide you on how to write one conditional statement for one of the buttons, say, clear

```
if (action === "clear" ) {
  console.log("Clear key!")
 }
```

Now, write the conditions for the remaining keys. Remember, those that have one class have one thing in common, use that to give them the same message! At this point, every calculator key should give you a response in the console window.