| Sprint 10, Assignment 4.4 Please also update the doc name with correct numbers. | |
| --- | --- |
| Assignment type: JS Interactive | |
| Assignment name  Can remain the same as the assignment topic name, or…  can be phrased in the active/imperative voice (depending on the task), e.g. “Create your own X” | “Insert JQuery library & Test” |
| Description  The essential tasks a student must perform in order to complete the assignment  ?+ a general, colorful introduction to the assignment? | While applying to various companies, you've come across a few who have asked for a working understanding of JQuery in order to deal with their aging code base. Given that you're becoming a JS savant, you have decided to download it and give it a try. Dig up any notes and code files you might have from lesson 3.2 ("JavaScript Events"), because you are going to try and duplicate your vanilla JavaScript code:  add JQuery to a copy of your 3.2 code file (or using the code in step 0), |
| Link to full code file (Answer)  Push a full code file to our [GitHub repo](https://github.com/bitdegree-foundation/academy-assignments-code-files) for this sprint’s assignment and paste a link to that file here. Please write the file in a concise and clear format, according to the “[example-js-code-file](https://github.com/bitdegree-foundation/academy-assignments-code-files/blob/master/example-js-code-file.js)” found in our repo. | (Upload) |

| # | Step  Write each small step of the task. These assignments should be doable by students on their own (without BitDegree-instructor help). It can be any number of steps, but keep them pretty clear and separate (don’t combine 2 steps into 1 step). | 1x hint  Write brief text or partial code that will help the student figure out how to either move forward or completely accomplish the given step. | Output Expected code that our platform will be able to run & check against student submissions |
| --- | --- | --- | --- |
| 0 | If you were not able to re-create the code from lesson 3.2, paste the following into a new html document:  <!DOCTYPE *html*>  <html *lang*="en">  <head>  <meta *charset*="UTF-8" />  <meta *http-equiv*="X-UA-Compatible" *content*="IE=edge" />  <meta *name*="viewport" *content*="width=device-width, initial-scale=1.0" />  <style>  .box {  width: 200px;  margin: 10px;  border: 4px solid black;  font-size: 300%;  text-align: center;  padding: 30px;  }  </style>  <title>  Sprint 10, Lesson 4.4 & 4.5, ("Insert JQuery Library & Test")  </title>  </head>  <body>  <div *class*="container">  <div *class*="box" *id*="box1">Box 1</div>  <div *class*="box" *id*="box2">Box 2</div>  </div>  <script *src*="./4.4.script.js"></script>  </body>  </html> |  |  |
| 1 | Go to https://code.jquery.com/ | Note: if this becomes broken, go to https://jquery.com/, click the download tab, and then look for language on the page that includes "to use the jquery CDN" |  |
| 2 | Select the "minified" version and copy the link | Note: if, later, you want to look under the hood, go back and examine the "uncompressed" version |  |
| 3 | Paste what you've copied into your html document | Hint: it can go in the head of your document  Hint: in theory, it is code that normally goes inside of <script> tags. However, see below note -  Note: you should not need to create "script" tags, as what you've copied (likely) already has them. | <!DOCTYPE *html*>  <html *lang*="en">  <head>  <meta *charset*="UTF-8" />  <meta *http-equiv*="X-UA-Compatible" *content*="IE=edge" />  <meta *name*="viewport" *content*="width=device-width, initial-scale=1.0" />  <style>  .box {  width: 200px;  margin: 10px;  border: 4px solid black;  font-size: 300%;  text-align: center;  padding: 30px;  }  </style>  *<!-- STEP 3 addition: -->*  <script  *src*="https://code.jquery.com/jquery-3.6.0.min.js"  *integrity*="sha256-/xUj+3OJU5yExlq6GSYGSHk7tPXikynS7ogEvDej/m4="  *crossorigin*="anonymous"  ></script>  <title>Sprint 10, Lesson 4.4, ("Insert JQuery Library & Test")</title>  </head>  <body>  <div *class*="container">  <div *class*="box" *id*="box1">Box 1</div>  <div *class*="box" *id*="box2">Box 2</div>  </div>  <script *src*="./4.4.Assignment.script.js"></script>  </body>  </html> |
| 4 | Create a new JavaScript file named "4.4.script.js |  |  |
| 5 | Store both box elements in new, JQuery-styled variables (remember to declare them using the JQuery dollar-sign convention in order to avoid confusion later). |  | const $box1 = $("#box1");  const $box2 = $("#box2"); |
| 6 | Add a double-click listener for your JQuery box1 object and have it log something like "JQuery was used on a double-click!" | Hint: instead of using querySelector etc, use the $() syntax | $("#box1").dblclick(function(){  console.log("JQuery was used on a double-click!");  }) |
| 7 | Pass the event through the function and store it in a variable; log the element |  | $("#box1").dblclick(function(e){  console.log("JQuery was used on a double-click!");  const element = e.target;  console.log(element);  }) |
| 8 | Make the dblclick event toggle the size of the box from normal to large | Hint: use a conditional | $("#box1").dblclick(function(e){  console.log("JQuery was used on a double-click!");  const element = e.target;  if (element.style.fontSize == "100px"){  element.style.fontSize = "45px"  }else{element.style.fontSize = "100px"};  }) |
| 9 | Add a mouseenter listner for your JQuery box2 object and have it log something like "JQuery was used on a mouse hover!" |  | $("#box2").mouseenter(function(e){  console.log("JQuery was used on a mouse hover!");  }) |
| 10 | Pass the event through the function and store it in a variable; log the element |  | $("#box2").mouseenter(function(e){  const element = e.target;  console.log(element);  }) |
| 11 | Make the mouseenter event toggle the border color from black to red | Hint: use a conditional | $("#box2").mouseenter(function(e){  const element = e.target;  if (element.style.borderColor == "black"){  element.style.borderColor = "red"  }  else{element.style.borderColor = "black"}  }) |
| 12 | Bonus: make a mouseleave listener that will toggle the border from red to black (so that now, the border will change color every time the mouse either enters or leaves!) | Hint: reverse the conditional logic of the mouseenter | $("#box2").mouseleave(function(e){  const element = e.target;  if (element.style.borderColor == "red"){  element.style.borderColor = "black"  }  else{element.style.borderColor = "red"}  }) |
| ... |  |  |  |