

# WEEK 7-8: JavaScript cont., jQuery

## Error handling in JavaScript

1. JavaScript error handling lets you test a block of statements and handle the error if it occur

```
<!DOCTYPE html>
<html>
<body>
<p id="message"></p>

<script>
try {
  println("When will I be famous?");
}
catch(err) {
  document.getElementById("message").innerHTML = err.message;
}
</script>
</body>
</html>
```

output:

println is not defined

With the throw statement a custom error can be raised.

```
<!DOCTYPE html>
<html>
<head>
<script>
function numbercheck()
{
  var message, num;
  message = document.getElementById("message");
  message.innerHTML = "OK";
  num = document.getElementById("mark").value;
  try
  {
```

```
    if(isNaN(num)) throw "not a number";
    if(num == "") throw "empty";

    num = Number(x);
    if(num < 1) throw "too low";
    if(num > 5) throw "too high";
  }
  catch(err)
  {
    message.innerHTML = "Grade: " + err;
  }
}
</script>
</head>

<body>

<h1>Grade (valid values are between 1 and 5)</h1>

<input id="mark" type="text">
<button type="button" onclick="numbercheck()">Test grade</button>
<p id="message"></p>

</body>
</html>
```

## JavaScript Arrow Function

Some JavaScript libraries rely heavily on arrow functions as they are very concise.

Evolution from normal function to arrow function:

Traditional function:

```
<!DOCTYPE html>
<html>
<head>
<script>
function hello ()
{
  document.write("Hello World!");
}
```

```
}  
hello();  
</script>  
</head>  
<body>
```

output:

Hello World!

The following syntax works similar:

```
<!DOCTYPE html>  
<html>  
<head>  
<script>  
hello = function()  
{  
  return "Hello World!";  
}  
document.write(hello());  
</script>  
</head>  
  
<body>
```

output:

Hello World!

Arrow function gets its name from an arrow (=>) in its definition:

```
<!DOCTYPE html>  
<html>  
<head>  
<script>  
hello = () =>  
{  
  return "Hello World!";  
}  
document.write(hello());  
</script>  
</head>
```

```
<body>
```

output:

Hello World!

Note that arrow function returns a value by default so it can be written in a simpler form when the function has only one statement:

```
<!DOCTYPE html>
<html>
<head>
<script>

hello = () => "Hello World!";

document.write(hello());
</script>
</head>

<body>
```

output:

Hello World!

Function parameters can be used to following way:

```
<!DOCTYPE html>
<html>
<head>
<script>
hello = (first,second) => "Hello, " + first + " and " + second;
document.write(hello("Johnny","Clyde"));
</script>
</head>
<body>
  <span id="message"></span>
</body>
<body>
```

output:

Hello, Johnny and Clyde

If using only one parameter, parentheses can be omitted:

```
<!DOCTYPE html>
<html>
<head>
<script>
hello = first => "Hello " + first;
document.write(hello("World"));
</script>
</head>
<body>
  <span id="message"></span>
</body>
<body>
```

output:

Hello World!

# jQuery

Javascript can do anything with a web page but sometimes seems over complicated and the programmer has to type a lot.

jQuery is a Javascript library which simplifies and speeds up Javascript programming.

jQuery manages cross-browser compatibility issues, simplify DOM manipulation, event handling, and AJAX.

jQuery was started in 2006 and has become the most popular JavaScript library.

jquery.com: "jQuery is a fast, small, and feature-rich JavaScript library. It makes things like HTML document traversal and manipulation, event handling, animation, and Ajax much simpler with an easy-to-use API that works across a multitude of browsers. With a combination of versatility and extensibility, jQuery has changed the way that millions of people write JavaScript."

```
<!DOCTYPE html>
<html>
<head>
<!-- always use latest version -->
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"></scri
pt>
<script>
$(document).ready(function()
{
  $("p").click(function()
  {
    $(this).css('color', 'red');
  });
});
</script>
</head>
<body>

<p>Paint me</p>
<p>Color me</p>
<p>Change my color</p>

</body>
</html>
```

Note: Libraries can be loaded from any URL, from a local file or from a CDN (Content Delivery Network, sometimes Content Distribution Network). Above example shows how to link jquery minimized version from Google Hosted Libraries.

Basic syntax:

```
<script>  
$(selector).action()  
</script>
```

Action can be an event in the browser or a method of the selected item (or items).

## jQuery events

jQuery events can belong to the following event groups: document, form, keyboard and mouse events. Some examples:

- Document events
  - load
  - unload
  - scroll
  - resize
- Form events
  - submit
  - change
  - focus
  - blur
- Keyboard events
  - keypress
  - keyup
  - keydown
- Mouse events
  - click
  - dblclick
  - mouseenter
  - mouseleave
  - mouseup
  - hover

In the above example we defined an action for the click event (on any paragraph (<p>) element):

```
$("#p").click(function()  
{  
    $(this).css('color', 'red');  
});
```

Wait for mouse crossing the horizontal ruler:

```
<!DOCTYPE html>  
<html>  
<head>  
<script  
src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></scri  
pt>  
<script>  
$(document).ready(function(){  
    $("#here").mouseenter(function(){  
        $("#message").html("You crossed the line");  
    });  
});  
</script>  
</head>  
<body>  
  
Move your mouse over the line:  
<hr id="here">  
<span id="message"></span>  
  
</body>  
</html>
```

The hover in the combination of the mouseenter and mouseleave functions.

```
<!DOCTYPE html>  
<html>  
<head>  
<script  
src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></scri  
pt>  
<script>  
$(document).ready(function(){
```



```
$( "#here" ).hover(function()  
  {  
    $( "#here" ).css("border-color", "yellow");  
  },  
  function()  
  {  
    $( "#here" ).css("border-color", "green");  
  });  
});  
</script>  
</head>  
<body>  
Move your mouse over the line:  
<hr id="here">  
</body>  
</html>
```

Styling can focus users eye to a specific part of a form:

```
<!DOCTYPE html>  
<html>  
<head>  
<script  
src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></scri  
pt>  
<script>  
$(document).ready(function(){  
  $("input").focus(function(){  
    $(this).css("background-color", "yellow");  
  });  
  $("input").blur(function(){  
    $(this).css("background-color", "lightgrey");  
  });  
});  
</script>  
</head>  
<body>  
  
First name: <input type="text" name="firstname"><br>  
Last name: <input type="text" name="lastname">  
  
</body>  
</html>
```

## jQuery selectors

Selectors are used to specify the set of HTML elements to which the given event applies. In other words, jQuery selectors are used to select HTML elements based on their id, name, classes, types, some attributes or values of attributes. jQuery selectors are mainly based on the existing CSS Selectors, and have some custom selectors. All selectors in jQuery start with the dollar sign and parentheses, just like a normal function called `$. ($())`.

### Selection by id

```
<script>
    $("#message").html("I'd hate to die twice. It's so boring.");
</script>
<span id="message"></span>
```

*Note: you may use id as unique. In other cases it will find the first occurrence but this cannot be expected in any way.*

### Selection by tag name

Change the color of all list elements:

```
<!DOCTYPE html>
<html>
<head>
<!-- always use latest version -->
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"></scri
pt>
<script>
$(document).ready(function()
{
    $("li").css("color", "hotpink");
});
</script>

</head>
<body>

<li>One</li>
<li>Two</li>
```

```
<li>Three</li>

</body>
</html>
```

output:

- One
- Two
- Three

## Selection by class

Signed elements will get pink color:

```
<!DOCTYPE html>
<html>
<head>
<!-- always use latest version -->
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"></scri
pt>
<script>
$(document).ready(function()
{
    $(".pink").css("color","hotpink");
});
</script>

</head>
<body>
<h2 class="pink">This is no way to live.</h2>
<li>One</li>
<li class="pink">Two</li>
<li>Three</li>

</body>
</html>
```

output:

## This is no way to live.

- One
- Two
- Three

### Other and mixed selections

Selecting the current HTML element: `$(this)`

The following code sets background to yellow only on clicked `<li>` elements:

```
<!DOCTYPE html>
<html>
<head>
<!-- always use latest version -->
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"></scri
pt>
<script>
$(document).ready(function()
{
    $("li").click(function(){
        $(this).css("background-color", "yellow");
    });
});
</script>

</head>
<body>

<li>One</li>
<li>Two</li>
<li>Three</li>

</body>
</html>
```

## Selecting all elements

```
$("#*").css("color", "black");
```

Selecting all <li> elements with class="pink"

```
$("li.pink")
```

```
<!DOCTYPE html>
<html>
<head>
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"></scri
pt>
<script>
$(document).ready(function()
{
    $("li.pink").css("color", "hotpink");
});
</script>

</head>
<body>
<h2 class="pink">Now why did I do that?</h2>
<li>One</li>
<li class="pink">Two</li>
<li>Three</li>

</body>
</html>
```

output:

## Now why did I do that?

- One
- Two
- Three

---

**Selects the first <li> element**

`$("li:first")`

```
<!DOCTYPE html>
<html>
<head>
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"></scri
pt>
<script>
$(document).ready(function()
{
    $("li:first").css("color","hotpink");
});
</script>

</head>
<body>
<h2 class="pink">Oh Lord, forgive the misprints!</h2>
<li>One</li>
<li class="pink">Two</li>
<li>Three</li>

</body>
</html>
```

output:

## Oh Lord, forgive the misprints!

- One
- Two
- Three

**Selects the last <li> element**

`$("li:last")`

```
$("li:last").css("color","hotpink");
```

`$("ul li:first")` Selects the first `<li>` element of the first `<ul>`

```
<!DOCTYPE html>
<html>
<head>
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"></scri
pt>
<script>
$(document).ready(function()
{
    $("ul li:first").css("color","hotpink");
});
</script>

</head>
<body>
<h2 class="pink">If the sun refused to shine, I would still be loving
you.</h2>

<ol>
<li>One</li>
<li>Two</li>
<li>Three</li>
</ol>

<ul>
<li>One</li>
<li>Two</li>
<li>Three</li>
</ul>

<ul>
<li>One</li>
<li>Two</li>
<li>Three</li>
</ul>

</body>
</html>
```

## If the sun refused to shine, I would still be loving you.

1. One
2. Two
3. Three

- One
- Two
- Three

- One
- Two
- Three

`$("ul li:first-child")`

Selects the first list index element of every unordered list:

```
<!DOCTYPE html>
<html>
<head>
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"></scri
pt>
<script>
$(document).ready(function()
{
    $("ul li:first-child").css("color","hotpink");
});
</script>

</head>
<body>
<h2 class="pink">I need you so much closer.</h2>

<ol>
<li>One</li>
<li>Two</li>
<li>Three</li>
</ol>
```



```
<ul>
<li>One</li>
<li>Two</li>
<li>Three</li>
</ul>

<ul>
<li>One</li>
<li>Two</li>
<li>Three</li>
</ul>

</body>
</html>
```

## I need you so much closer.

4. One
5. Two
6. Three

- One
- Two
- Three

- One
- Two
- Three

### Selecting an HTML element and an HTML attribute with the same name

`$(":button")` Selects all `<button>` elements and `<input type="button">` elements

```
<!DOCTYPE html>
<html>
<head>
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></scri
pt>
```

```
<script>
$(document).ready(function(){
  $("button").click(function(){
    $(".button").hide();
  });

  $("#butt").click(function(){
    $(".button").hide();
  });
});
</script>
</head>
<body>

<h2>Story of the Last Click</h2>

<button>Click me. It'll be your last click.</button>
<input id=butt type="button" value="Click me. It'll be your last click.">

</body>
</html>
```

### Selecting by HTML attribute

`$("[id]")` Selects all elements with an href attribute:

```
<!DOCTYPE html>
<html>
<head>
<!-- always use latest version -->
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"></scri
pt>
<script>
$(document).ready(function()
{
  $("[id]").css("background-color", "yellow");
});
</script>

</head>
<body>
```

```
<h2 class="pink">Push my buttons</h2>

<p id="message">Look my buttons</p>
<button id="1">Button</button>
<button id="2">Button</button>
<button >Button</button>
<p>The Pussycat Dolls</p>
</body>
</html>
```

### Push my buttons

Look my buttons

Button Button Button

The Pussycat Dolls

Selecting by the value on an HTML attribute

`$("#a[target='_blank']")` Selects all `<a>` elements where target attribute is `"_blank"` (link opens in new page):

```
<!DOCTYPE html>
<html>
<head>
<!-- always use latest version -->
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"></scri
pt>
<script>
$(document).ready(function()
{
    $("#a[target='_blank']").css("background-color","yellow");
});
</script>

</head>
<body>
<h2 class="pink">The heart of a father is the masterpiece of nature. Find a
link to him.</h2>

<ul>
<li><a target="_blank" href=new.html>Open link in new tab</a>
<li><a target="webshop" href=new.html>Open link in page (tab or frame)
```

```
called "webshop"</a>
  <li><a target="_self" href=new.html>Open link in the same page</a>
  <li><a target="_parent" href=new.html>Open link in the parent frame</a>
  <li><a target="_top" href=new.html>Open link in the "full" body of the
window</a>
</ul>
</body>
</html>
```

### The heart of a father is the masterpiece of nature. Find a link to him.

- [Open link in new tab](#)
- [Open link in page \(tab or frame\) called "webshop"](#)
- [Open link in the same page](#)
- [Open link in the parent frame](#)
- [Open link in the "full" body of the window](#)

*Note: target="\_self" is default behaviour of links.*

Instead of =, non-equal symbol (!=) can be used to select all values of HTML attributes except the given one. For example, `$("a[target!='_self']")` selects all `<a>` elements not opening in the same frame:

```
$("a[target!='_self']").css("background-color","pink");
```

### Selecting odd elements:

Selects all odd `<tr>` elements: `$("tr:odd")`

```
<!DOCTYPE html>
<html>
<head>
<!-- always use latest version -->
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"></scri
pt>
<script>
$(document).ready(function()
{
  $("tr:odd").css("background-color","grey");
```

```
});  
</script>  
  
</head>  
<body>  
<h2 class="pink">I love odd numbers so much.</h2>  
  
<table>  
<tr><th>Numbers</th></tr>  
<tr><td>One</td></tr>  
<tr><td>Two</td></tr>  
<tr><td>Three</td></tr>  
<tr><td>Four</td></tr>  
<tr><td>Five</td></tr>  
</table>  
  
</body>  
</html>
```

output:

I love odd numbers so much.

Numbers

One

Two

Three

Four

Five

### Selecting even elements

`$("tr:even")`    Selects all even `<tr>` elements

```
$("tr:even").css("background-color","lightgrey");
```