

to load a JavaScript library:

1. create new file
2. in file, define a function
3. function should provide feedback (alert, console log)
4. in html, load file using script tag

```
my_library.js
function shoutHello() {
  alert('Hello World');
}
<html>
  <head>
    <script src="my_library.js"></script>
  </head>
  <body>
    <script>
      shoutHello();
    </script>
  </body>
</html>
```

5. call the function in an arbitrary <script> block after your library has been loaded with a <script> tag

to include the jQuery library:

1. get link of latest version on Google Developers CDN (Content Delivery Network) index
2. download a copy and provide a relative path in your project, reference a <script> tag (cut and paste in)

```
<script src="https://ajax.googleapis.com/
ajax/libs/jquery/2.1.4/jquery.min.js"></script>
always first, then your js file link:
<script src="javascript_file.js"></script>
```

API library documentation explains functions

<http://api.jquery.com/>

always use	<pre> \$(x) - x is a string representing a CSS selector \$('body'); \$('#events-list'); \$('#events-list li'); \$('.event'); function \$(selector) { var result; return result; } </pre>
function takes as an argument a string representing a CSS selector, returns an array of elements matching that selector	
show or hide element on a page	<pre> \$('document').hide(); - like display: none \$('body').show(); </pre>
show or hide element on a page using fading animation	<pre> \$('h1').fadeOut(); \$('h2').fadeIn(); \$('h3').fadeOut(1000); - 1000 is 1 second </pre>
an optional argument, time in milliseconds, will fade out over 1 second	
slide elements out of view (hide)	<pre> \$('h1').slideUp(); \$('h2').slideDown(); \$('h3').slideUp(1000); </pre>
"animate" most CSS properties that are numeric	<pre> \$('my-element').animate({ opacity: 0.25, width: "70%" }, 2000); </pre>
change an element when it's clicked	<pre> \$('my-element').click() </pre>
button click	<pre> html: <button>click me</button> js: \$(document).ready(function() { \$('button').click(function() { console.log('clicked'); }); }) </pre>
click anywhere on page - console log	<pre> no html, just js: \$(document).ready(function() { \$('body').click(function() { console.log('clicked'); }); }) </pre>
click anywhere on page - h1 and h2 hide	<pre> \$(document).ready(function() { \$('body').click(function() { \$('h1').hide(); \$('h2').hide(); }); }) </pre>

anonymous functions and callbacks

named function

shoutHello - function name, shoutString - variable name

to call a named function

unnamed/anonymous function

to call an anonymous function

1. assign your function to a variable

once assigned, call using var name

2. pass your anonymous function as an argument to another function

this is a callback - a function called within another function

1. typically called when parent function finishes executing

2. parent calls back with the child function

setting up an event listener but the element to which we are listening is not yet present, so use:
conventional to wrap all your code in this callback so all elements you wish to select will be on the page

to use, pass an anonymous function as a callback to the ready method

```
function shoutHello(shoutString) {  
  alert('Hello ' + shoutString + '!');  
}  
shoutHello('class');
```

```
function(shoutString) {  
  alert('Hello ' + shoutString + '!');  
}
```

```
var shoutGoodbye = function(shoutString) {  
  alert('Goodbye ' + shoutString + '...');  
}  
shoutGoodbye('stranger');
```

```
function loopThenExec(num, func) {  
  for (var i=0, i<100, i++) {  
    console.log(i);  
  }  
  func();  
}
```

```
loopThenExec(1000, function() {  
  console.log('Loop is complete!');  
})
```

```
$(".menu").show( function() {  
  $(".menu").animate({ opacity: .5});  
});
```

```
<script>  
  $('button').click(function() {  
    console.log('Hello world!');  
  });  
</script>  
<button>Click me!</button>
```

```
$(document).ready() {  
  //the code that will execute once document has loaded  
});  
$(document).ready(function() {  
  //the code that will execute once document has loaded  
});
```

```
$(document).ready (  
  //the code that will execute once document has loaded  
  function(){  
    alert('Document is ready!');  
  }  
);
```

#1: your function

```
function() {  
  console.log('Hello');  
}
```

and your jQuery effect

```
$('#h1').hide()
```

#2: put the function in the jQuery effect's variable name brackets

```
$('#h1').hide(function() {  
  console.log('Hello');  
})
```

#3: add ready method

```
$(document).ready()
```

with your new function

```
function() {  
  $('#h1').hide(function() {  
    console.log('Hello');  
  });  
}
```

4. put the new function in the ready method's brackets

```
$(document).ready(function() {  
  $('#h1').hide(function() {  
    console.log('Hello');  
  });  
})
```

this is all you need and it will hide h1 headlines and print 'Hello' in the console

5. many can be grouped in one giant anonymous function of `(document).ready`