

ARCH A4988

Coding for Spatial Practices



<Object Basics>

Learning Objectives

Learning Objectives

1. Compare objects to arrays as data structures.
2. Explain the difference between object properties and methods.
3. From within a method, reference properties of an object using `this`.
4. Define what ***this*** represents in the context of an event listener.
5. Utilize the event object to find out about events that have occurred, and prevent the default action on anchors and submit buttons.

Agenda

1. Review
2. Object Basics
3. The Event Object
4. Exercise

Review

Basics of Interaction

Casas Ou Museus?

Lina Bo Bardi, 1958

EN | PT



Primeiro as Casas ou Museus?

Tudo de uma só vez: as casas, as escolas, os museus, as bibliotecas. Uma planificação urbanística não pode prescindir dos problemas culturais se a construção de novos bairros, de novas casas, é a base do projeto de uma cidade (nas casas queremos incluir mercados, escolas, serviços coletivos, como saúde, correios, etc.) – o programa, ou melhor, a planificação de uma cidade não pode esquecer dois edifícios públicos, que ainda hoje são considerados um luxo intelectual: o museu e a biblioteca.

Point of interaction

EN | PT

Basics of Interaction

There are **two steps** to working with events:

1. Select your point of interaction (HTML element)
2. We set up an event listener using `.addEventListener`
3. We **define an event handler**, a function that gets passed to `.addEventListener`

Basics of Interaction

Select the point of interaction ... HTML elements

Casas Ou Museus?

Lina Bo Bardi, 1958

EN | PT

```
// HTML
```

```
<span class="en">EN</span> | <span class="pt">PT</span>
```

```
// JAVASCRIPT
```

```
const en_btn = document.querySelector(".en");
```

```
const pt_btn = document.querySelector(".pt");
```


Basics of Interaction

Add the event listener to the selected element

Casas Ou Museus?

Lina Bo Bardi, 1958

EN | PT

```
// HTML
```

```
<span class="en">EN</span> | <span class="pt">PT</span>
```

```
// JAVASCRIPT
```

```
const en_btn = document.querySelector(".en");
```

```
const pt_btn = document.querySelector(".pt");
```

```
en_btn.addEventListener('click', translateToEnglish);
```

```
pt_btn.addEventListener('click', translateToPortuguese);
```

Basics of Interaction

Write a function that includes the steps when link is clicked

Casas Ou Museus?

Lina Bo Bardi, 1958

EN | PT

```
// HTML
```

```
<span class="en">EN</span> | <span class="pt">PT</span>
```

```
// JAVASCRIPT
```

```
const en_btn = document.querySelector(".en");
```

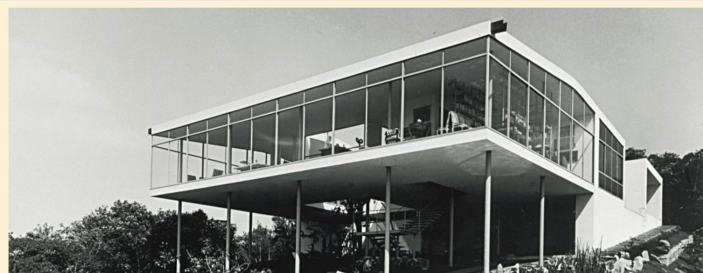
```
const pt_btn = document.querySelector(".pt");
```

```
en_btn.addEventListener('click', translateToEnglish);
```

```
pt_btn.addEventListener('click', translateToPortuguese);
```

Basics of Interaction

What actually happens when the PT link is clicked?



What should come first, houses or museums?

Everything at once: the houses, the schools, the museums, the libraries. Urban Planning cannot ignore cultural issues. If in the construction of new neighbourhoods, new housing forms the basis of the city plan (and by housing we also mean the market, the schools and the public services like the hospital

The problem of the museum has to be tackled today on 'didactic' and 'technical' grounds. These foundations are essential if the museum is not to become petrified, that is, entirely useless.

The experience gained in this field with the São Paulo Museum of Art can be of

```
// HTML
```

```
<section id="english_content" class="english">...</section>  
<section id="portuguese_content" class="portuguese">...</section>
```

```
// JAVASCRIPT
```

```
const english_text = document.querySelector("#english_content");  
const portuguese_text = document.querySelector("#portuguese_content");
```

```
function translateToPortuguese(){...}  
function translateToEnglish(){...}
```

Basics of Interaction

What actually happens when the PT link is clicked?

```
// HTML
```

```
<section id="english_content" class="english"></section>  
<section id="portuguese_content" class="portuguese"></section>
```

```
// JAVASCRIPT
```

```
const english_text = document.querySelector("#english_content");  
const portuguese_text = document.querySelector("#portuguese_content");
```

```
function translateToPortuguese(){  
  let h1 = document.querySelector('h1');  
  h1.textContent = 'casas ou museus?';  
  
  english_text.style.display = 'none';  
  portuguese_text.style.display = 'flex';  
}
```

Object Basics

Object Basics

Easier Access to Data

Let's say we are developing a website for an art gallery or a platform for a museum.

Using a regular array, we would store each artists' information:

```
let artist = [  
  "Frida",  
  "Kahlo",  
  "Surrealism",  
  1907,  
  "Mexico"  
];
```

Object Basics

Easier Access to Data

Now, let's take a look at how we could represent this artist with an object.

```
let artist = {  
  first_name: "Frida",  
  last_name: "Kahlo",  
  year_of_birth: 1907,  
  country_of_birth: "Mexico"  
};
```

Object Basics

What are objects good for?

Objects serve two main purposes in JavaScript:

- They act as a simple, structured data store that access values using keys instead of indices.
- They provide a fundamental programming paradigm that helps us structure and categorize code

Object Basics

Properties & Methods

Each object can also have its own properties and methods.

Object Basics

Properties

If a function is part of an object, it becomes a method. If a variable is part of an object, it becomes a property. In other words, properties tell us about an object.

```
let artist = {  
  first_name: "Frida",  
  last_name: "Kahlo",  
  year_of_birth: 1907,  
  country_of_birth: "Mexico",  
  fullName: function() {  
    return first_name + ' ' +  
    last_name;  
  }  
};
```

————— property

Object Basics

Method

Methods are used to represent how people interact with an object in the real world. In other words, methods are the actions that can be performed on objects.

```
let artist = {  
  first_name: "Frida",  
  last_name: "Kahlo",  
  year_of_birth: 1907,  
  country_of_birth: "Mexico",  
  fullName: function() { —— method  
    return first_name + ' ' +  
    last_name;  
  }  
};
```

Object Basics

Method

We can use methods to:

1. Retrieve the values of an object's properties to find out something about the object
2. Update the values of an object's properties

```
let artist = {  
  first_name: "Frida",  
  last_name: "Kahlo",  
  year_of_birth: 1907,  
  country_of_birth: "Mexico",  
  fullName: function() { —— method  
    return first_name + ' ' +  
    last_name;  
  }  
};
```

Object Basics

Creating Objects

Let's say we want to create objects for several popular artists, starting with Frida Kahlo.

We might want to add some properties for Frida Kahlo: a first name, last name, and country of birth.

```
let artist = {  
  // properties  
  firstName: "Frida",  
  lastName: "Kahlo",  
  countryOfBirth: "Mexico"  
};
```

Object Basics

Creating Objects

We might also want to add a method to our artist object which will return a short bio about Frida Kahlo.

```
let artist = {  
  // properties  
  firstName: "Frida",  
  lastName: "Kahlo",  
  countryOfBirth: "Mexico",  
  
  // methods  
  printShortBio: function() {  
    return this.firstName + " " +  
    this.lastName + " was born in" +  
    this.countryOfBirth;  
  }  
};
```

Object Basics

Dot Notation

We can access values using the object name, followed by a period, followed by the name of the property we want to access.

```
// Store 'Frida' in the  
artistFirstName variable.
```

```
let artistFirstName =  
artist.firstName;
```

```
// Store 'Kahlo' in the  
artistLastName variable.
```

```
let artistLastName =  
artist.lastName;
```

Object Basics

Updating Properties Using Dot Notation

To update values using dot notation, use the name of the object (in this case, `artist`), followed by the name of the property we want to update (in this case, `.firstName` or `.lastName`). Then use the assignment operator (`=`) followed by the new value.

```
artist.firstName = 'Magdalena Carmen  
Frida';
```


Object Basics

Adding New Properties Using Dot Notation

The syntax we use to update a property can also add a new one.

```
artist.spouse = 'Diego Rivera';
```

If the property we are trying to update doesn't yet exist, it will be added automatically.

Object Basics

Square Bracket Notation

We can also access values using square bracket syntax.

The name of our object is `artist`, followed by the property name, contained within quotes and square brackets.

```
// Store 'Frida' in the  
artistFirstName variable.  
let artistFirstName =  
artist['firstName'];
```

Object Basics

Accessing Methods

If we wanted to access – or call, our `printShortBio()` method – we could do so using the following syntax:

```
let bio = artist.printShortBio();  
  
> Frida Kahlo was born in Mexico
```

Object Basics

Iterating through an object

Like arrays, you can use a loop to iterate through an object.

```
let items = { a: 1, b: 2, c: 3 }  
  
for (property in items) {  
  console.log( items[property] );  
}
```

```
> 1
```

```
> 2
```

```
> 3
```

Object Basics

Iterating over an array of objects

Iterator functions apply a function to each element of an array.

`forEach()` takes a function as an argument and applied the called function to each element of an array.

```
let nums = [1, 2, 3, 4, 5, 6];
```

```
function square (num) {  
  console.log( num * num );  
}
```

```
nums.forEach(square)
```

```
> 1
```

```
> 4
```

```
> 9
```

Object Basics

Putting array elements in order, sort,

The `sort()` function sorts data lexicographically assuming the data elements are strings.

If you need to sort data elements that are numbers, you'll have to write an ordering function.

```
let fruits = ['grapes',  
             'watermelon', 'apple']
```

```
fruits.sort();  
console.log( fruits );
```

```
> apple  
> grapes  
> watermelon
```



Create your own JavaScript object representing your favorite movie.

Instructions:

Go to the **starter_code_week_12** folder and complete the **my_first_object** exercise.

Starter code:

<https://jsbin.com/zagezod/edit?js,console>



Iterate Over an Array of Objects



1. Iterate over this array of objects and print the name to the console.
2. Talk through the steps to render the name to the web page.
3. Now, render the name to the web page.

Instructions:

Go to the **starter_code_week_12** folder and complete the **search_names** exercise.

Adri
Becky
Chris
Dillon
Evan
Frank
Georgette
Hugh
Igor
Jacoby
Kristina
Lemony
Matilda
Nile

The Event Object

The Event Object

Accessing the Event Object

How do we gain access to the event object?

First, we need to pass the event object as a parameter.

EVENT OBJECT

Alice was beginning to get very tired of sitting by her sister on the bank, and of having nothing to do. Once or twice she had peeped into the book her sister was reading, but it had no pictures or conversations in it, "and what is the use of a book," thought Alice, "without pictures or conversations?"

So she was considering in her own mind (as well as she could, for the day made her feel very sleepy and stupid), whether the pleasure of making a daisy-chain would be worth the trouble of getting up and picking the daisies, when suddenly a White Rabbit with pink eyes ran close by her.

There was nothing so very remarkable in that, nor did Alice think it so very much out of the way to hear the Rabbit say to itself, "Oh dear! Oh dear! I shall be too late!" But when the Rabbit actually took a watch out of its waistcoat-pocket and looked at it and then hurried on, Alice started to her feet, for it flashed across her mind that she had never before seen a rabbit with either a waistcoat-pocket, or a watch to take out of it, and, burning with curiosity, she ran across the field after it and was just in time to see it pop down a large rabbit-hole, under the hedge. In another moment, down went Alice after it, and

[VIEW COMMENTS](#)

```
const viewComments = (event) => {  
  console.log(event)  
}
```

```
circle.addEventListener('click',  
viewComments);
```

The Event Object

Accessing the Event Object

How do we gain access to the event object?

First, we need to pass the event object as a parameter.

EVENT OBJECT

Alice was beginning to get very tired of sitting by her sister on the bank, and of having nothing to do. Once or twice she had peeped into the book her sister was reading, but it had no pictures or conversations in it, "and what is the use of a book," thought Alice, "without pictures or conversations?"

So she was considering in her own mind (as well as she could, for the day made her feel very sleepy and stupid), whether the pleasure of making a daisy-chain would be worth the trouble of getting up and picking the daisies, when suddenly a White Rabbit with pink eyes ran close by her.

There was nothing so very remarkable in that, nor did Alice think it so very much out of the way to hear the Rabbit say to itself, "Oh dear! Oh dear! I shall be too late!" But when the Rabbit actually took a watch out of its waistcoat-pocket and looked at it and then hurried on, Alice started to her feet, for it flashed across her mind that she had never before seen a rabbit with either a waistcoat-pocket, or a watch to take out of it, and, burning with curiosity, she ran across the field after it and was just in time to see it pop down a large rabbit-hole, under the hedge. In another moment, down went Alice after it, and

[VIEW COMMENTS](#)

Reader #1: Great read!

Reader #2: One of my favorite books!

Reader #3: A rabbit with a watch.

The Event Object

Preventing Default Behavior

Some events, such as clicking on a link or submitting a form, are meant to take you to another page.

But, maybe you don't want to go to another page. Maybe, you want an image to appear on the page.

```
const viewImage = (event) => {  
  console.log(event)  
  event.preventDefault();  
  let image =  
    document.querySelector('.image')  
    image.className = 'show-image'  
}  
  
button.addEventListener('click',  
  viewImage);
```

The Event Object

target

Some events, such as clicking on a link or submitting a form, are meant to take you to another page.

But, maybe you don't want to go to another page. Maybe, you want an image to appear on the page.

```
<a href="#">Show Image</a>
```

```
const viewImage = (event) => {  
  console.log(event.target)  
}
```

Exercise

The Event Object

Build a grid of flowers using provided in the starter code. The images are BIG! How can you solve for that?

Instructions:

Go to the **starter_code_week_12** folder and complete the **flower** exercise.

Flowers

Tulip

yellow



Daffodil

yellow



Sunflower

yellow



Bluebell

blue

