

[#javascript](#) : Understand "this" keyword

this is weird

Understanding JavaScript this keyword

You might have used "this" keyword in a real-life situation without realizing it. Suppose you are walking along with your brother and meet a friend along the way. This is how you would introduce your brother to your friend.

This is my brother.

"This" shows a reference to your brother. this represents the brother in the current sentence. It is the same way JavaScript uses this keyword.

```
const family = {  
  brother_name: "John",  
  brother: function() {  
    console.log(this)  
    console.log(` ${this.brother_name} is my brother.`)  
  }  
}  
  
family.brother\(\)
```

```
//{brother_name: 'John', brother: f}  
//John is my brother.
```

We used "this" keyword to refer to the family. Meaning this refers to its family object. It refers to the context where the anonymous function is called. And this will bind to the family object to return the name of the brother.

Now moving one step further with arrow functions,

```
const family = {  
  brother_name: "John",  
  brother: () => {  
    console.log(this)  
    console.log(`${this.brother_name} is my brother.`)  
  }  
}  
  
family.brother\(\)  
//Window {window: Window, self: Window, document: document, name: "",  
location: Location, ...}  
//undefined is my brother.
```

we get undefined.....

In the regular function, a function always defines its "this" value. Arrow functions treat "this" keyword differently. They don't define their own context since it doesn't have its own "this" context. They inherit that from the parent scope whenever you call this.

In the above code, "this" refers to a window object, because the context of a family object is a window. So this refers to the scope (the enclosing context) where the family object is present.

follow and add Rizwan Mushtaq Dhudhaal  in your friend list