WEEK 6 ASSESSMENT

**HTML, CSS, JAVASCRIPT, JQUERY, ANGULAR, NODE**

# IN YOUR TEXT FILE, WRITE THE CODE TO COMPLETE THE FOLLOWING

HTML / CSS

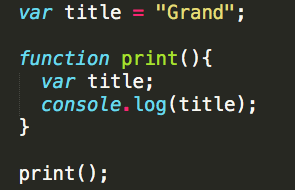
1. Create a paragraph tag with the text “Lorem Ipsum” as its content.
2. Create an anchor tag that links to “http://www.google.com” and the text ‘Google It!’ as its content.
3. Create a div element with a class of “wrapper”.
4. Create a paragraph element with a class of ‘product-description’. Inside of it, add a span element with a class of ‘highlight’.
5. Create a div element with an attribute of data-file with a value of “names.txt”.
6. Create a CSS selector that selects all h2 elements.
7. Create a CSS selector that selects all elements that have the ‘info-detail’ class applied to it.
8. Create a CSS selector that selects an element with an ID of ‘navigation’.
9. Create a CSS selector that selects all paragraph tags nested in elements with a class of “subscription-info” that is nested in an element with a class of ‘subscription’
10. Create a CSS declaration that will style the selected elements’ text blue.
11. Create a CSS declaration that will style the background of the selected element to be #ffAA00.
12. There are 4 ways to specify color in CSS. List at least 3 of them and give examples of their use.
13. True or False: When an element has the float property applied to it, it is removed from the normal document flow.
14. True or False: When an element has its position property set to relative, it is removed from the normal document flow.
15. Create a CSS declaration that will apply a red border to the bottom of an element that is 3 pixels wide and uses a double line style.
16. Taking the box model into account: If an element has a width of 150 pixels, padding of 10 pixels, a 5px border, and margin of 10 pixels. What is the total width of the element.
17. Create a CSS selector for an anchor element that will only apply its style when the user’s mouse passes over element.
18. Create a CSS rule that will import an external font.
    1. The name the font will be referred to as in the css file will be “Open Sans”.
    2. The path to the font file will be “css/fonts/open-sans/open-sans.otf”
19. Create a CSS declaration that will create rounded corners of 10 pixels for the element it is applied to.
20. Create a CSS declaration that will set the element’s typeface to Open Sans. If Open Sans is not available, it will use Helvetica. If Helvetica is not available, it will use Arial. If Arial is not available, it will use any sans-serif font.

JAVASCRIPT

1. Declare a variable called currentScore and initialize it to the number 42.
2. What are the six data-types in JavaScript?
3. There are several falsy values in JavaScript. Name three.
4. Construct an if statement to check if the variable limit is above or equal to 21. If true, the script should log the message, ‘limit is met or exceeded’.
5. Create a for loop that logs the integers from 1 to 1000 to the console.
6. Create a while loop that logs the integers from 1 to 1000 to the console.
7. Declare a function called processEntries(). Inside the body of the function, write a comment that reads, “Process logic here”.
8. Declare a function called storeCustomer() that accepts a parameter called customer. Inside the body of the function, return the parameter customer.
9. Call the function storeCustomer() with the argument of “Aisha”.
10. Declare a variable called detroitTeams, initialize it as an array of strings with the values “Tigers”, “Red Wings”, “Pistons” and “Lions”.
11. Declare a variable called numbers, initialize it as an array of numbers with the values 1, 2, and 3. Using the array method forEach(), log each element of the array to the console.
12. Declare a variable called catalog, initialize it as an array of objects. Each object should have properties of productName, description and unitPrice. Add an object for each of these catalog items:

|  |  |  |
| --- | --- | --- |
| Product Name | Description | Price per unit |
| Lamp | “Standing lamp.” | 8.73 |
| Chair | “What you sit in.” | 66.35 |
| Paperweight | “For holding things down.” | 3.46 |

1. Create a variable called rows. Initialize rows to a collection of elements with the class of table-row using the document method getElementsByClassName().
2. Create a variable called summary. Initialize summary to an element with the id of title, using the document method getElementById(). Set summary’s innerText property equal to the string “Lorem Ipsum”.
3. Consider the following code, what will be printed to the log? In a few sentences, explain why.



JQUERY

1. Rewrite this line using a jQuery selector: document.getElementById(‘summary’);
2. There are two ways to write use jQuery’s ready function to delay execution of your Javascript until the document has loaded. Write one.
3. Using jQuery, select an element with an id of submit-button. Using the on() method, the first argument should indicate the handler is for the click event and the second argument should be an anonymous function.
4. Using jQuery’s $.get() function, make a call to the url “http://example.com/json”. In the second argument, provide a callback that will log the data returned from the url to the console.
5. Using jQuery, select an element with the id of “nav-item” and apply the css property of “background-color: #FA0”.

ANGULAR

1. Create an html body element and attach the ng-app directive with a value of “storefront”.
2. Create an input element with an attribute type of “text”. Add an ng-model directive with a value of “message”.
3. Create an h1 element. Inside of it, add an angular expression with the property message inside of it.
4. Create an li element, add an ng-repeat directive to it. Set the value of the ng-repeat to be “item in items”. Then, inside the li element, add an angular expression with item inside it.
5. Create an div element, add an ng-repeat directive to it. Set the value of the ng-repeat to be “product in products”. Inside the div element, add a paragraph element with an angular expression. Inside the angular expression, add product.price and apply the currency filter.
6. Create an angular controller, name it “checkoutView”, pass it an anonymous callback as its second argument, inject the ‘$scope’ and the ‘$interval’ objects.
7. Declare a new angular module called “legal”. Inject the ‘ngRoute’ and ‘ui.bootstrap’ module dependencies.
8. Create an angular service called “customerInfo”, pass it an array as its second argument. The array should be populated with a string value of “$http” for its first element and an anonymous function as its second element. Pass the “$http” service into the anonymous function as an argument.
   1. Inside the anonymous function, create a variable called account and initialize it to an empty object.
   2. Declare a function called getAccountInfo() that accepts a parameter of “accountId”. Inside getAccountInfo() return the $http function call with an object literal as its argument. The object literal should include the following properties:

|  |  |
| --- | --- |
| Key | Value |
| url | “some/url” |
| method | “GET” |
| Params | { id : accountId } |

* 1. The anonymous function should return an object literal. The object literal should have a property of account with a value of account, and a property of getAccountInfo with a value of getAccountInfo.

1. Create an angular directive called “customerDetails”. Pass an anonymous function in as the second argument.
   1. The anonymous function should return an object literal with three properties.
   2. The first should be the property restrict with the value of “AE”.
   3. The second property should be templateUrl with the value of “path/to/my/view.html”
   4. The third property should be link with the value of an anonymous function. The function should accept three parameters: scope, elem, & attrs.
   5. In the body of the link property function, assign the value “John Doe” to the scope.name property.
2. Consider the directive created in question 49: Write the HTML that would add this directive to a view.

NODE.JS

1. Declare a variable called http and initialize it to the node module “http” using the require() function.
2. Given you have a JavaScript program in a file called “example.js”. Write the terminal command to execute this program using Node.
3. Write the terminal command to globally install the node module stylus.
4. Declare a variable player and initialize it to a node module called “media-player”.
   1. Declare a function play() that accepts a parameter songId.
   2. Inside the play() function, execute the play function on the player object passing in the songId.
   3. Export the play() function by reference using module.exports.