

W03 - "this" in JavaScript

"this" is pretty simple - a key word with a value that changes depending on how it is called.

this in global context - defaults to window object

```
window { stop: function, open: function, alert: function, confirm: function, prompt: function }
```

this in object construction - this references the instance of the object

```
function Human (age) {  
  this.age = age  
}  
let greg = new Human (22);  
let thomas = new Human (24);  
console.log (greg); // this.age = 22  
console.log (thomas); // this.age = 24
```

this in an object method - this within any method refers to the object itself

this in a simple function - window object

this in an arrow function - same as the object around it.

this in Event Listeners - this is set to the element that fired the event

I NEED TO STUDY THIS AND ITS RAMIFICATIONS MORE CLOSELY

reference to object:

```
function LeetSpeaker (elem) {  
  return {  
    listenClick () {  
      const self = this;  
      elem.addEventListener ('click', function () {  
        self.speakLeet ();  
      })  
    },  
    speakLeet () { console.log ('1337 15 4W35ØM3')}  
  }  
}
```

using => notation
eliminates requirement
to provide reference for
this, since => notation
this reference parent

W03 - "this" in JavaScript

To use "this" to reference an object in an event listener, use `bind()` to manually create a "this" context.

→ `this.listener = this.speakLeet.bind(this);`

`bind()` allows changing the `this` context

say.bind({hippy: 'hipster'}, 1, 2, 3);

^ `this` arguments in bound function

→ `bind` does not work with `⇒` notation

...THIS IS USED IN OOP NOT FUNCTIONAL PROGRAMMING...