# jQuery Event Handling Part 1

SENG 4640
Software Engineering for Web Apps
Winter 2023

Sina Keshvadi
Thompson Rivers University

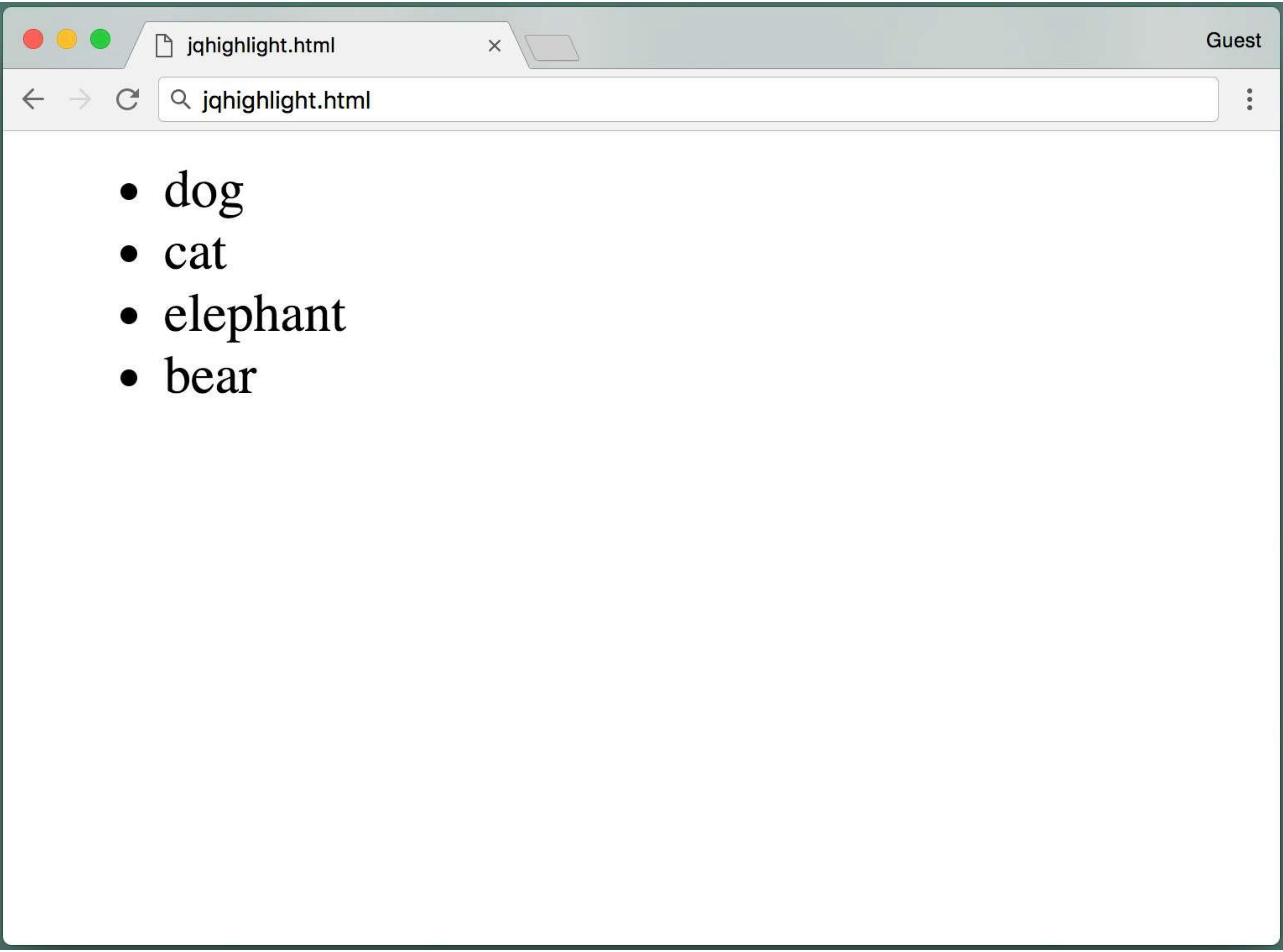
# Review

- Previously we saw how to use jQuery to select and modify DOM elements and add event listeners

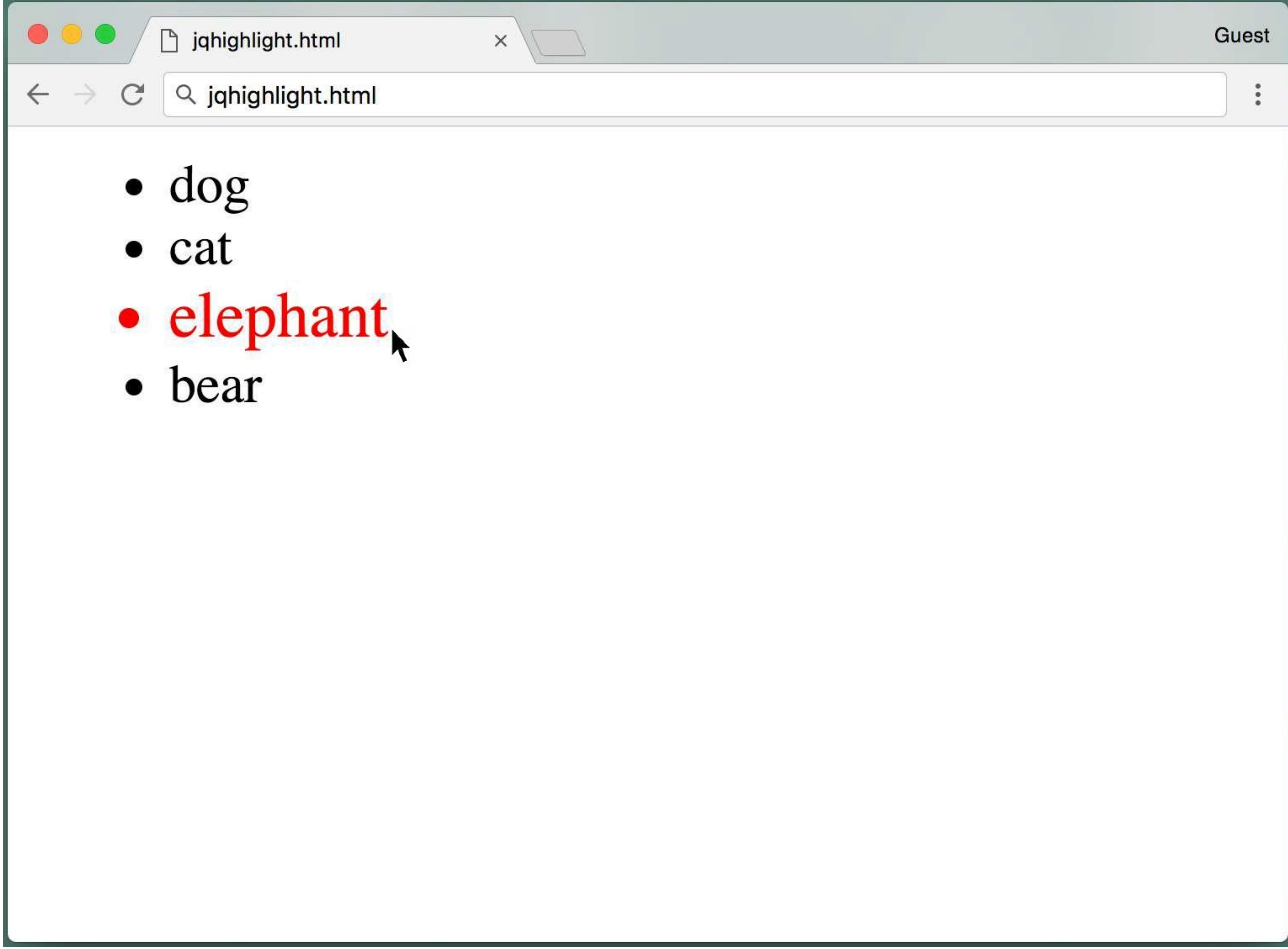
- How else can we specify selectors and add event listeners?

# jQuery

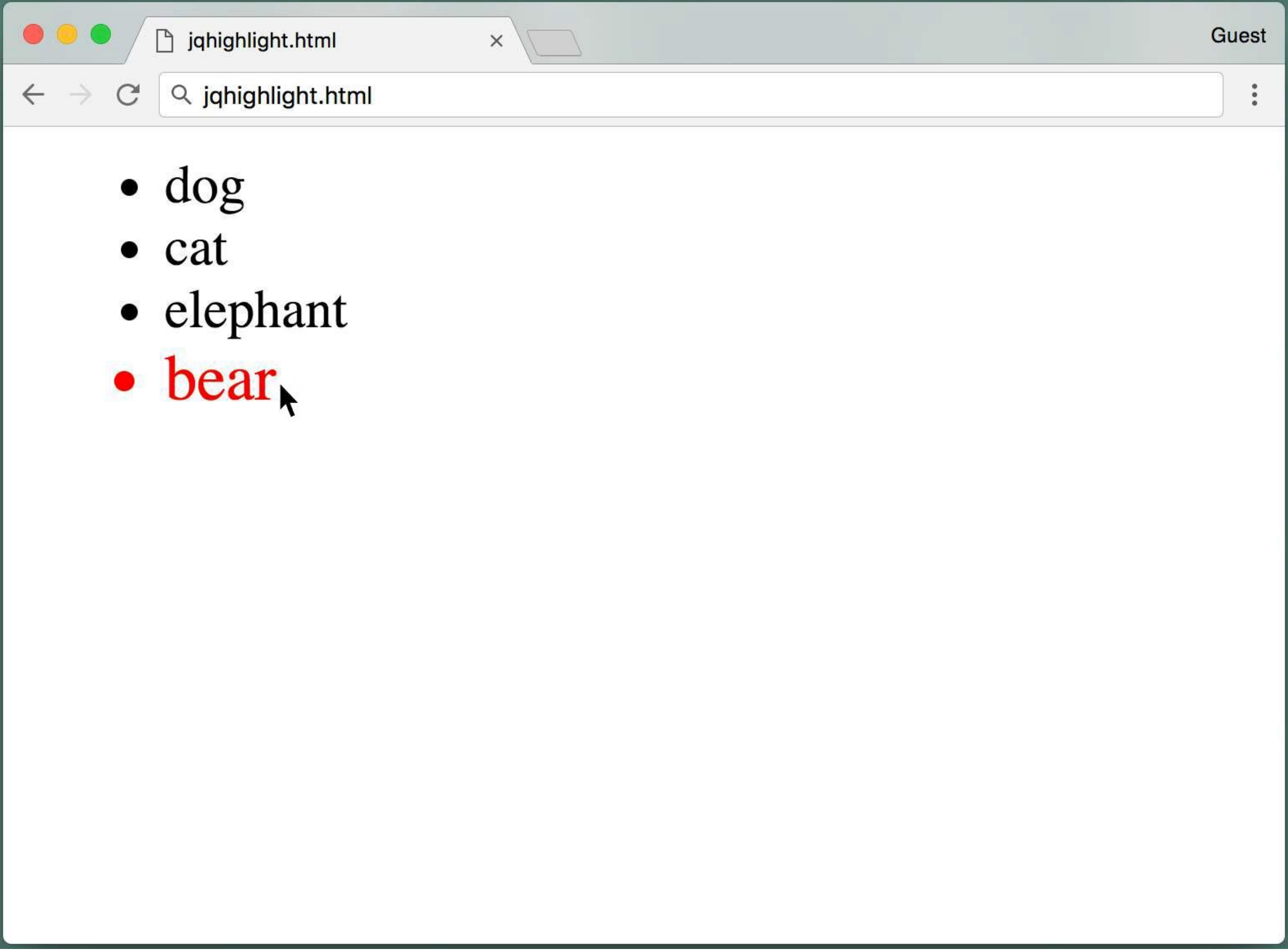
## Example - Highlight mouse over Item



- dog
- cat
- elephant
- bear



- dog
- cat
- elephant
- bear



- dog
- cat
- elephant
- bear

```
<html>
<head><script src="jquery.js"></script></head>
<body>
<ul>
<li>dog</li>
<li>cat</li>
<li>elephant</li>
<li>bear</li>
</ul>

<script>

$('li').mouseenter(function() {
  $(this).css('color', 'red');
  $(this).css('font-size', '120%');
});

$('li').mouseleave(function() {
  $(this).css('color', 'black');
  $(this).css('font-size', '100%');
});

</script>

</body>
</html>
```

every html element in jQuery has a css function to set the style using css notation

```
<html>
<head><script src="jquery.js"></script></head>
<body>
<ul>
<li>dog</li>
<li>cat</li>
<li>elephant</li>
<li>bear</li>
</ul>

<script>

$('li').mouseenter(function() {
  $(this).css('color', 'red');
  $(this).css('font-size', '120%');
});

$('li').mouseleave(function() {
  $(this).css('color', 'black');
  $(this).css('font-size', '100%');
});

</script>

</body>
</html>
```

add jQuery to use jQuery

```
<html>
<head><script src="jquery.js"></script></head>
<body>
<ul>
<li>dog</li>
<li>cat</li>
<li>elephant</li>
<li>bear</li>
</ul>

<script>

$('li').mouseenter(function() {
  $(this).css('color', 'red');
  $(this).css('font-size', '120%');
});

$('li').mouseleave(function() {
  $(this).css('color', 'black');
  $(this).css('font-size', '100%');
});

</script>

</body>
</html>
```

mouseenter event handler for all li items

```
<html>
<head><script src="jquery.js"></script></head>
<body>
<ul>
<li>dog</li>
<li>cat</li>
<li>elephant</li>
<li>bear</li>
</ul>

<script>

$('li').mouseenter(function() {
  $(this).css('color', 'red');
  $(this).css('font-size', '120%');
});

$('li').mouseleave(function() {
  $(this).css('color', 'black');
  $(this).css('font-size', '100%');
});

</script>

</body>
</html>
```

note: there are various ways to change font-size
Check Here

```
<html>
<head><script src="jquery.js"></script></head>
<body>
<ul>
<li>dog</li>
<li>cat</li>
<li>elephant</li>
<li>bear</li>
</ul>

<script>

$('li').mouseenter(function() {
  $(this).css('color', 'red');
  $(this).css('font-size', '120%');
});

$('li').mouseleave(function() {
  $(this).css('color', 'black');
  $(this).css('font-size', '100%');
});

</script>

</body>
</html>
```

mouseleave event is when a user away mouse from an item

```
<html>
<head><script src="jquery.js"></script></head>
<body>
<ul>
<li>dog</li>
<li>cat</li>
<li>elephant</li>
<li>bear</li>
</ul>

<script>

$('li').mouseenter(function() {
  $(this).css('color', 'red');
  $(this).css('font-size', '120%');
});

$('li').mouseleave(function() {
  $(this).css('color', 'black');
  $(this).css('font-size', '100%');
});

</script>

</body>
</html>
```

**hover** is a shortcut in jQuery for the common combination of mouseenter and mouseleave

```
<html>
<head><script src="jquery.js"></script></head>
<body>
<ul>
<li>dog</li>
<li>cat</li>
<li>elephant</li>
<li>bear</li>
</ul>

<script>

$('li').hover( function() {
    $(this).css('color', 'red');
    $(this).css('font-size', '120%');
  },
  function() {
    $(this).css('color', 'black');
    $(this).css('font-size', '100%');
});

</script>

</body>
</html>
```

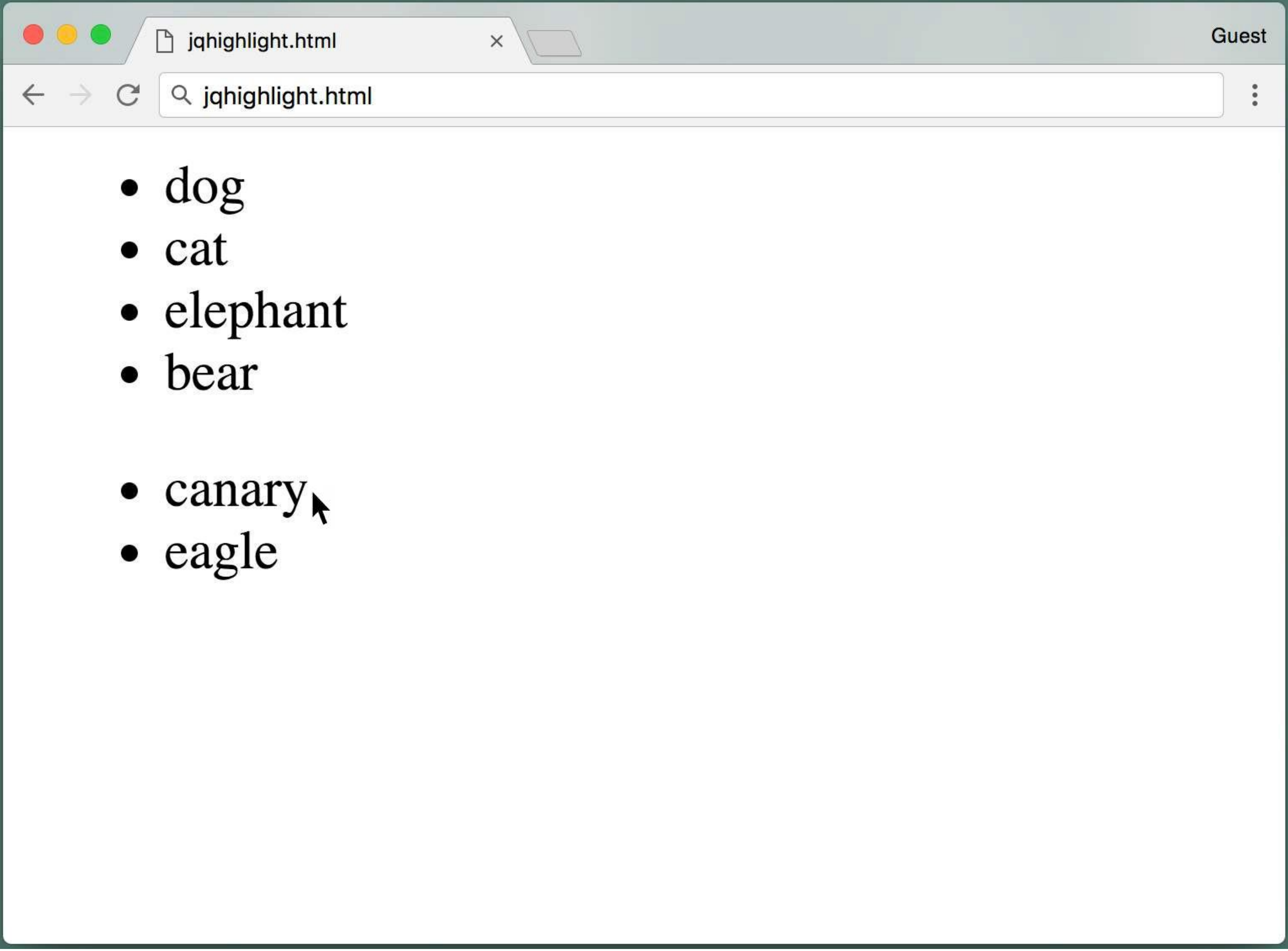
hover gets two functions

```
<html>
<head><script src="jquery.js"></script></head>
<body>
<ul>
<li>dog</li>
<li>cat</li>
<li>elephant</li>
<li>bear</li>
</ul>

<script>

$('li').on({
  mouseenter: function() {
     $(this).css('color', 'red');
     $(this).css('font-size', '120%');
  },
  mouseleave: function() {
     $(this).css('color', 'black');
     $(this).css('font-size', '100%');
  },
  click: function() {
     $(this).css('background-color', 'yellow');
  }
});


</script>

</body>
</html>
```

on allows to bind multiple event handlers to one item

```
<html>
<head><script src="jquery.js"></script></head>
<body>
<ul>
<li>dog</li>
<li>cat</li>
<li>elephant</li>
<li>bear</li>
</ul>

<script>

$('li').on({
  mouseenter: function() {
    $(this).css('color', 'red');
    $(this).css('font-size', '120%');
  },
  mouseleave: function() {
    $(this).css('color', 'black');
    $(this).css('font-size', '100%');
  },
  click: function() {
    $(this).css('background-color', 'yellow');
  }
});


</script>

</body>
</html>
```

on allows to bind multiple event handlers to one item



- dog
- cat
- elephant
- bear

- canary
- eagle

```
<html>
<head><script src="jquery.js"></script></head>

<body>
<ul>
<li class="highlight">dog</li>
<li class="highlight">cat</li>
<li class="highlight">elephant</li>
<li class="highlight">bear</li>
</ul>

<ul>
<li>canary</li>
<li>eagle</li>
</ul>

<script>

$("li.highlight").on({
mouseenter: function() { . . . },
mouseleave: function() { . . . },
click: function() { . . . }
});

</script>

</body>
</html>
```

In this example we have two different lists

```
<html>
<head><script src="jquery.js"></script></head>

<body>
<ul>
<li class="highlight">dog</li>
<li class="highlight">cat</li>
<li class="highlight">elephant</li>
<li class="highlight">bear</li>
</ul>

<ul>
<li>canary</li>
<li>eagle</li>
</ul>

<script>

$("li.highlight").on({
mouseenter: function() { . . . },
mouseleave: function() { . . . },
click: function() { . . . }
});

</script>

</body>
</html>
```

define a class for li items that you want to change

```
<html>
<head><script src="jquery.js"></script></head>

<body>
<ul>
<li class="highlight">dog</li>
<li class="highlight">cat</li>
<li class="highlight">elephant</li>
<li class="highlight">bear</li>
</ul>

<ul>
<li>canary</li>
<li>eagle</li>
</ul>

<script>

$("li.highlight").on({
mouseenter: function() { . . . },
mouseleave: function() { . . . },
click: function() { . . . }
});

</script>

</body>
</html>
```

only apply on li within the highlight class

```
<html>
<head><script src="jquery.js"></script></head>

<body>
<ul class="highlight">
<li>dog</li>
<li>cat</li>
<li>elephant</li>
<li>bear</li>
</ul>

<ul>
<li>canary</li>
<li>eagle</li>
</ul>

<script>

$("ul.highlight").find("li").on({
mouseenter: function() { . . . },
mouseleave: function() { . . . },
click: function() { . . . }
});

</script>

</body>
</html>
```

or define the whole list as highlight class and select lis inside that list

```
<html>
<head><script src="jquery.js"></script></head>

<body>
<ul class="highlight">
<li>dog</li>
<li>cat</li>
<li>elephant</li>
<li>bear</li>
</ul>

<ul>
<li>canary</li>
<li>eagle</li>
</ul>

<script>

$("ul.highlight").find("li").on({
mouseenter: function() { . . . },
mouseleave: function() { . . . },
click: function() { . . . }
});

</script>

</body>
</html>
```

find finds the items (li here) inside the selecter (ul with highlight class here)

# Advanced Selectors

- `$(`*`someNodes`*`).find(`*`selector`*`)` will search *someNodes*' children for selector.

# Advanced Selectors

- `$(`*`someNodes`*`).find(`*`selector`*`)` will search *someNodes*' children for selector.

- `$` selectors can be chained
  - `$("div.book")` selects the div with class="book"
  - `$("div, .book")` selects all divs **and** all elements with class="book"

# Advanced Selectors

- `$(`*`someNodes`*`).find(`*`selector`*`)` will search *someNodes*' children for selector.
- **$** selectors can be chained
  - `$("div.book")` selects the div with class="book"
  - `$("div, .book")` selects all divs **and** all elements with class=”book”
- : can be used to specify element properties
  - `$("p:hidden")` selects all `<p>` elements that are visually hidden

# Advanced Selectors

- `$(`*`someNodes`*`).find(`*`selector`*`)` will search *someNodes*' children for selector.
- **$** selectors can be chained
  - `$("div.book")` selects the div with class="book"
  - `$("div, .book")` selects all divs **and** all elements with class=”book”
- : can be used to specify element properties
  - `$("p:hidden")` selects all `<p>` elements that are visually hidden
- These selectors are all CSS selectors!
  - All other CSS selectors also work in **$**

# Summary

- jQuery can be used to retrieve and modify DOM elements and add event listeners

- We can use advanced selectors and handlers to simplify functionality