This keyword

This keyword is a special variable that is created for every execution context (every function) takes the value of (point to) the owner of the function in which the this keyword is used.
This keyword is not always same its value change according to how the function is called and its value is only assigned when the function is actually called.

FOUR DIFFERENT WAYS IN WHICH THE FUNCTION CAN BE CALLED

first way to call a function is as a method

So when we call a method then the this keyword inside that method will simply point to the object on which the method is called or in other words,it points to the object that is calling the method.

So the method
Here is the
calcAge
method

Method example:

```
const jonas = {
  name: 'Jonas',
  year: 1989,
  calcAge: function() {
    return 2037 - this.year;
  }
};
jonas.calcAge(); // 48
```

we then call the method and as you see inside the method, we used the this keyword. Now, according to what we just learned, what should be the value of the this keyword here? And that's right, it should be Jonas, because that is the object that is calling the method down there in the last line.

So in this case,
writing 'Jonas.year'
would have the exact
same
effect
as 'this.year'.
But doing it this way
is a way better solution,

SIMPLE FUNCTION CALL-

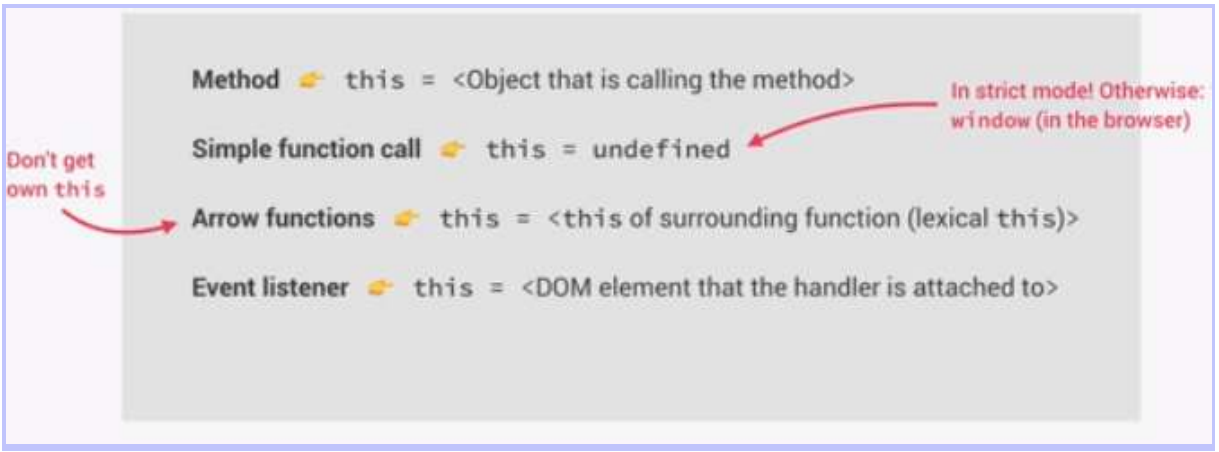
In this the this keyword points to undefined However, that is only valid for strict mode.So if you're not in strict mode, this will actually point to the global object, which in case of the browser is the window object. And that can be very problematic

ARROW FUNCTION -

It's an important kind of function that we need to consider,because, remember, arrow functions do not get their own 'this keyword'. Instead, if you use 'the this variable' in an arrow function, it will simply be the this keyword of the surrounding function.
So of the parent function

Event Listener-

And finally, if a function is called as an event listener, then the this keyword will always point to the DOM element that the handler function is attached to



What the, this keyword not

this does **NOT** point to the function itself, and also **NOT** the its variable environment!