

Exercise - A Sandwich Calculator

1. Write a function called sandwich calculator. This should accept one value: slicesOfBread
2. The function should return the total number of possible sandwiches based on the amount of slices available, so if there are 10 slices, it should return 5. Test your function with an alert box.
3. Extend your function so it accepts two values, slicesOfBread and slicesOfCheese.
4. It takes two slices of bread and one of cheese to make a sandwich. The function should return the total number of possible sandwiches, so if there are 10 slices of bread, but only 1 of cheese, it should return 1. You'll need an if statement to make this work.

Exercise - Pizza

1. Create an array of pizza toppings
2. Write a function that we can call
3. Function should return a string that describes the pizza, example "A delicious pizza with Cheese and Pepperoni and Garlic and Peppers."

Exercise - Capitalise every element

1. Write a short function to capitalise every element of the array.
2. You must use the Array#map function and the String#uppercase function.

Exercise - Styling

1. Use jQuery to select the body element
2. Use the .css() function to style the body. Give it a red background and blue text.

Exercise - Append

1. Use jQuery to select the body element
2. Use the .append() function to append a new hyperlink.
3. Use the .attr() function to set an href attribute on the hyperlink. Open it in your DOM and check it's clickable.

Exercise - Make it red

1. Create a web page with half a dozen paragraphs on it.
2. Add a click handler. When you click a paragraph it goes red.

Exercise - Keyup

1. Create a text input field. Listen for the keyup event. Alert the value of the text box whenever keyup occurs.
2. Get the value of an input field using: `$('#input').val();`
3. For bonus points, append the value of the text box into a div.

Exercise - Rollover

1. Review the list of events on: <http://api.jquery.com/category/events/>
2. Create code that listens for a hover event. When the user rolls over a div, place the word "over" in that div.
3. When the user mouses out of the div, place the word "out" in the div.

Exercise - A big mess

You can generate a random integer between 0 and 500 using the following code:

```
Math.random() * 500
```

You can generate a jQuery object that contains everything on the page inside the body element like so:

```
$('#body *');
```

1. Write a jQuery function that iterates over everything on the page using each, makes them all position:absolute, and assigns random top and left values. You need to use each here to give them all different random values.

Even More

1. Write a program that prints 'Hello World' to the screen.
2. Write a program that asks the user for their name and greets them with their name.
3. Modify the previous program such that only the users Alice and Bob are greeted with their names.
4. Write a program that asks the user for a number n and prints the sum of the numbers 1 to n .
5. Modify the previous program such that only multiples of three or five are considered in the sum, e.g. 3, 5, 6, 9, 10, 12, 15 for $n=17$.
6. Write a program that asks the user for a number n and gives them the possibility to choose between computing the sum and computing the product of $1, \dots, n$.
7. Write a program that prints a multiplication table for numbers up to 12.
8. Write a program that prints all prime numbers. (Note: if your programming language does not support arbitrary size numbers, printing all primes up to the largest number you can easily represent is fine too.)
9. Write a guessing game where the user has to guess a secret number. After every guess the program tells the user whether their number was too large or too small. At the end the number of tries needed should be printed. It counts only as one try if they input the same number multiple times consecutively.
10. Write a program that prints the next 20 leap years.
11. Write a function that returns the largest element in a list.
12. Write a function that reverses a list, preferably in place.
13. Write a function that checks whether an element occurs in a list.
14. Write a function that returns the elements on odd positions in a list.
15. Write a function that computes the running total of a list.
16. Write a function that tests whether a string is a palindrome.
17. Write three functions that compute the sum of the numbers in a list: using a for-loop, a while-loop and recursion. (Subject to availability of these constructs in your language of choice.)

18. Write a function `on_all` that applies a function to every element of a list. Use it to print the first twenty perfect squares. The perfect squares can be found by multiplying each natural number with itself. The first few perfect squares are $1*1=1$, $2*2=4$, $3*3=9$, $4*4=16$. Twelve for example is not a perfect square because there is no natural number m so that $m*m=12$. (This question is tricky if your programming language makes it difficult to pass functions as arguments.)
19. Write a function that concatenates two lists. $[a,b,c], [1,2,3] \rightarrow [a,b,c,1,2,3]$
20. Write a function that combines two lists by alternately taking elements, e.g. $[a,b,c], [1,2,3] \rightarrow [a,1,b,2,c,3]$.
21. Write a function that merges two sorted lists into a new sorted list. $[1,4,6],[2,3,5] \rightarrow [1,2,3,4,5,6]$. You can do this quicker than concatenating them followed by a sort.
22. Write a function that rotates a list by k elements. For example $[1,2,3,4,5,6]$ rotated by two becomes $[3,4,5,6,1,2]$. Try solving this without creating a copy of the list. How many swap or move operations do you need?
23. Write a function that computes the list of the first 100 Fibonacci numbers. The first two Fibonacci numbers are 1 and 1. The $n+1$ -st Fibonacci number can be computed by adding the n -th and the $n-1$ -th Fibonacci number. The first few are therefore 1, 1, $1+1=2$, $1+2=3$, $2+3=5$, $3+5=8$.
24. Write a function that takes a number and returns a list of its digits. So for 2342 it should return $[2,3,4,2]$.
25. Write a function that translates a text to Pig Latin and back. English is translated to Pig Latin by taking the first letter of every word, moving it to the end of the word and adding 'ay'. "The quick brown fox" becomes "Hetay uickqay rownbay oxfay".