Object Methods, “This”

* Objects are usually created to represent entities of the real world, like users, orders and so on
* A function that is a property of an object is called its method.
* When we write our code using objects to represent entities, that’s called [object-oriented programming](https://en.wikipedia.org/wiki/Object-oriented_programming),
  + So, with this knowledge. JavaScript is an Object-Oriented Programming language that works best with HTML and CSS? Can JavaScript live alone without Web Development Language?
* It’s common that an object method needs to access the information stored in the object to do its job.
  + **To access the object, a method can use the this keyword.**
* In JavaScript, keyword this behaves unlike most other programming languages. It can be used in any function, even if it’s not a method of an object.
  + Then This works only with the what it is attached to.
* If you come from another programming language, then you are probably used to the idea of a "bound this", where methods defined in an object always have this referencing that object.
  + The concept of run-time evaluated this has both pluses and minuses. On the one hand, a function can be reused for different objects. On the other hand, the greater flexibility creates more possibilities for mistakes.
* Arrow functions are special: they don’t have their “own” this. If we reference this from such a function, it’s taken from the outer “normal” function.

This in JavaScript

* this is a keyword whose value changes depending on how a function gets called.
  + this in global context
    - When this is called outside of any function, in a global context, this defaults to the Window object in the browser.
  + this in object construction
    - When you create a new instance of an object with the new keyword, this refers to the instance.
  + this in an object method
    - Methods are fancy words for functions that are associated with an object
  + this in a simple function
    - Anonymous functions written in the same form are also considered simple functions.
  + this in an arrow function
    - this in an arrow function is always the same as this around it (in its immediate scope)
  + this in an event listener
  + Does other Object Oriented Languages also have this style and this many of this(s)?
* bind is a method that is present in every function. It allows you to change the this context.
* bind doesn’t work with arrow functions.

CHAPT 5 Objects

* An object in JavaScript is a self-contained set of related values and functions.
* If a property’s value is a function, it is known as amethod.
* JavaScript objects are similar to a hash or associative array in other programming languages (or even a JavaScript map ).
  + JavaScript Map?
* Objects are often used to keep any related information and functionality together in the same place.
* An object literal is an object that is created directly in the language by wrapping all its properties and methods in curly braces {}
  + So object literal is like a combination of class and a variable? Is it like forming an Array?
* All objects are mutable at any time when a program is running. This means that its properties and methods can be changed or removed, and new properties and methods can be added to the object, even if it was declared using const .
* You can access the properties of an object using the dot notation that we’ve already seen in previous chapters.
* You can also access an object’s properties using bracket notation
* Dot notation is much more common, but bracket notation has a few advantages: it's the only way to access nonstandard property and method names that don’t follow the variable naming rules.
* The ability to create objects with computed property keys was introduced in ES6.
  + When did ES6 come out?
* To call an object’s method we can also use dot or bracket notation. Calling a method is the same as invoking a function, so parentheses need to be placed after the method name
* The in operator can be used to check whether an object has a particular property.
* Another way is to use the hasOwnProperty() method. As mentioned earlier, objects can inherit properties from other objects, so all objects have a method called hasOwnProperty()
* New properties and methods can be added to objects at any time in a program. This is done by simply assigning a value to the new property.
* It's important to note that properties don’t always appear in the order they were entered. An object is not an ordered list like an array, set or map, so you should never rely on the properties being in a certain order.
* Any property can be removed from an object using the delete operator.
* It’s even possible for an object to contain other objects. These are known as nested objects
* The values in nested objects can be accessed by referencing each property name in order using either dot or bracket notation
* objects are assigned byreference. This means that if a variable is assigned to an object that already exists, it will simply point to the exact same space in memory.