Functions

bind
bindAll
partial
memoize
delay
defer

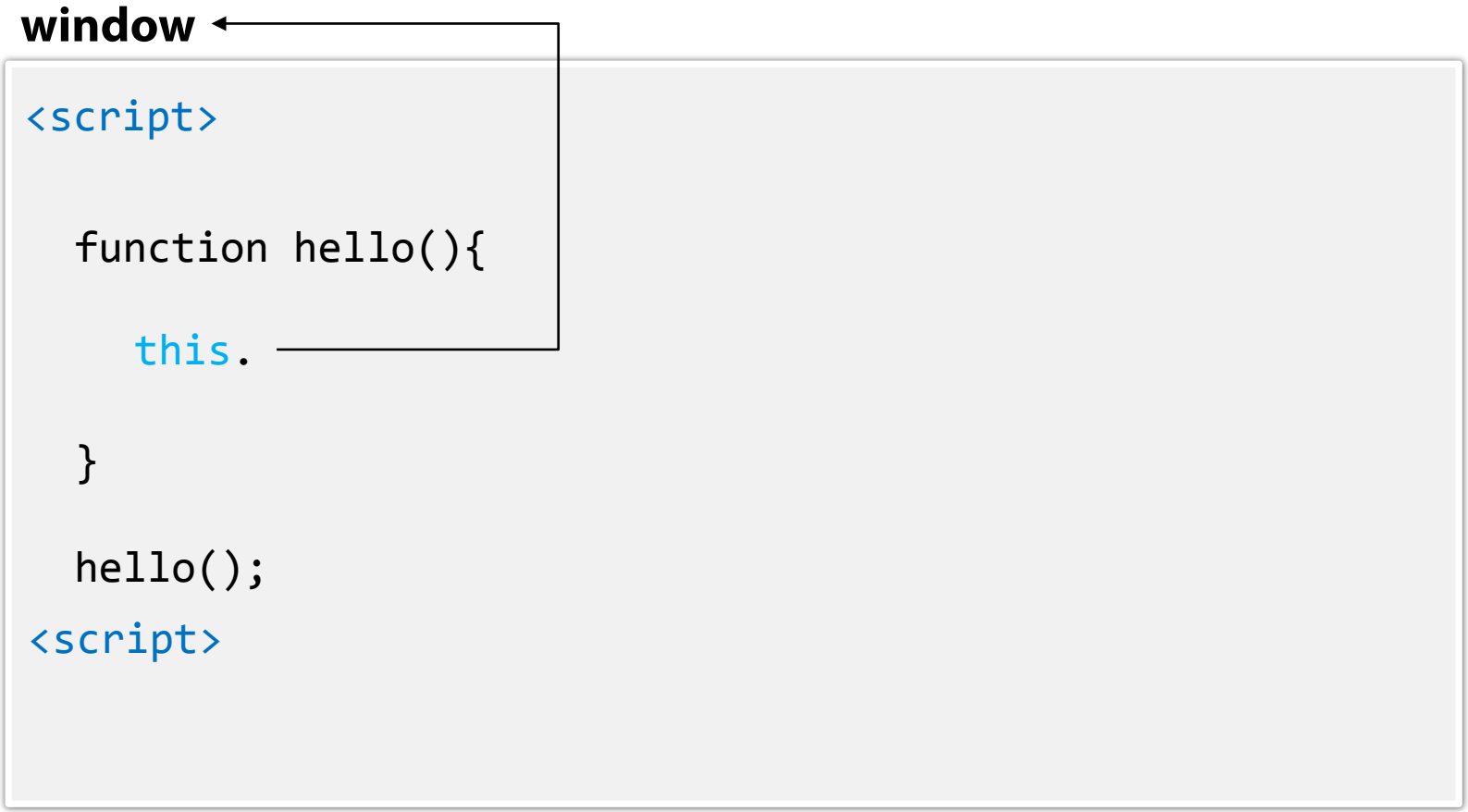
throttle
debounce
once
after
wrap
compose

JavaScript `this`

Mozilla Developer Network: this

https://developer.mozilla.org/en-US/docs/JavaScript/Reference/Operators/this

JavaScript `this`



JavaScript `this`

window

```
<script>

  function hello(){
     this.
  }

  $('#button-save').click(hello);


<script>
...
<button id='#button-save'>Save</button>
```

Without Binding

window

```
<script>
  var vm = {
    name: 'Craig',
    sayHi: function(){
      alert('Hello, ' + this.name);
    }
  };

  $('#button-save').click(vm.sayHi);
<script>
...
<button id='#button-save'>Save</button>
```

With Binding

window

```
<script>
  var vm = {
    name: 'Craig',
    sayHi: function(){
      alert('Hello, ' + this.name);
    }
  };

  _.bindAll(vm);

  $('#button-save').click(vm.sayHi);
<script>
...
<button id='#button-save'>Save</button>
```

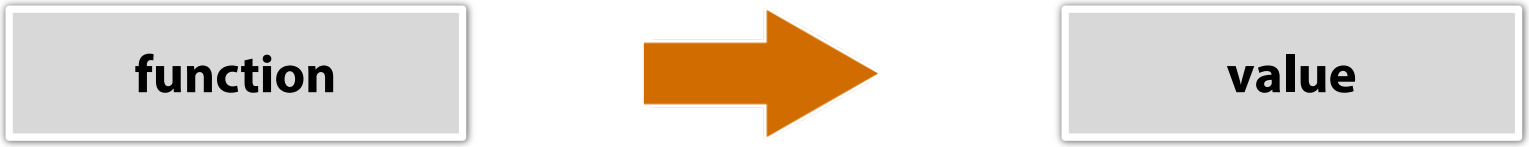
Memoization

Memoization is derived from the Latin word memorandum (to be remembered), and thus carries the meaning of turning [the results of] a function into something to be remembered.

Source: http://en.wikipedia.org/wiki/Memoization

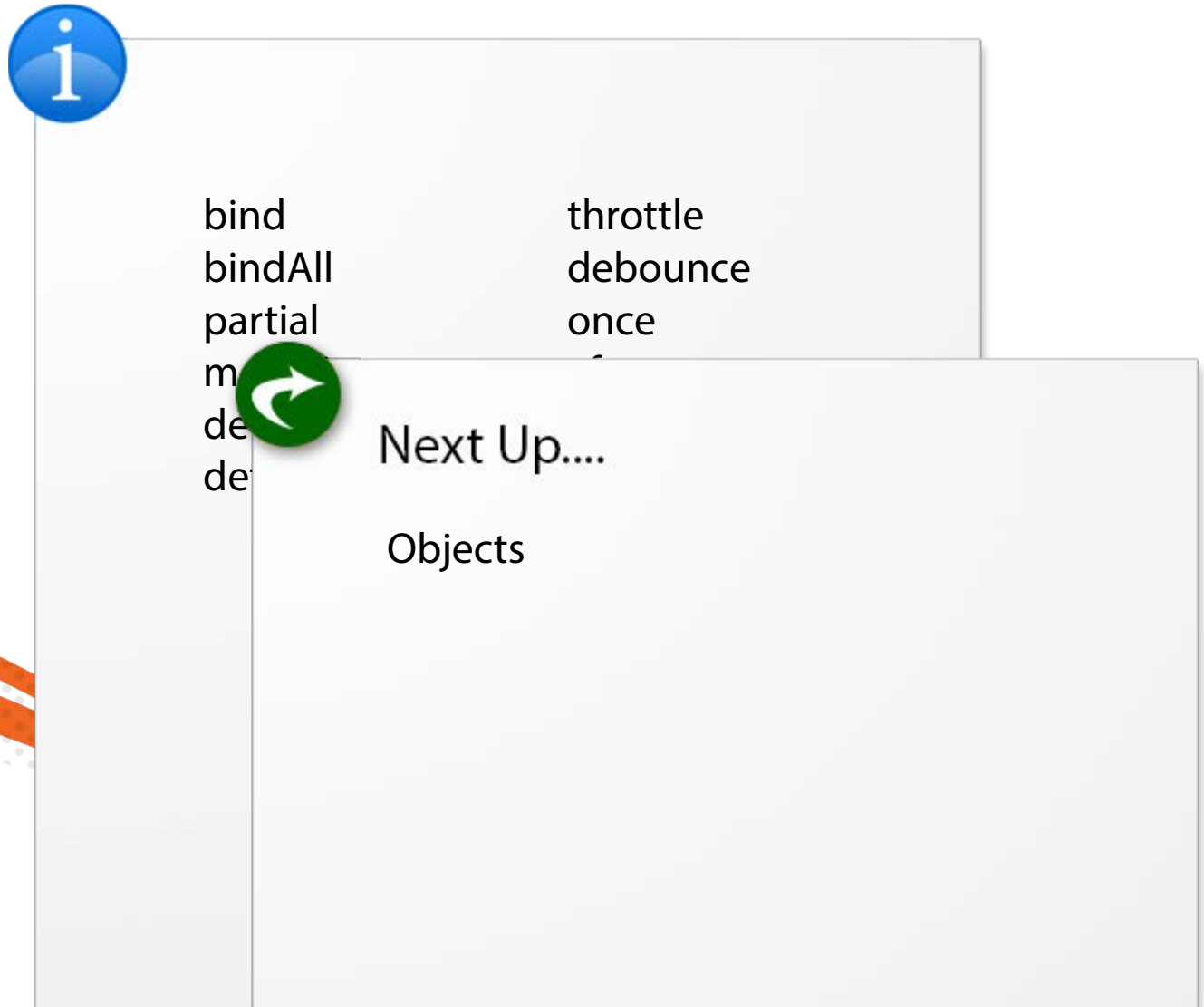
Memoization

First execution



Memoization

Summary


bind
bindAll
partial
throttle
debounce
once
Next Up....
Objects